Computer Physics Communications 185 (2014) 1606–1615

Contents lists available at ScienceDirect

Computer Physics Communications

journal homepage: www.elsevier.com/locate/cpc

# Local artificial boundary conditions for Schrödinger and heat equations by using high-order azimuth derivatives on circular artificial boundary

Hongwei Li [a,*], Xiaonan Wu [b], Jiwei Zhang [c]

[a] School of Mathematical Sciences, Shandong Normal University, Jinan, 250014, PR China
[b] Department of Mathematics, Hong Kong Baptist University, Hong Kong, China
[c] Courant Institute of Mathematical Sciences, New York University, New York, NY 10012, USA

ARTICLE INFO

*Article history:*
Received 19 January 2013
Received in revised form
8 February 2014
Accepted 3 March 2014
Available online 11 March 2014

*Keywords:*
Exterior problems
Circular artificial boundary
Local artificial boundary conditions
Auxiliary variables
Finite difference method

ABSTRACT

The aim of the paper is to design high-order artificial boundary conditions for the Schrödinger equation on unbounded domains in parallel with a treatment of the heat equation. We first introduce a circular artificial boundary to divide the unbounded definition domain into a bounded computational domain and an unbounded exterior domain. On the exterior domain, the Laplace transformation in time and Fourier series in space are applied to achieve the relation of special functions. Then the rational functions are used to approximate the relation of the special functions. Applying the inverse Laplace transformation to a series of simple rational function, we finally obtain the corresponding high-order artificial boundary conditions, where a sequence of auxiliary variables are utilized to avoid the high-order derivatives in respect to time and space. Furthermore, the finite difference method is formulated to discretize the reduced initial–boundary value problem with high-order artificial boundary conditions on a bounded computational domain. Numerical experiments are presented to illustrate the performance of our method.

## 1. Introduction

In this paper we consider the design of high-order local artificial boundary conditions (ABCs) for the linear partial differential equations (PDEs) in the forms of

$$u_t(\mathbf{x},t) = a\Delta u(\mathbf{x},t) + f(\mathbf{x},t), \quad (\mathbf{x},t) \in \Omega \times (0,T], \tag{1}$$

$$u(\mathbf{x},0) = u_0(\mathbf{x}), \quad \mathbf{x} \in \Omega, \tag{2}$$

$$u(\mathbf{x},t) \to 0, \quad \text{as } |\mathbf{x}| \to +\infty, \tag{3}$$

where $\Delta$ denotes the two-dimensional spatial Laplace operator, the source term $f(\mathbf{x},t)$ and initial condition $u_0(\mathbf{x})$ are compactly supported functions, and $T$ presents the final time point. Here $a$ can be endowed with different values, that is to say, if $a = 1$, Eq. (1) is the heat equation, if $a = i$, Eq. (1) represents the Schrödinger equation. Many problems in computational physics require the solutions of these two equations. For example, the heat equation on unbounded domains arises as a modeling task in a variety of heat diffusion, fluid dynamics, and financial applications. The Schrödinger equation is one of the basic equations of quantum mechanics, and it has many other important applications including electromagnetic wave propagation, plasma physics, seismic migration, etc. One quite important application of the Schrödinger equation, especially in a circular domain, arises in the context of wave propagation in optical fibers.

The numerical solution and analysis of this kind of linear PDEs on unbounded domains have so far received many attentions. To overcome the unboundedness of the defined domain, the customary way is to truncate the unbounded domain around the region of interest and implement the artificial boundary conditions on the truncation boundary. Givoli [1] introduced DtN artificial boundary condition on the given artificial boundary to study the heat problems on unbounded domains. Greengard and Lin [2] proposed a new algorithm which is based on the evolution of the continuous spectrum of the solution for solving the parabolic problem on unbounded domains. Han and Huang [3,4] constructed an exact artificial boundary condition to reduce the original heat

[*] Corresponding author. Tel.: +86 15066658261.
*E-mail addresses:* lhwqust@gmail.com (H. Li), xwu@hkbu.edu.hk (X. Wu), jzhang@cims.nyu.edu (J. Zhang).

http://dx.doi.org/10.1016/j.cpc.2014.03.001
0010-4655/© 2014 Elsevier B.V. All rights reserved.

equation to an initial–boundary-value problem on a finite computational domain. Wu and Sun [5] proved the unconditional stability of the heat equation, when cooperated with a delicately designed finite difference scheme. Wu and Zhang [6] constructed the high-order local absorbing boundary conditions for the heat equation, and avoided high-order derivatives by using auxiliary variables on the artificial boundary. Han and Huang [7] got the exact artificial boundary condition for the two-dimensional Schrödinger equation via the expression of the solution by Hankel functions. Baskakov and Popov [8] obtained the exact transparent boundary conditions for the Schrödinger equation in halfplane. Fevens and Jiang [9] constructed some local artificial boundary conditions for the Schrödinger equation. X. Antoine and his co-authors [10] constructed and studied a Crank–Nicolson-type discretization of the two-dimensional linear Schrödinger equation using the non-reflecting boundary conditions. In [11], the authors proposed discrete transparent boundary conditions for the time-dependent Schrödinger equation on a circular computational domain, and illustrated the accuracy, stability, and efficiency of the proposed method. A. Schädle [12] considered the nonreflecting boundary conditions for the two dimensional Schrödinger equation on unbounded domain. The absorbing boundary conditions for the general nonlinear Schrödinger equation and the two dimensional Schrödinger equation with an exterior potential on unbounded domain were obtained by X. Antoine et al. [13–15]. A. Arnold et al. [16] proposed a way to efficiently treat the well-known transparent boundary conditions for the Schrödinger equation and proved the stability of the resulting initial–boundary value scheme. Recently, Zhang et al. [17] designed high-order ABCs for the one dimensional Schrödinger equation and gave the second-order finite difference scheme for the reduced initial–boundary value problem. For many other great contributions made on the artificial boundary method, one can see [18–44] and the references therein.

Generally, the artificial boundary can be divided into polygon, circle or ellipse and so on. For two-dimensional domain, if we use the rectangle as artificial boundary, it is hard to get suitable high-order boundary conditions at corners. To avoid this difficulty, it is natural to utilize the circular artificial boundary. This paper focuses on high-order artificial boundary conditions for model Eq. (1) on the circular artificial boundary $\Gamma_R = \{(r, \theta)|r = R, 0 \le \theta < 2\pi\}$ with radius $R$, which divides the unbounded domain $\Omega$ into two parts, denote the computational domain by $\Omega_{in}$, and the exterior domain by $\Omega_e = \Omega\backslash\Omega_{in} = \{(r, \theta)|0 \le \theta < 2\pi, R < r < \infty\}$. We first rewrite the model Eq. (1) in a polar coordinate form, then use the Laplace transformation in time and Fourier series in space to arrive at the relation of the modified Bessel functions $K_n(x)$ $(n = 0, 1, \dots)$ by turning to condition (3). After applying the high-order rational functions to approximate the combination of the modified Bessel function, and using the inverse Laplace transformation, we obtain the high-order ABCs. For many proposed strategies of rational approximations, we refer the reader to [45–47] and references therein.

There are two attractive computational features [18] for the corresponding ABCs. First, the boundary conditions substantially reduce the (unphysical) reflections from the artificial boundary; second, the boundary conditions are local. Somehow we cannot prove that the boundary conditions together with the interior differential equation define a well-posed problem. However, the procedure for constructing the ABCs reveals that the ABCs annihilate only the energy (or wave flow) arising in the truncated computational domain; it will not propagate energy (or wave flow) into the interior domain to disrupt the true solution. Thus, the perturbation appearing in the artificial boundaries will have no (or small) effect on the interior solution, and hence the obtained high-order ABCs are efficient and stable. Numerical examples also demonstrate the stability of our method.

The paper is organized as follows. In Section 2, the high-order local artificial boundary conditions are constructed by approximating the modified Bessel function at the given artificial boundary. To illustrate the construction of high-order ABCs, we consider two types of PDEs (the Schrödinger equation and the heat equation), respectively. We obtain the reduced initial–boundary value problem on the bounded computational domain. In Section 3, the finite difference formulation is presented for the Schrödinger equation with high-order ABCs on a bounded computational domain. In Section 4, numerical examples are reported to demonstrate the performance of our method. We end this paper with some concluding remarks.

## 2. Construction of high-order local artificial boundary conditions

First, the Schrödinger equation is considered to introduce the basic idea how to design high-order ABCs, and in parallel with treating the heat equation in Section 2.2.

