Computer Physics Communications 143 (2002) 113–135

Computer Physics Communications

www.elsevier.com/locate/cpc

Comparison of the discrete singular convolution algorithm and the Fourier pseudospectral method for solving partial differential equations

S.Y. Yang, Y.C. Zhou, G.W. Wei *

Department of Computational Science, National University of Singapore, 117543 Singapore

Received 28 May 2001; received in revised form 10 September 2001

Abstract

This paper explores the utility, tests the accuracy and examines the limitation of the discrete singular convolution (DSC) algorithm for solving partial differential equations (PDEs). The standard Fourier pseudospectral (FPS) method is also implemented for a detailed comparison so that the performance of the DSC algorithm can be better evaluated. Three two-dimensional PDEs of different nature, the heat equation, the wave equation and the Navier–Stokes equation, are employed to make our assessment. Either the fourth-order Runge–Kutta or the Crank–Nicolson scheme is employed for the temporal discretization. The DSC algorithm is projected into the Fourier domain for analyzing its numerical resolution. It is demonstrated that the accuracy of the DSC algorithm is controllable. Comprehensive comparisons are given based on a variety of time increment, grid spacing, wavenumber, and Reynolds number. It is found that the DSC algorithm is an accurate, stable and robust approach for solving these PDEs.

1. Introduction

There has been an ongoing interest in computational methodology in the past few decades, due to the availability of inexpensive high-performance computers, which has given tremendous impetus to scientific and engineering computing. Most effort has been focused on developing either global methods [1–7] or local methods [8–17] for solving a variety of partial differential equations (PDEs). There has been much debate among the numerical computation communities over the advantages and disadvantages of various numerical methods over the past few decades. In general, the local methods, such as methods of finite differences, finite volumes and finite elements, are highly localized in the spatial domain, yet delocalized in their spectral domain. In contrast, global methods, such as the Fourier spectral method, are highly localized in their spectral representations and delocalized in the spatial domain. As a consequence, global methods appear to be more accurate than local methods. The main advantage of local methods is their flexibility for satisfying special boundary conditions and for treating complex geometries. For example, methods of finite elements are the dominant approach in

* Corresponding author.
E-mail address: cscweigw@nus.edu.sg (G.W. Wei).

0010-4655/02/$ – see front matter
PII: S0010-4655(01)00427-1

structural analysis in association with unstructured grids. Finite volume methods are commonly used in fluid flow simulations. Finite difference methods are the major approach for electromagnetic wave propagations. Nevertheless, spectral methods have had tremendous success in several areas, where a high computational accuracy is of particular importance while the geometry of the underlying problem is relatively simple. Typical examples of these areas include weather prediction, nonlinear waves, simulation of turbulence and seismic modeling. It is desirable to have methods which combine the accuracy of global methods and the flexibility of local methods.

Discrete singular convolution (DSC) algorithm [18–20] has been proposed as a potential numerical approach for numerically handling singular integrations. In particular, it has emerged as an intriguing alternative in many situations—and as a superior one in a few cases for solving partial differential equations. The DSC algorithm has been applied to a number of scientific and engineering problems, including eigenvalue problems of both quantum [21] and classical [22] origins, analysis of stochastic process [18,19], simulation of fluid flow in simple [23,24] and complex [25] geometries, and nanoscale pattern formation in a circular domain [26]. It facilitated a synchronization scheme for shock capturing [27] and was utilized to integrate the sine-Gordon equation with initial values being close to the homoclinic orbit for which numerically induced chaos can easily occur [28]. In structural analysis, the DSC algorithm is the only available numerical method for solving a class of pressing structural design problems which require the prediction of high-frequency vibration levels [22,29–32]. The underlying mathematical structure of the DSC approach is the theory of distributions [33] and the theory of wavelet analysis [20,34]. One of the distributions used in the aforementioned applications is the Dirac delta function which is a generalized function following from the fact that it is integrable inside a particular interval but it does not have a value in the interval. Dirac was the first person who explicitly discussed the properties of δ in his classic text on quantum mechanics, thus δ is often called Dirac delta function. The general orthogonal series analysis of the delta distribution was studied by Walter [35], who discussed the numerical use of delta sequences as probability density estimators.