### 2.1. Schrödinger equation

To design high-order artificial boundary conditions, we rewrite the Schrödinger equation on the exterior domain in the polar coordinates $(r, \theta)$ in the form of

$$i\frac{\partial u}{\partial t} = -\left(\frac{\partial^2 u}{\partial r^2} + \frac{1}{r}\frac{\partial u}{\partial r} + \frac{1}{r^2}\frac{\partial^2 u}{\partial \theta^2}\right), \quad \text{in } \Omega_e \times (0, T]. \tag{4}$$

$$u|_{t=0} = 0, \quad \text{in } \Omega_e. \tag{5}$$

$$u \to 0, \quad \text{as } r \to +\infty. \tag{6}$$

Applying the Laplace transform to Eq. (4), we have

$$is\hat{u} = -\left(\frac{\partial^2 \hat{u}}{\partial r^2} + \frac{1}{r}\frac{\partial \hat{u}}{\partial r} + \frac{1}{r^2}\frac{\partial^2 \hat{u}}{\partial \theta^2}\right), \quad \text{in } \Omega_e. \tag{7}$$

Let

$$\hat{u}(r, \theta, s) = \sum_{n=-\infty}^{\infty} C_n(r, s)e^{in\theta}. \tag{8}$$

Inserting (8) into (7), we have

$$is\sum_{n=-\infty}^{\infty} C_n(r, s)e^{in\theta}$$
$$= -\sum_{n=-\infty}^{\infty}\left[\frac{\partial^2 C_n(r, s)}{\partial r^2} + \frac{1}{r}\frac{\partial C_n(r, s)}{\partial r} + \frac{1}{r^2}(-n^2)C_n(r, s)\right]e^{in\theta},$$

that is,

$$r^2\frac{\partial^2 C_n(r, s)}{\partial r^2} + r\frac{\partial C_n(r, s)}{\partial r} - (p^2r^2 + n^2)C_n(r, s) = 0, \tag{9}$$

where $p = \sqrt{-is}$. Let $r = \bar{r}/p$ and $C_n(r, s) = \bar{C}_n(\bar{r}, s)$, then we obtain

$$\bar{r}^2\frac{\partial^2 \bar{C}_n(\bar{r}, s)}{\partial \bar{r}^2} + \bar{r}\frac{\partial \bar{C}_n(\bar{r}, s)}{\partial \bar{r}} - (\bar{r}^2 + n^2)\bar{C}_n(\bar{r}, s) = 0. \tag{10}$$

Eq. (10) is the modified Bessel equation of order $n$, there are two linearly independent solutions $K_n(pr)$ and $I_n(pr)$, and the general solution is

$$\bar{C}_n(\bar{r}, s) = \alpha_n(s)K_n(pr) + \beta_n(s)I_n(pr),$$

where $\alpha_n(s)$ and $\beta_n(s)$ are arbitrary functions.

Under the condition (3), we have $\beta_n(s) = 0$ and $C_n(r,s) = \bar{C}_n(\bar{r}, s) = \alpha_n(s)K_n(pr)$. Inserting $C_n(r,s)$ into Eq. (8), we obtain

$$\frac{\hat{u}(r,\theta,s)}{K_0(pr)} = \alpha_0(s) + \sum_{n=\pm 1}^{\infty} \alpha_n(s)\frac{K_n(pr)}{K_0(pr)}e^{in\theta}.$$

Differentiating the above equation with respect to $r$ and $\theta$, we get

$$\frac{\partial}{\partial r}\left[\frac{\hat{u}(r,\theta,s)}{K_0(pr)}\right] = \sum_{n=\pm 1}^{\infty} \alpha_n(s)\frac{\partial}{\partial r}\left[\frac{K_n(pr)}{K_0(pr)}\right]e^{in\theta}, \tag{11}$$

$$\frac{\partial^{2k}}{\partial \theta^{2k}}\left[\frac{\hat{u}(r,\theta,s)}{K_0(pr)}\right] = \sum_{n=\pm 1}^{\infty} (in)^{2k}\alpha_n(s)\frac{K_n(pr)}{K_0(pr)}e^{in\theta}. \tag{12}$$

Following the method in [43,48], we assume that Eq. (11) is equivalent to

$$\frac{\partial}{\partial r}\left[\frac{\hat{u}(r,\theta,s)}{K_0(pr)}\right] = \sum_{k=1}^{\infty} d_k \frac{\partial^{2k}}{\partial \theta^{2k}}\left[\frac{\hat{u}(r,\theta,s)}{K_0(pr)}\right]. \tag{13}$$

Substituting (12) into (13) we get

$$\frac{\partial}{\partial r}\left[\frac{\hat{u}(r,\theta,s)}{K_0(pr)}\right] = \sum_{n=\pm 1}^{\infty}\left[\alpha_n(s)\frac{K_n(pr)}{K_0(pr)}\sum_{k=1}^{\infty}(in)^{2k}d_k\right]e^{in\theta}. \tag{14}$$

Combining Eqs. (11) and (14), we obtain

$$\sum_{k=1}^{\infty}(in)^{2k}d_k = \frac{pK_n'(pr)}{K_n(pr)} - \frac{pK_0'(pr)}{K_0(pr)}, \quad n = 1, 2, \ldots. \tag{15}$$

In practical computation, taking the first $N$ terms of Eq. (15), we get

$$\sum_{k=1}^{N}(in)^{2k}d_k = \frac{pK_n'(pr)}{K_n(pr)} - \frac{pK_0'(pr)}{K_0(pr)}, \quad n = 1, 2, \ldots, N. \tag{16}$$

We can obtain the coefficients $d_k$ by computing the following linear system:

$$\begin{pmatrix} i^2 & i^4 & \cdots & i^{2N} \\ (2i)^2 & (2i)^4 & \cdots & (2i)^{2N} \\ \vdots & \vdots & \vdots & \vdots \\ (Ni)^2 & (Ni)^4 & \cdots & (Ni)^{2N} \end{pmatrix}\begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_N \end{pmatrix} = \begin{pmatrix} \frac{pK_1'(pr)}{K_1(pr)} - \frac{pK_0'(pr)}{K_0(pr)} \\ \frac{pK_2'(pr)}{K_2(pr)} - \frac{pK_0'(pr)}{K_0(pr)} \\ \vdots \\ \frac{pK_N'(pr)}{K_N(pr)} - \frac{pK_0'(pr)}{K_0(pr)} \end{pmatrix}. \tag{17}$$

Here we list two simple examples:

for $N = 1$,

$$d_1 = \frac{pK_0'(pr)}{K_0(pr)} - \frac{pK_1'(pr)}{K_1(pr)},$$

for $N = 2$,

$$d_1 = \frac{5}{4}\frac{pK_0'(pr)}{K_0(pr)} - \frac{4}{3}\frac{pK_1'(pr)}{K_1(pr)} + \frac{1}{12}\frac{pK_2'(pr)}{K_2(pr)},$$

$$d_2 = \frac{1}{4}\frac{pK_0'(pr)}{K_0(pr)} - \frac{1}{3}\frac{pK_1'(pr)}{K_1(pr)} + \frac{1}{12}\frac{pK_2'(pr)}{K_2(pr)}.$$

With $d_0 = \frac{pK_0'(pr)}{K_0(pr)}$ and $\{d_k\}$ $(k = 1, 2, \ldots, N)$ are determined by (17), we rewrite (13) by

$$\frac{\partial \hat{u}(r,\theta,s)}{\partial r} = \sum_{k=0}^{N} d_k \frac{\partial^{2k}\hat{u}(r,\theta,s)}{\partial \theta^{2k}}. \tag{18}$$

We remark that here $d_k$ are the combination of the special functions $K_k(pr)$ and $K_0'(pr)$ with $p = \sqrt{-is}$ and $k = 0, 1, \ldots, N$. From [49], we have the following asymptotic formulas for the modified Bessel functions

$$\lim_{s\to 0} K_0(s) = \ln(s), \qquad \lim_{s\to 0} K_n(s) = \frac{\Gamma(n)}{2}\left(\frac{2}{s}\right)^n, \quad n > 0.$$

Hence, we get

$$\lim_{p\to 0}\frac{pK_0'(pr)}{K_0(pr)} = \lim_{p\to 0}\frac{-pK_1(pr)}{K_0(pr)} = \lim_{p\to 0}\frac{-p\frac{\Gamma(1)}{2}\frac{2}{pr}}{\ln(pr)}$$
$$= \lim_{p\to 0}\frac{-1}{r\ln(pr)} = 0,$$

and

$$\lim_{p\to 0}\frac{pK_n'(pr)}{K_n(pr)} = \lim_{p\to 0} p\left[\frac{n}{pr} - \frac{K_{n+1}(pr)}{K_n(pr)}\right]$$
$$= \lim_{p\to 0}\left[\frac{n}{r} - p\frac{\frac{\Gamma(n+1)}{2}\left(\frac{2}{pr}\right)^{n+1}}{\frac{\Gamma(n)}{2}\left(\frac{2}{pr}\right)^{n}}\right]$$
$$= \frac{n}{r} - \frac{2\Gamma(n+1)}{r\Gamma(n)} = -\frac{n}{r}, \quad n > 0.$$

Therefore, we obtain that the limit of $d_k$ $(k = 0, 1, 2, 3)$ is $0, \frac{37}{90}, \frac{1}{12}, \frac{1}{180}$ as $s \to 0$, $N = 3$ and $r = 3$, respectively.

When the inverse Laplace transform is applied to Eq. (18), we can get the global approximate artificial boundary condition. Unfortunately, it is difficult to implement the inverse Laplace transformation of $d_k$. In order to obtain manageable boundary conditions, an effective substitution is to approximate the coefficients $d_k$ through rational function on a finite interval $s \in [s_E, s_W]$. Provided there exists a partition $s_E = s_0 \le s_1 \le \cdots \le s_{2L} = s_W$ of the interval $[s_E, s_W]$, then for a given $r = R$, and using the rational approximation, we have

$$d_k(R,s) \approx \frac{P_L(s)}{Q_L(s)} = \frac{a_0 + a_1 s + \cdots + a_L s^L}{1 + b_1 s + \cdots + b_L s^L}, \tag{19}$$

the coefficients $a_l$ $(l = 0, \ldots, L)$, $b_l$ $(l = 1, \ldots, L)$ are calculated by solving the corresponding linear algebra problems for given $2L+1$ interpolation nodes in an interval $s \in [s_E, s_W]$. We equivalently rewrite Eq. (19) by

$$d_k(R,s) \approx \frac{P_L(s)}{Q_L(s)} = c_{k,0} + \sum_{l=1}^{L}\frac{f_{k,l}s}{s - s_{k,l}}, \tag{20}$$

where $s_{k,l}$ $(l = 1, 2, \ldots, L)$ are the roots of polynomial $1 + b_1 s + \cdots + b_L s^L$, the constants $c_{k,0}$ and $f_{k,l}$ are obtained by fraction.

Since $d_k$ involves the complex, Fig. 1 shows the approximate solutions of $d_k$ $(k = 0, 1, 2, 3)$ for imaginary and real parts and the corresponding exact solutions as $N = 3$ and $L = 5$ in the finite interval $s \in [1.0\text{e} - 10, 20.0]$ and $R = 3$, respectively. Fig. 1 illustrates that the $d_k$ can be approximated by the rational function very well. Moreover, Fig. 1 presents that the $d_k$ decay very fast as $k$ grows, which means that the influence of the coefficients $d_k$ is weaker when $k$ becomes larger for large $N$.

Table 1 lists the $\tilde{L}_1$ error of the exact solution and the approximate solution for $d_k$ $(k = 0, 1, 2, 3)$ with different $L$ as $N = 3$. The $\tilde{L}_1$ error is defined by $\tilde{L}_1 = \frac{1}{2L+1}\sum_{j=0}^{2L}\left|d_{k(ex)}(R,s_j) - d_{k(nu)}(R,s_j)\right|$,

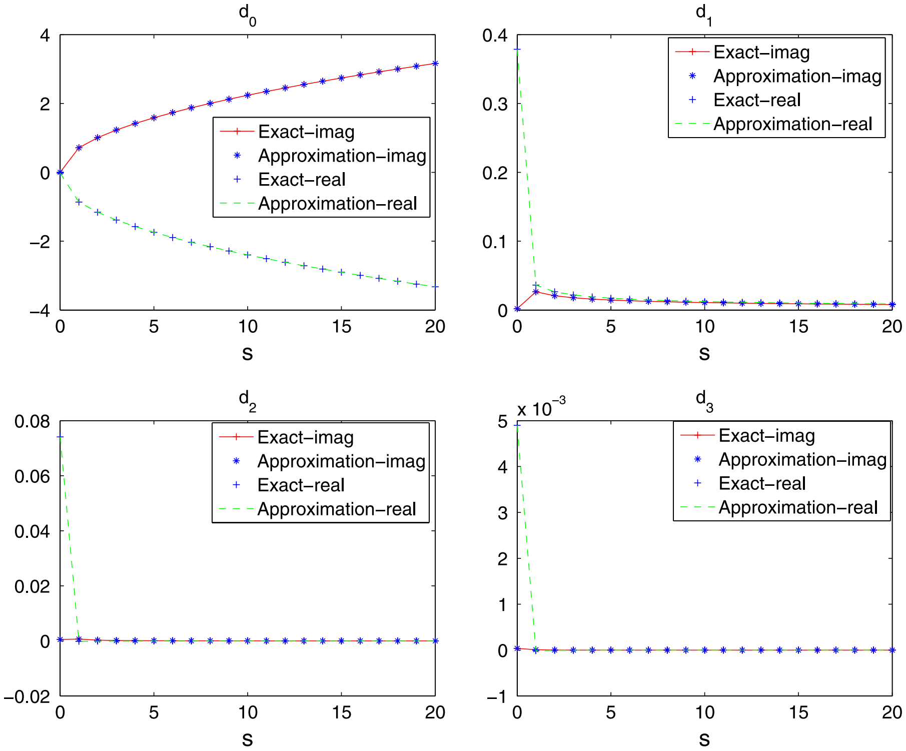

**Fig. 1.** The exact solutions and the approximate solutions of $d_k$ $(k = 0, 1, 2, 3)$ for imag and real parts when $N = 3$ and $L = 5$ in the interval [1.0e−10, 20.0].

**Table 1**
The $\tilde{L}_1$ error of the exact solution and the approximate solution for $d_k$ $(k = 0, 1, 2, 3)$ with different $L$ as $N = 3$.

| $L$ | 3 | 5 | 7 | 8 |
|---|---|---|---|---|
| $d_0$ | 8.095e−004 | 1.760e−005 | 3.327e−007 | 4.315e−008 |
| $d_1$ | 2.664e−005 | 6.103e−007 | 1.811e−008 | 3.918e−009 |
| $d_2$ | 4.025e−007 | 1.074e−008 | 6.864e−010 | 2.557e−010 |
| $d_3$ | 1.549e−008 | 3.238e−010 | 4.009e−011 | 8.240e−012 |

where $d_{k(ex)}$ and $d_{k(nu)}$ represent the exact solution and the approximate solution of $d_k$, respectively. This table shows that the error decreases as $L$ grows, and the errors are very small. We remark that we use the equal partition of the interval $[s_E, s_W]$ here and below.

Substituting Eq. (20) into (18) and constrained the results on the artificial boundary, we obtain

$$\frac{\partial \hat{u}(R,\theta,s)}{\partial r} = \sum_{k=0}^{N}\left(c_{k,0} + \sum_{l=1}^{L}\frac{f_{k,l}s}{s-s_{k,l}}\right)\frac{\partial^{2k}\hat{u}(R,\theta,s)}{\partial\theta^{2k}}. \tag{21}$$

Applying the inverse Laplace transformation directly to (21) will lead to the appearance of high-order derivatives. To render the ABCs by eliminating all the high-order derivatives, a family of auxiliary variables are introduced. Let $\hat{\varphi}_k = \frac{\partial^{2k}\hat{u}(R,\theta,s)}{\partial\theta^{2k}}$ and $-\hat{\omega}_{k,l} = \frac{s}{s-s_{k,l}}\hat{\varphi}_k$, $(k = 0,\dots,N, l = 1,\dots,L)$, then Eq. (21) can be written as

$$\begin{cases} \dfrac{\partial \hat{u}}{\partial r} = \displaystyle\sum_{k=0}^{N} c_{k,0}\hat{\varphi}_k - \sum_{k=0}^{N}\sum_{l=1}^{L} f_{k,l}\hat{\omega}_{k,l}, & \text{(a)}\\ \hat{\varphi}_0 = \hat{u}, & \text{(b)}\\ \hat{\varphi}_k = \dfrac{\partial^2\hat{\varphi}_{k-1}}{\partial\theta^2}, \quad (k = 1,\dots,N), & \text{(c)}\\ -\hat{\omega}_{k,l} = \dfrac{s}{s-s_{k,l}}\hat{\varphi}_k, \quad (k = 0,\dots,N,\ l = 1,\dots,L). & \text{(d)} \end{cases} \tag{22}$$

Applying the inverse Laplace transform to Eqs. (22)(a)–(d) and coupling them with the Schrödinger equation, we obtain the reduced initial–boundary value problem

$$\begin{cases} iu_t = -\Delta u + f, \quad \text{in } \Omega_{in} \times (0,T], & \text{(a)}\\ u|_{t=0} = u_0, \quad \text{in } \Omega_{in}, & \text{(b)}\\ \dfrac{\partial u}{\partial r} = \displaystyle\sum_{k=0}^{N} c_{k,0}\varphi_k - \sum_{k=0}^{N}\sum_{l=1}^{L} f_{k,l}\omega_{k,l}, \quad \text{on } \Gamma_R, & \text{(c)}\\ \varphi_0 = u, \quad \text{on } \Gamma_R, & \text{(d)}\\ \varphi_k = \dfrac{\partial^2\varphi_{k-1}}{\partial\theta^2}, \quad (k = 1,\dots,N), \ \text{on } \Gamma_R, & \text{(e)}\\ \partial_t\varphi_k = -\partial_t\omega_{k,l} + s_{k,l}\omega_{k,l}, & \\ \quad (k = 0,\dots,N,\ l = 1,\dots,L), \ \text{on } \Gamma_R. & \text{(f)} \end{cases} \tag{23}$$