The connection of various numerical methods has always been a topic of great concern. The relation between the Galerkin algorithm and the Ritz variational principle was studied [36]. The reduction from a global Galerkin method to a global collocation method was made [5]. A unified view of finite difference and spectral collocation was given [6]. The connection of the finite-element, finite-difference and finite-volume methods is now well understood [37]. Recently, it was demonstrated that methods of global, local, Galerkin method, collocation and finite difference can be derived from a single starting point in the framework of the DSC algorithm [19].

The purpose of this paper is to make a detailed comparison of the performance between the DSC algorithm and the Fourier pseudospectral (FPS) method aiming at a better understanding of both approaches. This is illustrated by applying both approaches to three two-dimensional (2D) PDEs: the heat equation, the wave equation and the Navier–Stokes equation. These problems are selected because they differ in nature. Moreover, these problems are chosen in favor of the FPS method, which solves problems with periodic boundary conditions. Since the DSC algorithm is a local approach in general, there is no doubt that it is much more flexible than the FPS method in dealing with complex geometry and boundary conditions. Computational accuracy is of particular concern in this work. Overall, the DSC method is found to be competitive with, or even better than the spectral method on many computational aspects.

This paper is organized as follows. A brief description of both the DSC algorithm and FPS method is given in Section 2. The treatment of spatial discretizations by using both the DSC and the FPS methods is presented. These methods are further discussed in connection with the Runge–Kutta scheme for the temporal discretization. The numerical resolution of the DSC kernel is analyzed by discrete Fourier representation. Section 3 is devoted to numerical experiments and discussion. Three 2D problems are solved by using both numerical approaches, with a variety of parameters and time increments. Results are compared against accuracy. This paper ends with a conclusion.

2. Numerical methods

In this section, we give brief reviews to various spatial and temporal discretization schemes used in this work. The Fourier pseudospectral method is described after a short introduction to the discrete singular convolution algorithm. The latter is analyzed by using the discrete Fourier transform for its numerical resolution. Both methods are discussed in connection with the fourth-order Runge–Kutta scheme for the temporal discretization.

2.1. Discrete singular convolution

Singular convolutions are a special class of mathematical transformations, which occur in many science and engineering problems, such as Hilbert transform, Abel transform and Radon transform. It is most convenient to discuss the singular convolution in the context of the theory of distributions. The latter has a significant impact in mathematical analysis. It not only provides a rigorous justification for a number of informal manipulations in physical and engineering sciences, but also opens a new area of mathematics, which in turn gives impetus to many other mathematical disciplines, such as operator calculus, differential equations, functional analysis, harmonic analysis and transformation theory. In fact, the theory of wavelets and frames, a new mathematical branch developed in recent years, can also find its root in the theory of distributions.

Let T be a distribution and $\eta(t)$ be an element of the space of test functions. A singular convolution is defined as

$$F(t) = (T * \eta)(t) = \int_{-\infty}^{\infty} T(t-x)\eta(x)\,\mathrm{d}x. \tag{1}$$

Here $T(t-x)$ is a singular kernel. Depending on the form of the kernel T, the singular convolution is the central issue for a wide range of science and engineering problems. For example, the singular kernels of Hilbert type have a general form of

$$T(x) = \frac{1}{x^n}, \quad n = 1, 2, \ldots. \tag{2}$$

Here, kernel $T(x) = 1/x$ commonly occurs in electrodynamics, theory of linear response, signal processing, theory of analytic functions, and the Hilbert transform. When $n = 2$, $T(x) = 1/x^2$ is the kernel used in tomography. Another interesting example is the singular kernels of Abel type

$$T(x) = \frac{1}{x^\beta}, \quad 0 < \beta < 1. \tag{3}$$

These kernels can be recognized as the special cases of the singular integral equations of Volterra type of the first kind. The singular kernels of Abel type have applications in the area of holography and interferometry with phase objects (of practical importance in aerodynamics, heat and mass transfer, and plasma diagnostics). They are intimately connected with the Radon transform, for example, in determining the refractive index from the knowledge of a holographic interferogram. The other important example is the singular kernels of delta type