## 2.2. Heat equation

Now we extend the above method to the heat equation. On the exterior domain the heat equation can be equally rewritten in polar coordinates $(r,\theta)$ as:

$$u_t = \frac{\partial^2 u}{\partial r^2} + \frac{1}{r}\frac{\partial u}{\partial r} + \frac{1}{r^2}\frac{\partial^2 u}{\partial\theta^2}, \quad \text{in } \Omega_e \times (0,T], \tag{24}$$

$$u|_{t=0} = 0, \quad \text{in } \Omega_e, \tag{25}$$

$$u \to 0, \quad \text{as } r \to +\infty. \tag{26}$$

Similar to the Schrödinger equation part, we get

$$\sum_{k=1}^{\infty}(in)^{2k}d_k = \frac{K_0(\sqrt{sr})}{K_n(\sqrt{sr})}\frac{\partial}{\partial r}\left[\frac{K_n(\sqrt{sr})}{K_0(\sqrt{sr})}\right] = \frac{\sqrt{s}K_n'(\sqrt{sr})}{K_n(\sqrt{sr})} - \frac{\sqrt{s}K_0'(\sqrt{sr})}{K_0(\sqrt{sr})}. \tag{27}$$

In practical computation, we replace the infinite terms by the finite terms as follows:

$$\sum_{k=1}^{N}(in)^{2k}d_k = \frac{\sqrt{s}K_n'(\sqrt{sr})}{K_n(\sqrt{sr})} - \frac{\sqrt{s}K_0'(\sqrt{sr})}{K_0(\sqrt{sr})}, \quad n = 1,2,\dots,N, \tag{28}$$

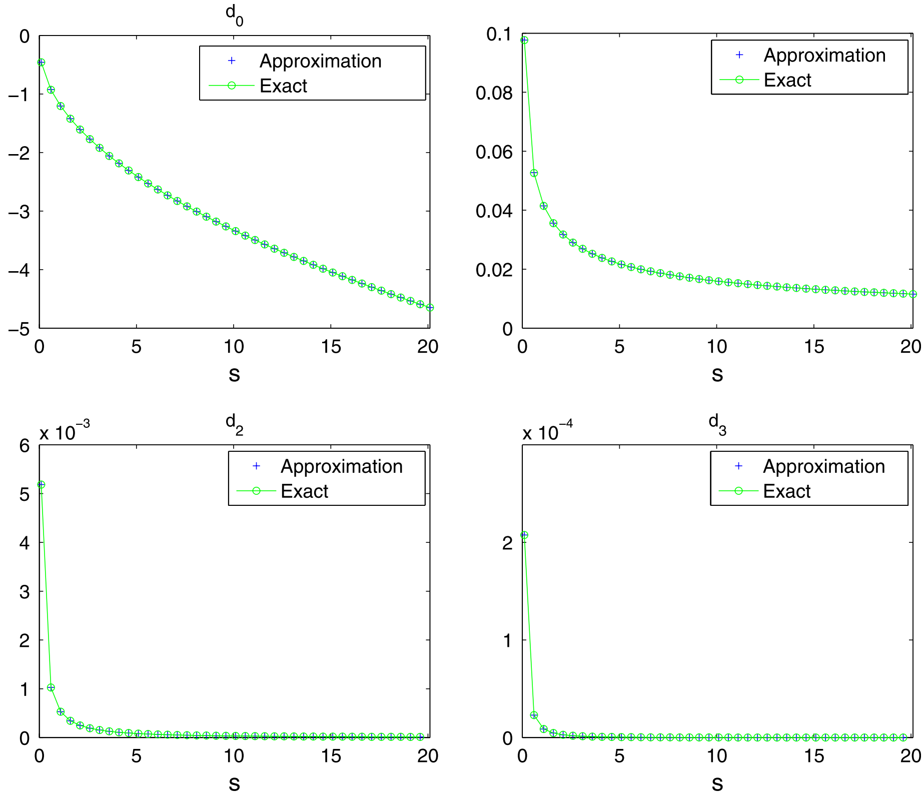

**Fig. 2.** The exact solutions and the approximate solutions of $d_k$ ($k = 0, 1, 2, 3$) when $N = 3$ and $L = 5$ in the interval $[0.1, 20.1]$.

**Table 2**
The $\tilde{L}_1$ error of the exact solution and the approximate solution for $d_k$ ($k = 0, 1, 2, 3$) with different $L$ as $N = 3$.

| $L$ | 3 | 5 | 7 | 8 |
|---|---|---|---|---|
| $d_0$ | 1.333e−003 | 6.297e−005 | 4.428e−006 | 1.275e−006 |
| $d_1$ | 5.548e−005 | 3.380e−006 | 2.563e−007 | 1.428e−007 |
| $d_2$ | 8.718e−008 | 2.424e−008 | 6.378e−009 | 3.125e−009 |
| $d_3$ | 4.946e−009 | 1.012e−009 | 1.602e−010 | 4.918e−011 |

where $N$ is a given positive integer, and the coefficients $\{d_k\}$ ($k = 1, 2, \ldots, N$) are determined by a matrix form:

$$\begin{pmatrix} i^2 & i^4 & \cdots & i^{2N} \\ (2i)^2 & (2i)^4 & \cdots & (2i)^{2N} \\ \vdots & \vdots & \vdots & \vdots \\ (Ni)^2 & (Ni)^4 & \cdots & (Ni)^{2N} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_N \end{pmatrix} = \begin{pmatrix} \frac{\sqrt{s}K_1'(\sqrt{s}r)}{K_1(\sqrt{s}r)} - \frac{\sqrt{s}K_0'(\sqrt{s}r)}{K_0(\sqrt{s}r)} \\ \frac{\sqrt{s}K_2'(\sqrt{s}r)}{K_2(\sqrt{s}r)} - \frac{\sqrt{s}K_0'(\sqrt{s}r)}{K_0(\sqrt{s}r)} \\ \vdots \\ \frac{\sqrt{s}K_N'(\sqrt{s}r)}{K_N(\sqrt{s}r)} - \frac{\sqrt{s}K_0'(\sqrt{s}r)}{K_0(\sqrt{s}r)} \end{pmatrix}. \quad (29)$$

For examples: $N = 1$

$$d_1 = \frac{\sqrt{s}K_0'(\sqrt{s}r)}{K_0(\sqrt{s}r)} - \frac{\sqrt{s}K_1'(\sqrt{s}r)}{K_1(\sqrt{s}r)},$$

and $N = 2$

$$d_1 = \frac{5}{4}\frac{\sqrt{s}K_0'(\sqrt{s}r)}{K_0(\sqrt{s}r)} - \frac{4}{3}\frac{\sqrt{s}K_1'(\sqrt{s}r)}{K_1(\sqrt{s}r)} + \frac{1}{12}\frac{\sqrt{s}K_2'(\sqrt{s}r)}{K_2(\sqrt{s}r)},$$

$$d_2 = \frac{1}{4}\frac{\sqrt{s}K_0'(\sqrt{s}r)}{K_0(\sqrt{s}r)} - \frac{1}{3}\frac{\sqrt{s}K_1'(\sqrt{s}r)}{K_1(\sqrt{s}r)} + \frac{1}{12}\frac{\sqrt{s}K_2'(\sqrt{s}r)}{K_2(\sqrt{s}r)}.$$

Thus with the finite terms, we have

$$\frac{\partial \hat{u}(r, \theta, s)}{\partial r} = \sum_{k=0}^{N} d_k \frac{\partial^{2k}\hat{u}(r, \theta, s)}{\partial \theta^{2k}}, \quad (30)$$

where $d_0 = \frac{\sqrt{s}K_0'(\sqrt{s}r)}{K_0(\sqrt{s}r)}$ and $\{d_k, k = 1, 2, \ldots, N\}$ are determined by a matrix form (29). Let $r = R$, we expand the function $d_k$ by

$$d_k(R, s) \approx \frac{P_L(s)}{Q_L(s)} = \frac{a_0 + a_1 s + \cdots + a_L s^L}{1 + b_1 s + \cdots + b_L s^L}, \quad s \in \{s_0, \ldots, s_{2L}\}. (31)$$

Eq. (31) is equivalent to

$$d_k(R, s) \approx \frac{P_L(s)}{Q_L(s)} = c_{k,0} + \sum_{l=1}^{L} \frac{f_{k,l}s}{s - s_{k,l}}, \quad s \in \{s_0, \ldots, s_{2L}\}. \quad (32)$$

The coefficients $s_{k,l}$ ($l = 1, \ldots, L$) are roots of polynomial $1 + b_1 s + \cdots + b_L s^L$, $c_{k,0}$ and $f_{k,l}$ ($l = 1, \ldots, L$) are obtained by solving the linear algebra problem $P_L(s) = Q_L(s)(c_{k,0} + \sum_{l=1}^{L} \frac{f_{k,l}s}{s - s_{k,l}})$.

In a finite interval $s \in [0.1, 20.1]$ and $R = 3$, Fig. 2 presents the approximate solutions of $d_k$ ($k = 0, 1, 2, 3$) and the corresponding exact solutions for $N = 3$ and $L = 5$, respectively. Fig. 2 shows that the rational functions $\frac{P_L(s)}{Q_L(s)}$ fit $d_k$ perfectly. Table 2 lists the $\tilde{L}_1$ error of the exact solution and the approximate solution for $d_k$ ($k = 0, 1, 2, 3$) with different $L$ as $N = 3$. Furthermore, the coefficients $d_k$ decay very fast as $k$ grows from Fig. 2. That is to say, the first few terms of $d_k$ are dominant in Eq. (30).

We substitute the approximation (31) into (30) on the artificial boundary $r = R$, and obtain

$$\frac{\partial \hat{u}(R, \theta, s)}{\partial r} = \sum_{k=0}^{N} \left( c_{k,0} + \sum_{l=1}^{L} \frac{f_{k,l}s}{s - s_{k,l}} \right) \frac{\partial^{2k}\hat{u}(R, \theta, s)}{\partial \theta^{2k}}. \quad (33)$$

Let $\hat{\varphi}_k = \frac{\partial^{2k}\hat{u}(R,\theta,s)}{\partial\theta^{2k}}$ and $-\hat{\omega}_{k,l} = \frac{s}{s-s_{k,l}}\hat{\varphi}_k$, $(k = 0,\ldots,N, l = 1,\ldots,L)$. Similarly to the Schödinger equation, we have the reduced initial–boundary value problem for the heat equation by

$$\begin{cases} u_t = \Delta u + f, \quad \text{in } \Omega_{in} \times (0,T], & \text{(a)} \\ u|_{t=0} = u_0, \quad \text{in } \Omega_{in}, & \text{(b)} \\ \dfrac{\partial u}{\partial r} = \displaystyle\sum_{k=0}^{N} c_{k,0}\varphi_k - \sum_{k=0}^{N}\sum_{l=1}^{L} f_{k,l}\omega_{k,l}, \quad \text{on } \Gamma_R, & \text{(c)} \\ \varphi_0 = u, \quad \text{on } \Gamma_R, & \text{(d)} \\ \varphi_k = \dfrac{\partial^2\varphi_{k-1}}{\partial\theta^2}, \quad (k = 1,\ldots,N), \text{ on } \Gamma_R, & \text{(e)} \\ \partial_t\varphi_k = -\partial_t\omega_{k,l} + s_{k,l}\omega_{k,l}, & \\ \quad (k = 0,\ldots,N,\ l = 1,\ldots,L), \text{ on } \Gamma_R. & \text{(f)} \end{cases} \tag{34}$$

## 3. Numerical discretization

In this section we are devoted to designing the corresponding discretization of the reduced initial–boundary-value problem with the high-order artificial boundary conditions. For the sake of simplicity, we only give the discretized finite difference schemes for the Schrödinger equation, the discretization for the reduced heat equation can be obtained similarly. Due to the coordinate transformation, there will appear a non-essential singularity at point (0, 0) for polar coordinate. We can overcome this shortcoming by the method proposed by Zhang et al. [50].

First we cover the truncated computational domain $[0,R] \times [0,2\pi]$ by a uniform grid, parallel to the axes of the polar coordinate system. For given positive integers $I, J$ and $M$, we have the spatial and temporal mesh sizes by defining $\Delta r = R/I$, $\Delta\theta = 2\pi/J$ and $\Delta t = T/M$, respectively. Thus the grid point can be written as $r_\zeta = \zeta\Delta r$, $\theta_j = j\Delta\theta$ and $t_m = m\Delta t$, correspondingly. Denote the operators $D_+$, $D_-$ and $D_0$ by forward, backward and centered differences, respectively, $S_+$ represents forward sums. Take an example, $D_+^t u_{\zeta,j}^m = \frac{u_{\zeta,j}^{m+1} - u_{\zeta,j}^m}{\Delta t}$, $S_+^t u_{\zeta,j}^m = \frac{u_{\zeta,j}^{m+1} + u_{\zeta,j}^m}{2}$, where $u_{\zeta,j}^m$ presents the approximation of $u(r_\zeta, \theta_j, t_m)$.

We consider the Crank–Nicolson finite difference scheme for the Schrödinger equation in the interior domain, which is one of the commonly used discretization methods for the partial differential equations.

$$iD_+^t u_{\zeta,j}^m = -\left(D_+^r D_-^r S_+^t + \frac{1}{r_\zeta} D_0^r S_+^t + \frac{1}{r_\zeta^2} D_+^\theta D_-^\theta S_+^t\right) u_{\zeta,j}^m + f(r_\zeta, \theta_j, t_{m+\frac{1}{2}}), \tag{35}$$

$$u_{\zeta,0}^m = u_{\zeta,J}^m, \qquad u_{\zeta,-1}^m = u_{\zeta,J-1}^m, \tag{36}$$

with $0 < \zeta < I$, $0 \le j < J$, $0 \le m \le M$.

Since there exists the singularity of the coefficient of the equation at $r = 0$, we cannot discretize Eq. (23)(a) directly. In order to avoid the non-essential singularity at point (0, 0), we refer to the method proposed by Zhang et al. [50]. Selecting a small disk $\Omega_{\frac{\Delta r}{2}} := \{(r,\theta) | 0 \le r \le \frac{\Delta r}{2},\ \ 0 \le \theta < 2\pi\}$, and taking integration by parts to Eq. (23)(a) in this small region, we arrive at

$$-\frac{\Delta r}{2}\int_0^{2\pi} \frac{\partial u}{\partial r} d\theta = \int_{\Omega_{\frac{\Delta r}{2}}} (iu_t + f) dxdy.$$

By using the mid-point rules, we have

$$-\frac{\Delta r \Delta\theta}{2}\sum_{j=0}^{J} \left.\frac{\partial u}{\partial r}\right|_{(\frac{\Delta r}{2},\theta_j)} = \frac{\pi\Delta r^2}{4}(iu_t(0,0) + f).$$

We use the approximation $\frac{\partial u}{\partial r}|_{(\frac{\Delta r}{2},\theta_j)} = [u(\Delta r, \theta_j) - u(0,0)]/\Delta r$, and obtain

$$-\frac{\Delta\theta}{2}\sum_{j=0}^{J}\left[u(\Delta r,\theta_j) - u(0,0)\right] = \frac{\pi\Delta r^2}{4}(iu_t(0,0) + f).$$

Thus with the forward sum the discretization of Eq. (23)(a) at point (0, 0) is obtained as

$$-\frac{2\Delta\theta}{\pi\Delta r^2}\left[\sum_{j=0}^{J} S_+^t u_{1,j}^m - (J+1) S_+^t u_{0,0}^m\right] = iD_+^t u_{0,0}^m + f(0,0,t_{m+\frac{1}{2}}). \tag{37}$$

Now the artificial boundary conditions (23)(a)–(d) are remained to be discretized. For the given $N$ and $L$, the coefficients $c_{k,0}, f_{k,l}$ and $s_{k,l}$ can be obtained from Eq. (19). Then the boundary conditions are approximated by:

$$D_+^r S_+^t u_{I-1,j}^m = \sum_{k=0}^{N} c_{k,0} S_+^t \varphi_{k,j}^m - \sum_{k=0}^{N}\sum_{l=1}^{L} f_{k,l} S_+^t \omega_{k,l}^m, \tag{38}$$

$$S_+^t \varphi_{k,j}^m = D_+^\theta D_-^\theta S_+^t \varphi_{k-1,j}^m, \tag{39}$$

$$D_+^t \varphi_{k,j}^m = -D_+^t \omega_{k,l}^m + s_{k,l} S_+^t \omega_{k,l}^m. \tag{40}$$

Note that

$$\varphi_{0,j}^m = u_{I-1,j}^m, \qquad u_{0,j}^m = u_{0,0}^m, \quad (j = 1,\ldots,J, m = 0,1,\ldots,M). \tag{41}$$

Therefore, the discretizations of Eq. (23)(a)–(d) are described by (35)–(41). The above method is easy to extend to the heat equation.

Finally, we summarized the local absorbing boundary conditions by the following algorithm:

1. For fixed $N$, calculate the $d_k$ $(k = 0,1,\ldots,N)$ with formulas (17).
2. For fixed $L$, calculate the corresponding coefficients $c_{k,0}, f_{k,l}$ and $s_{k,l}$ via Eq. (19) for each $d_k$ $(k = 0,1,\ldots,N)$.
3. The coefficients $c_{k,0}, f_{k,l}$ and $s_{k,l}$ are applied for the calculation of (35)–(41) for each $m$ $(m = 0,1,\ldots,M)$.