$$T(x) = \delta^{(n)}(x), \quad n = 0, 1, 2, \ldots. \tag{4}$$

Here, kernel $T(x) = \delta(x)$ is of particular importance for the interpolation of surfaces and curves (including atomic, molecular and biological potential energy surfaces, engineering surfaces and a variety of image processing and pattern recognition problems involving low-pass filters). Higher-order kernels, $T(x) = \delta^{(n)}(x)$, $n = 1, 2, \ldots$, are essential for numerically solving partial differential equations and for image processing, noise estimation, etc. However, since these kernels are singular, they cannot be directly digitized in computers. Hence, the singular convolution, (1), is of little numerical merit. To avoid the difficulty of using singular expressions directly in computer, we construct sequences of approximations (T_α) to the distribution T

$$\lim_{\alpha \to \alpha_0} T_\alpha(x) \to T(x), \tag{5}$$

where α_0 is a generalized limit. Obviously, in the case of $T(x) = \delta(x)$, each element in the sequence, $T_\alpha(x)$, is a delta sequence kernel. Note that one retains the delta distribution at the limit of a delta sequence kernel. Computationally, the Fourier transform of the delta distribution is unity. Hence, it is a *universal reproducing kernel* for numerical computations and an *all pass filter* for image and signal processing. Therefore, the delta distribution can be used as a starting point for the construction of either band-limited reproducing kernels or approximate reproducing kernels. By the Heisenberg uncertainty principle, exact reproducing kernels have bad localization in the time (spatial) domain, whereas, approximate reproducing kernels can be localized in both time and frequency representations. Furthermore, with a sufficiently smooth approximation, it is useful to consider a *discrete singular convolution* (DSC)

$$F_\alpha(t) = \sum_k T_\alpha(t - x_k) f(x_k), \tag{6}$$

where $F_\alpha(t)$ is an approximation to $F(t)$ and $\{x_k\}$ is an appropriate set of discrete points on which the DSC, Eq. (6), is well defined. Note that, the original test function $\eta(x)$ has been replaced by $f(x)$. The mathematical property or requirement of $f(x)$ is determined by the approximate kernel T_α. In general, the convolution is required to be Lebesgue integrable.

The delta distribution or the so-called Dirac delta function (δ) is a generalized function which is integrable inside a particular interval but itself needs not to have a value. It is given as a continuous linear functional on the space of test functions, $\mathcal{D}(-\infty, \infty)$,

$$\langle \delta, \phi \rangle = \delta(\phi) = \int_{-\infty}^{\infty} \delta\phi = \phi(0). \tag{7}$$

A delta sequence kernel, $\{\delta_\alpha(x)\}$, is a sequence of kernel functions on $(-\infty, \infty)$ which is integrable over every compact domain and their inner product with every test function ϕ converges to the delta distribution

$$\lim_{\alpha \to \alpha_0} \int_{-\infty}^{\infty} \delta_\alpha \phi = \langle \delta, \phi \rangle, \tag{8}$$

where the (real or complex) parameter α approaches α_0, which can either be ∞ or a limit value, depending on the situation (such a convention for α_0 is used throughout this paper). If α_0 represents a limit value, the corresponding delta sequence kernel is a fundamental family. Depending on the explicit form of δ_α, the condition on ϕ can be relaxed. For example, if δ_α is given as

$$\delta_\alpha(x) = \begin{cases} \alpha & \text{for } 0 < x < 1/\alpha,\ \alpha = 1, 2, \ldots, \\ 0 & \text{otherwise,} \end{cases} \tag{9}$$

then Eq. (8) makes sense for every ϕ in $C(-\infty, \infty)$.

There are many delta sequence kernels arising in the theory of partial differential equations, Fourier transforms and signal analysis, with completely different mathematical properties. The delta sequence kernels of Dirichlet type have very distinct mathematical properties and have been used in the present work. Let $\{\delta_\alpha\}$ be a sequence of functions on $(-\infty, \infty)$ which are integrable over every bounded interval. We call $\{\delta_\alpha\}$ a delta sequence kernel of Dirichlet type if

(1) $\int_{-a}^{a} \delta_\alpha \to 1$ as $\alpha \to \alpha_0$ for some finite constant a.
(2) For every constant $\gamma > 0$: $(\int_{-\infty}^{-\gamma} + \int_{\gamma}^{\infty})\delta_\alpha \to 0$ as $\alpha \to \alpha_0$.
(3) There are positive constants C_1 and C_2 such that

$$\|\delta_\alpha(x)\| \leqslant \frac{C_1}{\|x\|} + C_2$$

for all x and α.