## 4. Numerical results

In order to demonstrate the effectiveness of the obtained artificial boundary conditions, two numerical examples for the heat equation and three examples for the Schrödinger equation are discussed in this section, respectively. In the calculations, we set the source term $f(\mathbf{x},t) = 0$.

### 4.1. Numerical tests for the heat equation

**Example 1.** We consider the initial function for the heat equation

$$u_0(r,\theta) = \frac{1}{4\pi t_0}\exp\left(-\frac{r^2}{4t_0}\right),$$

for which the exact solution is given by

$$u_{ex}(r,\theta,t) = \frac{1}{4\pi(t+t_0)}\exp\left[-\frac{r^2}{4(t+t_0)}\right].$$

In the calculation, let $t_0 = 0.1$ and $T = 0.8$.

Generally, in order to obtain more accurate numerical solution, the computational domain should be larger, and the cost of computation also will be expensive. We first study the dependence of the computational domain, and choose $R = 3.0, 3.5, 4.0, 4.5, 5.0, 7.0,$

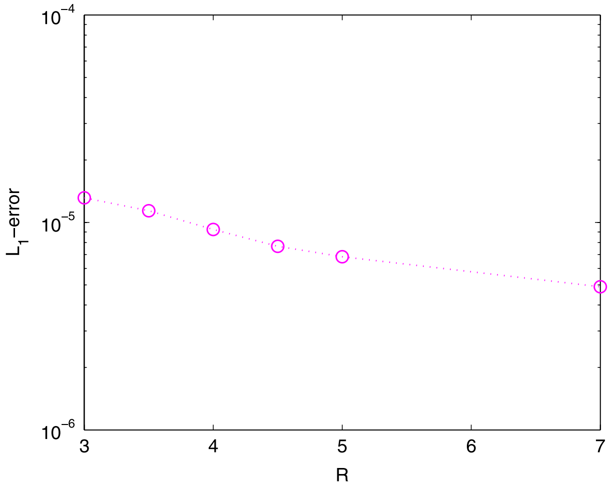

**Fig. 3.** $L_1$-error for different $R$ when fixed $N = 1$ and $L = 4$.

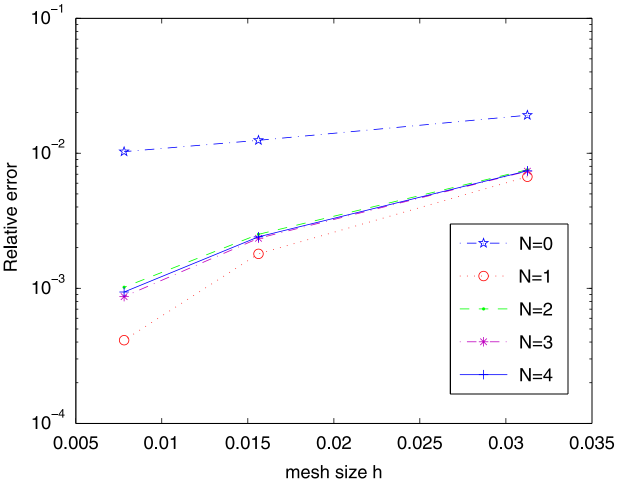

**Fig. 4.** Relative error for different $N$ when $L = 5$.

**Table 3**
Relative error for different $L$ when $N = 1$.

| $L$ | 3 | 4 | 5 | 6 |
|---|---|---|---|---|
| 32 × 32 | 2.739e−003 | 2.739e−003 | 2.739e−003 | 2.739e−003 |
| 64 × 64 | 7.209e−004 | 7.209e−004 | 7.209e−004 | 7.209e−004 |
| 128 × 128 | 1.850e−004 | 1.851e−004 | 1.851e−004 | 1.851e−004 |

respectively. The corresponding mesh sizes are taken by $\Delta r = \frac{1}{32}$, $\Delta\theta = \frac{2\pi}{64}$ and $\Delta t = 3.0\text{e} - 3$. The relative errors are defined by $\frac{\max_{\zeta,j}|u_{ex}(r_\zeta,\theta_j,t)-u_{nu}(r_\zeta,\theta_j,t)|}{\max_{\zeta,j}|u_{ex}(r_\zeta,\theta_j,t)|}$, where $u_{ex}$ and $u_{nu}$ denote the exact solution and the numerical solution, respectively. Despite the relative error plays an important role in investigating the efficiency of our method, sometimes the $L_1$-error is another important way to measure the performance of ABCs, and is defined by

$$L_1(t) = \frac{1}{I \times J}\sum_{\zeta=0}^{I}\sum_{j=0}^{J}\left|u_{ex}(r_\zeta,\theta_j,t) - u_{nu}(r_\zeta,\theta_j,t)\right|.$$

Fig. 3 shows the $L_1$-errors for different $R$ when $N = 1$ and $L = 4$ and illustrates that the errors decrease slowly as the $R$ increases. Our method can achieve a high accuracy even the computational domain is small. Thus we set the radius $R = 3.0$ in the following simulations.

Since the solution of this example is symmetric, the parameter $N$ has little influence on the efficiency of the obtained artificial boundary conditions. Hence we can test the dependence of the parameter $L$ on the efficiency of the obtained boundary conditions. For different refinement meshes, Table 3 lists the relative errors. Table 4 shows the $L_1$-errors and convergence rates with different parameter $L$ and mesh sizes. From Tables 3 and 4, one can see that the convergence order is almost second order when $L = 4$.

**Example 2.** In order to test the dependence of the parameter $N$, we design the asymmetric solution as

$$u_{ex}(r,\theta,t) = \frac{1}{4\pi(t+t_0)}\exp\left[-\frac{r^2-2r\cos\theta+1}{4(t+t_0)}\right].$$

In practical computation, let $t = 0$, $t_0 = 0.1$ be the initial function and $T = 0.8$.

Fig. 4 and Table 5 show the relative error and the $L_1$-error for different $N$ and fixed $L = 5$, respectively. We can observe that the convergence order is close to 2 when $N = 1$ and $L = 5$.

Fig. 5 shows the $L_1$-errors in space as a function of time for various $N$ with the fixed mesh 64 × 64 and $L = 4$. The $L_1$-error decreases with the parameter $N$ increases. Overall, a good performance is obtained when $N = 1$ and $L = 4$, though certainly all parameter choices led to excellent accuracy at low cost.

### 4.2. Numerical tests for the Schrödinger equation

**Example 3.** In order to investigate the dependence of ABCs for the Schrödinger equation on the choice of the parameters $L$ and $N$, we

**Table 4**
$L_1$-error and convergence order for different $L$ when $N = 1$.

| $L$ | 32 × 32 | | 64 × 64 | | 128 × 128 | |
|---|---|---|---|---|---|---|
| | $L_1$-error | Order | $L_1$-error | Order | $L_1$-error | Order |
| 3 | 1.035e−004 | – | 2.757e−005 | 1.91 | 7.500e−006 | 1.88 |
| 4 | 1.029e−004 | – | 2.711e−005 | 1.92 | 7.095e−006 | 1.93 |
| 5 | 1.029e−004 | – | 2.715e−005 | 1.92 | 7.134e−006 | 1.93 |
| 6 | 1.029e−004 | – | 2.715e−005 | 1.92 | 7.130e−006 | 1.93 |

**Table 5**
$L_1$-error and convergence order for different $N$ when $L = 5$.

| $L = 5$ | 32 × 32 | | 64 × 64 | | 128 × 128 | |
|---|---|---|---|---|---|---|
| | $L_1$-error | Order | $L_1$-error | Order | $L_1$-error | Order |
| $N = 0$ | 1.831e−004 | – | 7.553e−005 | 1.28 | 4.776e−005 | 0.66 |
| $N = 1$ | 1.287e−004 | – | 3.111e−005 | 2.05 | 7.922e−006 | 1.97 |
| $N = 2$ | 1.302e−004 | – | 3.189e−005 | 2.03 | 8.133e−006 | 1.97 |
| $N = 3$ | 1.301e−004 | – | 3.189e−005 | 2.03 | 8.162e−006 | 1.97 |
| $N = 4$ | 1.301e−004 | – | 3.188e−005 | 2.03 | 8.152e−006 | 1.97 |

**Table 6**
$L_1$-error and convergence order for different $L$ when $N = 1$.

| $L$ | $32 \times 32$ | | $64 \times 64$ | | $128 \times 128$ | |
|---|---|---|---|---|---|---|
| | $L_1$-error | Order | $L_1$-error | Order | $L_1$-error | Order |
| 5 | 7.910e−004 | – | 2.022e−004 | 1.97 | 6.332e−005 | 1.68 |
| 6 | 7.764e−004 | – | 1.859e−004 | 2.06 | 4.447e−005 | 2.06 |
| 7 | 7.776e−004 | – | 1.871e−004 | 2.06 | 4.454e−005 | 2.07 |
| 8 | 7.785e−004 | – | 1.880e−004 | 2.05 | 4.552e−005 | 2.05 |

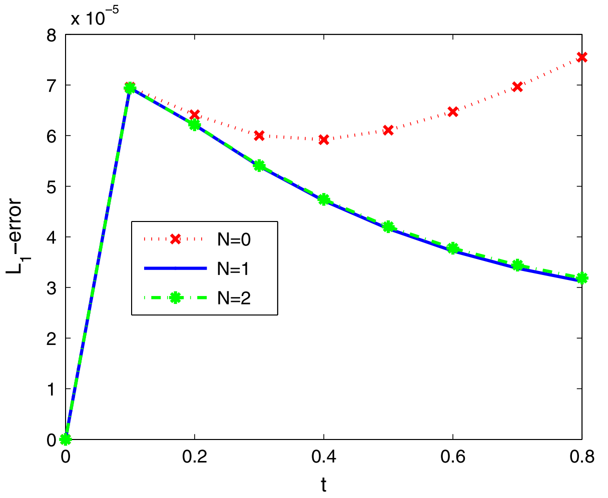

**Fig. 5.** The evolution of $L_1$-error between the numerical solutions and the exact solutions for different $N$ at mesh $64 \times 64$ and $L = 4$.

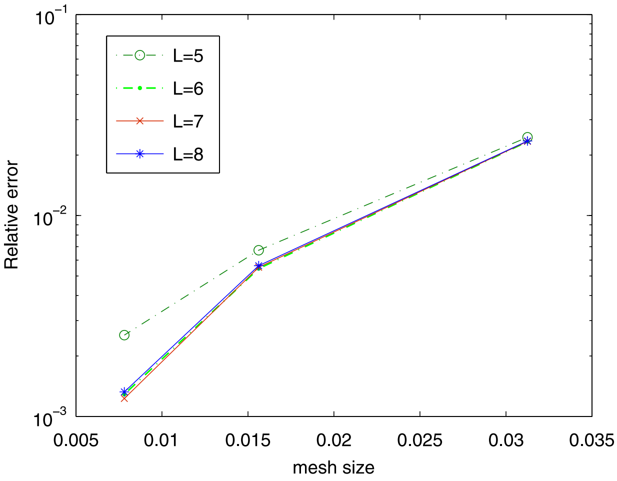

**Fig. 6.** Relative error for different $L$ when $N = 1$.

first consider the Schrödinger equation with the symmetric initial function

$$u_0(r, \theta) = \frac{1}{4\pi t_0} \exp\left(-\frac{r^2}{4t_0}\right),$$

and the exact solution is given by

$$u_{ex}(r, \theta, t) = \frac{1}{4\pi(it + t_0)} \exp\left[-\frac{r^2}{4(it + t_0)}\right]$$

with $t_0 = 0.1$ and $T = 0.6$.

The accuracy of local ABCs mainly depends on the choice of the parameter $L$ owing to this example that is symmetric. Table 6 and Fig. 6 list the $L_1$-errors and relative errors with different parameter $L$ and mesh sizes, respectively. The convergence rate reaches to the second order when $L \geq 6$.

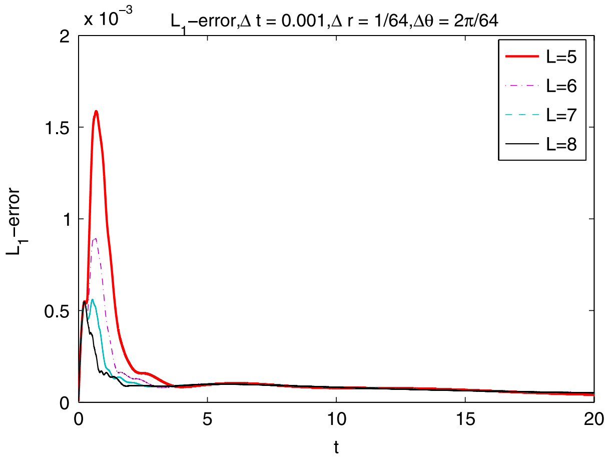

**Fig. 7.** $L_1$-error for the long time for different $L$ when $N = 2$.

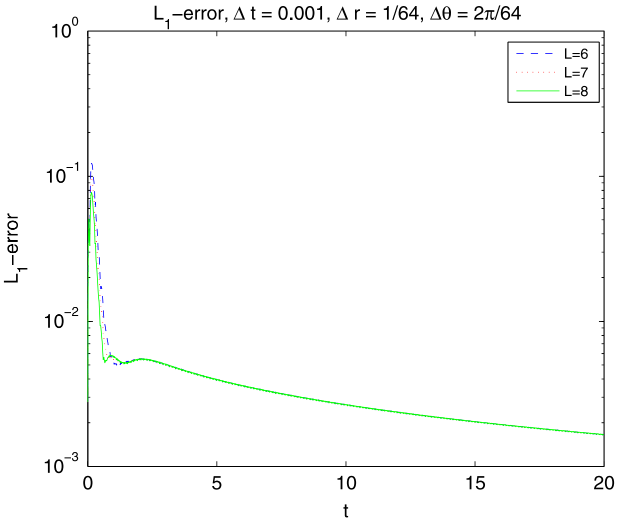

**Fig. 8.** $L_1$-error for the long time $t = 20$ and the amplitude drops more than 10 for different $L$ when $N = 2$.

The long time behavior of the $L_1$-error due to our local high-order ABCs is also considered. We choose the discretization parameters $\Delta t = 0.001$, $\Delta r = 1/64$ and $\Delta\theta = 2\pi/64$. For $t_0 = 0.1$, Fig. 7 shows the $L_1$-error for the long time for different $L$ when $N = 2$. Fig. 8 shows the $L_1$-error for the long time and the amplitude drops more than 10 ($t_0 = 0.007$) for different $L$ when $N = 2$. Long time calculations with local ABCs are stable from Figs. 7–8.

**Example 4.** To demonstrate the dependence of our proposed boundary conditions on the parameter $N$, we consider the Schrödinger equation with the initial function

$$u_0(r, \theta) = \frac{1}{4\pi t_0} \exp\left(-\frac{r^2 - 2r\cos\theta + 1}{4t_0}\right),$$

**Table 7**
$L_1$-error and convergence order for different $L$ when $N = 1$.

| $L$ | $32 \times 32$ | | $64 \times 64$ | | $128 \times 128$ | |
|---|---|---|---|---|---|---|
| | $L_1$-error | Order | $L_1$-error | Order | $L_1$-error | Order |
| 5 | 1.300e−003 | – | 3.266e−004 | 1.99 | 1.010e−004 | 1.69 |
| 6 | 1.298e−003 | – | 3.233e−004 | 2.01 | 9.457e−005 | 1.77 |
| 7 | 1.297e−003 | – | 3.219e−004 | 2.01 | 9.258e−005 | 1.80 |
| 8 | 1.297e−003 | – | 3.219e−004 | 2.01 | 9.238e−005 | 1.80 |

**Table 8**
$L_1$-error and convergence order for different $L$ when $N = 2$.

| $L$ | $32 \times 32$ | | $64 \times 64$ | | $128 \times 128$ | |
|---|---|---|---|---|---|---|
| | $L_1$-error | Order | $L_1$-error | Order | $L_1$-error | Order |
| 5 | 1.300e−003 | – | 3.245e−004 | 2.00 | 8.885e−005 | 1.87 |
| 6 | 1.298e−003 | – | 3.210e−004 | 2.02 | 8.295e−005 | 1.95 |
| 7 | 1.297e−003 | – | 3.197e−004 | 2.02 | 8.104e−005 | 1.98 |
| 8 | 1.297e−003 | – | 3.197e−004 | 2.02 | 8.087e−005 | 1.98 |

**Table 9**
$L_1$-error and convergence order for different $L$ when $N = 3$.

| $L$ | $32 \times 32$ | | $64 \times 64$ | | $128 \times 128$ | |
|---|---|---|---|---|---|---|
| | $L_1$-error | Order | $L_1$-error | Order | $L_1$-error | Order |
| 5 | 1.300e−003 | – | 3.243e−004 | 2.00 | 8.868e−005 | 1.87 |
| 6 | 1.297e−003 | – | 3.209e−004 | 2.01 | 8.281e−005 | 1.95 |
| 7 | 1.297e−003 | – | 3.196e−004 | 2.02 | 8.093e−005 | 1.98 |
| 8 | 1.297e−003 | – | 3.196e−004 | 2.02 | 8.076e−005 | 1.98 |

**Table 10**
$L_1$-error for $L = 7$ when $N = 3$ and the mesh $128 \times 128$ in different interval.

| | $s \in [1.0e - 1, 20.1]$ | $s \in [1.0e - 2, 20]$ | $s \in [1.0e - 5, 20]$ | $s \in [1.0e - 10, 20]$ | $s \in [0, 20]$ |
|---|---|---|---|---|---|
| $L_1$-error | 8.093e−005 | 8.074e−005 | 8.074e−005 | 8.074e−005 | 8.074e−005 |

**Table 11**
$L_1$-error and convergence order for different $N$ when $L = 8$ in the interval $s \in [1.0e - 10, 20]$.

| $N$ | $32 \times 32$ | | $64 \times 64$ | | $128 \times 128$ | |
|---|---|---|---|---|---|---|
| | $L_1$-error | Order | $L_1$-error | Order | $L_1$-error | Order |
| 2 | 7.744e−003 | – | 1.591e−003 | 2.28 | 4.117e−004 | 1.95 |
| 3 | 7.741e−003 | – | 1.587e−003 | 2.28 | 4.080e−004 | 1.96 |

and its exact solution is asymmetric given by

$$u_{ex}(r, \theta, t) = \frac{1}{4\pi(it + t_0)} \exp\left[-\frac{r^2 - 2r\cos\theta + 1}{4(it + t_0)}\right]$$

with $t_0 = 0.1$ and $T = 0.6$.