Shannon's delta sequence kernel (or Dirichlet's continuous delta sequence kernel) is one of the most important examples of the delta sequence kernel of Dirichlet type and is given by the following (inverse) Fourier transform of the characteristic function, $\chi_{[-\alpha/2\pi,\alpha/2\pi]}$,

$$\delta_\alpha(x) = \int_{-\infty}^{\infty} \chi_{[-\alpha/2\pi,\alpha/2\pi]} \mathrm{e}^{-\mathrm{i}2\pi\xi x}\,\mathrm{d}\xi = \frac{\sin(\alpha x)}{\pi x}. \tag{10}$$

Alternatively, Shannon's delta sequence kernel can be given as an integration

$$\delta_\alpha(x) = \frac{1}{\pi}\int_0^{\alpha} \cos(xy)\,\mathrm{d}y, \tag{11}$$

or as the limit of a continuous product

$$\delta_\alpha(x) = \lim_{N\to\infty} \frac{\alpha}{\pi}\prod_{k=1}^{N} \cos\left(\frac{\alpha}{2^k}x\right) = \lim_{N\to\infty} \frac{1}{2^N\pi}\frac{\sin(\alpha x)}{\sin(\frac{\alpha}{2^N}x)}. \tag{12}$$

Numerically, Shannon's delta sequence kernel is one of the most important cases, because of its property of being an element of the Paley–Wiener reproducing kernel Hilbert space $B^2_{1/2}$

$$f(x) = \int_{-\infty}^{\infty} f(y)\frac{\sin\pi(x-y)}{\pi(x-y)}\,\mathrm{d}y, \quad \forall f \in B^2_{1/2}, \tag{13}$$

where $\forall f \in B^2_{1/2}$ indicates that, in its Fourier representation, the L^2 function f vanishes outside the interval $[-\frac{1}{2},\frac{1}{2}]$. The Paley–Wiener reproducing kernel Hilbert space $B^2_{1/2}$ is a subspace of the Hilbert space $L^2(R)$.

Shannon's delta sequence kernel is also known as a wavelet scaling function $\phi(x) = \delta_\pi(x)$. Shannon's mother wavelet can be constructed from the scaling function as

$$\psi(x) = \frac{\sin 2\pi x - \sin\pi x}{\pi x}, \tag{14}$$

with its Fourier expression

$$\hat{\psi}(\omega) = \chi_{[-1,1]}(\omega) - \chi_{[-1/2,1/2]}(\omega). \tag{15}$$

This is recognized as the ideal band pass filter and it satisfies the orthonormality conditions

$$\sum_{n=-\infty}^{\infty} \hat{\psi}(\omega+n) = 1 \tag{16}$$

and

$$\sum_{n=-\infty}^{\infty} \left\|\hat{\psi}(\omega+n)\right\|^2 = 1. \tag{17}$$

Technically, it can be shown that a system of orthogonal wavelets are generated from a single function, the "mother" wavelet ψ, by standard operations of translation and dilation

$$\psi_{mn}(x) = 2^{-m/2}\psi\left(\frac{x}{2^m} - n\right), \quad m,n \in Z. \tag{18}$$

Shannon's delta sequence kernel can be derived from the generalized Lagrange interpolation formula

$$S_k(x) = \frac{G(x)}{G'(x_k)(x-x_k)}, \tag{19}$$

where $G(x)$ is an entire function given by

$$G(x) = (x - x_0) \prod_{k=1}^{\infty} \left(1 - \frac{x}{x_k}\right)\left(1 - \frac{x}{x_{-k}}\right), \tag{20}$$

and G' denotes the derivative of G. For a function bandlimited to B, the generalized Lagrange interpolation formula $S_k(x)$ of Eq. (19) can provide an exact result

$$f(x) = \sum_{k \in Z} f(y_k) S_k(x), \tag{21}$$

whenever the set of non-uniform sampling points satisfies

$$\sup_{k \in Z} \left| x_k - \frac{k\pi}{B} \right| < \frac{\pi}{4B}, \tag{22}$$

where the symbol Z denotes the set of all integers. This is called the Paley and Wiener sampling theorem in the literature.