Tables 7–9 list the $L_1$-error with different $L$ and mesh sizes when $N = 1, 2, 3$, respectively. From Tables 7–9, the convergence order is nearly 2 when $N = 2$ and $L = 7$.

Table 10 lists the $L_1$-error for $L = 7$ when $N = 3$ and the mesh $128 \times 128$ in different interval. Since $d_k$ $(k = 0, 1, 2, 3)$ is not defined at $s = 0$, we take the value of $d_k$ $(k = 0, 1, 2, 3)$ as $\lim_{s \to 0} d_k$ $(k = 0, 1, 2, 3)$. We can observe that the choice of the interval $s$ is not very sensitive for the accuracy of local ABCs.

**Example 5.** In order to illustrate the accuracy of local ABCs for higher azimuth harmonics in the solution, we consider the Schrödinger equation with the initial function

$$u_0(r, \theta) = \frac{1}{4\pi t_0} \exp\left[-\frac{r^2 - 2r\cos(4\theta) + 1}{4t_0}\right],$$

with $t_0 = 0.1$ and $T = 0.4$. The exact solution of this problem is not known. We take $R = 9$, and consider the corresponding solution as the exact solution. The computational domain is the same as above, i.e., $R = 3$.

Table 11 lists the $L_1$-error and convergence order for different $N$ when $L = 8$ in the interval $s \in [1.0e - 10, 20]$. Again, we see that the performance of local ABCs for higher azimuth harmonics is good from this example.

## 5. Conclusion

This paper studied local high-order ABCs for Schrödinger and heat equations using a circular artificial boundary. By using the finite difference formulation, various numerical examples demonstrated the accuracy and efficiency of our method. From numerical simulation, we found that the artificial boundary conditions work very well. However, there are some theoretical issues of these problems that remain to be answered, including a rigorous stability analysis of the reduced initial–boundary value problem with our proposed local high-order ABCs on the bounded computational domain. Also, the ratio $T/R$ is a very important parameter in test calculations of wave problems, where $T$ is the modeling time and $R$ is the radius of the artificial boundary. The case of small value $T/R = 0.2$ has been studied; the cases of mid and big values $(T/R = 1, T/R = 10, \ldots)$ require a separate study. We intend to address these questions in our future study.

## Acknowledgments

The authors gratefully acknowledge the two anonymous referees for their careful reading and many constructive suggestions which lead to a great improvement of the paper. This research is supported by FRG of Hong Kong Baptist University, (Grant No. FRG1/11-12/051) and National Natural Science Foundation of China (Grant Nos. 11326227 and 11301310).

## References

[1] D. Givoli, in: R.W. Lewis, K. Morgen (Eds.), In Numerical Methods in Thermal Problems, Vol. VI, Swansea, UK, 1989, pp. 1094–1104.
[2] L. Greengard, P. Lin, On the numerical solution of the heat equation in unbounded domains (Part I), in: Courant Mathematics and Computing Laboratory, in: Tech. Note, vol. 98002, New York University, 1998.
[3] H. Han, Z. Huang, Comput. Math. Appl. 44 (2002) 655.
[4] H. Han, Z. Huang, Comput. Math. Appl. 43 (2002) 889.
[5] X. Wu, Z. Sun, Appl. Numer. Math. 50 (2004) 261.
[6] X. Wu, J. Zhang, J. Comput. Math. 29 (2011) 74.
[7] H. Han, Z. Huang, Commun. Math. Sci. 2 (1) (2004) 79.
[8] V. Baskakov, A. Popov, Wave Motion 14 (1991) 123.
[9] T. Fevens, H. Jiang, SIAM J. Sci. Comput. 21 (1) (1999) 255.
[10] X. Antoine, C. Besse, V. Mouysset, Math. Comp. 73 (2004) 1779.
[11] A. Arnold, M. Ehrhardt, M. Schulte, I. Sofronov, Commun. Math. Sci. 10 (3) (2012) 889.
[12] A. Schädle, Wave Motion 35 (2001) 181.
[13] X. Antoine, C. Besse, P. Klein, SIAM J. Sci. Comput. 33 (2) (2011) 1008.
[14] X. Antoine, C. Besse, P. Klein, M3AS 22 (10) (2012) 1250026.
[15] X. Antoine, C. Besse, P. Klein, Numer. Math. 125 (2) (2013) 191.
[16] A. Arnold, M. Ehrhardt, I. Sofronov, Commun. Math. Sci. 1 (3) (2003) 501.
[17] J. Zhang, Z.Z. Sun, X. Wu, D. Wang, Commun. Comput. Phys. 10 (3) (2011) 742.
[18] B. Engquist, A. Majda, Math. Comp. 31 (1977) 629.
[19] R. Higdon, Math. Comp. 47 (1986) 437.
[20] H. Han, X. Wu, J. Comput. Math. 3 (1985) 179.
[21] D. Yu, J. Comput. Math. 3 (1985) 219.
[22] L. Halpern, J. Rauch, Numer. Math. 71 (1995) 185.
[23] M.N. Guddati, J.L. Tassoulas, J. Comput. Acoust. 8 (2000) 139.
[24] X. Antoine, A. Arnold, C. Besse, M. Ehrhardt, A. Schädle, Commun. Comput. Phys. 4 (4) (2008) 729.
[25] D. Givoli, Numerical Methods for Problems in Infinite Domains, Elsevier, Amsterdam, 1992.
[26] S.V. Tsynkov, Appl. Numer. Math. 27 (1998) 465.
[27] H. Han, The artificial boundary method-numerical solutions of partial differential equations in unbounded domains, in: T. Li, P. Zhang (Eds.), Frontiers and Prospents of Contemporary Applied Mathematics, Higher Education Press and World Scientific, 2006, pp. 33–66.
[28] D. Givoli, Wave Motion 39 (2004) 319.
[29] T. Hagstrom, T. Warburton, Wave Motion 39 (2004) 327.
[30] T. Hagstrom, M. de Castro, D. Givoli, D. Tsemach, J. Comput. Acoust. 15 (2007) 1.
[31] T. Hagstrom, A. Mar-Or, D. Givoli, J. Comput. Phys. 227 (2008) 3322.
[32] Z.Z. Sun, X. Wu, J. Comput. Phys. 214 (2006) 209.
[33] X. Antoine, C. Besse, J. Comput. Phys. 188 (2003) 157.
[34] A.Y. Suhov, A. Ditkowski, SIAM J. Sci. Comput. 33 (4) (2011) 1765.
[35] Z. Xu, H. Han, X. Wu, J. Comput. Phys. 225 (2007) 1577.
[36] S. Jiang, L. Greengard, Comm. Pure Appl. Math. 61 (2007) 261.
[37] W. Bao, H. Han, Numer. Math. 93 (2003) 415.
[38] L. Di Menza, Numer. Funct. Anal. Optim. 18 (1997) 759.
[39] E. Bécache, D. Givoli, T. Hagstrom, J. Comput. Phys. 229 (2010) 1099.
[40] D. Givoli, B. Neta, J. Comput. Phys. 186 (2003) 24.
[41] J. Zhang, Z. Xu, X. Wu, Phys. Rev. E 78 (2008) 026709.
[42] J. Zhang, Z. Xu, X. Wu, Phys. Rev. E 79 (2009) 046711.
[43] H. Han, X. Wu, Artificial Boundary Method, Springer-Verlag, Berlin, Heidelberg, 2013, Tsinghua University Press, Beijing.
[44] G. Lancioni, Comput. Methods Appl. Mech. Engrg. 209–212 (2012) 74.
[45] D. Elliott, Math. Comp. 21 (1967) 398.
[46] L. Wuytack, Padé Approximation and its Applications, Springer, Berlin, 1979.
[47] J.C. Butcher, Appl. Numer. Math. 59 (2009) 558.
[48] W. Bao, H. Han, Comput. Methods Appl. Mech. Engrg. 188 (2000) 455.
[49] L. Andrews, Special Functions of Mathematics for Engineers, McGraw-Hill, Singapore, 1992.
[50] J. Zhang, H. Han, H. Brunner, J. Sci. Comput. 49 (2011) 367.