If $\{x_k\}_{k \in Z}$ are limited to a set of points on a uniform infinite grid ($x_k = k\Delta = -x_{-k}$), Eq. (20) can be simplified as

$$G(x) = x \prod_{k=-\infty,\ k \neq 0}^{\infty} \left(1 - \frac{x}{k\Delta}\right) = x \prod_{k=1}^{\infty} \left(1 - \frac{x^2}{k^2\Delta^2}\right) = \Delta \frac{\sin \frac{\pi}{\Delta} x}{\pi}. \tag{23}$$

Since $G'(x_k)$ reduces to

$$G'(x_k) = (-1)^k \tag{24}$$

on a uniform grid, Eq. (19) gives rise to

$$S_k(x) = \frac{G(x)}{G'(x_k)(x - x_k)} = \frac{(-1)^k \sin \frac{\pi}{\Delta} x}{\frac{\pi}{\Delta}(x - k\Delta)} = \frac{\sin \frac{\pi}{\Delta}(x - x_k)}{\frac{\pi}{\Delta}(x - x_k)}. \tag{25}$$

Obviously, $\frac{\sin(\pi/\Delta)(x - x_k)}{(\pi/\Delta)(x - x_k)}$ is an approximation to the delta distribution

$$\lim_{\Delta \to 0} \frac{\sin \frac{\pi}{\Delta}(x - x_k)}{\frac{\pi}{\Delta}(x - x_k)} \to \delta(x - x_k). \tag{26}$$

In fact, the generalized Lagrange interpolation formula directly gives rise to the delta distribution under an appropriate limit

$$\lim_{\max \Delta x \to 0} S_k(x) = \lim_{\max \Delta x \to 0} \frac{G(x)}{G'(x_k)(x - x_k)} \to \delta(x - x_k), \tag{27}$$

where $\max \Delta x$ is the largest Δx on the grid.

Both $\phi(x)$ and its associated wavelet play a crucial role in information theory and the theory of signal processing. However their usefulness is limited by the fact that $\phi(x)$ and $\psi(x)$ are infinite impulse response (IIR) filters and their Fourier transforms $\hat{\phi}(\omega)$ and $\hat{\psi}(\omega)$ are not differentiable. From the computational point of view, $\phi(x)$ and $\psi(x)$ do not have finite moments in the coordinate space; in other words, they are de-localized. This non-local feature in the coordinate space is related to its bandlimited character in the Fourier representation by the Heisenberg uncertainty principle.

According to the theory of distributions, the smoothness, regularity and localization of a temper distribution can be improved by a function of the Schwartz class. We apply this principle to regularize singular convolution kernels

$$\delta_{\alpha,\sigma}(x) = R_\sigma(x)\delta_\alpha(x), \quad \sigma > 0, \tag{28}$$

where R_σ is a *regularizer* which has properties

$$\lim_{\sigma\to\infty} R_\sigma(x) = 1 \tag{29}$$

and

$$R_\sigma(0) = 1. \tag{30}$$

Here Eq. (29) is a general condition that a regularizer must satisfy, while Eq. (30) is specifically for a *delta regularizer*, which is used in regularizing a delta kernel. Various delta regularizers can be used for numerical computations. A good example is the Gaussian

$$R_\sigma(x) = \exp\left[-\frac{x^2}{2\sigma^2}\right]. \tag{31}$$

Gaussian regularizer is a Schwartz class function and has excellent numerical performance. However, we noted that in certain eigenvalue problems, no regularization is required if the potential is smooth and bounded from below (e.g., the harmonic oscillator potential $\frac{1}{2}x^2$).

The spatial discretization by using the DSC algorithm is discussed for solving PDEs. Consider a PDE of the general form

$$\frac{\partial u}{\partial t} = F(u,t), \tag{32}$$

where $u(x,y,t) \in R^2 \times (0,\infty)$. It is assumed that the part of Eq. (32) that involves differential operators can be written as

$$\mathcal{D} = \sum_{n=0}\sum_{m=0} c_{nm}\left(\frac{\partial}{\partial x}\right)^n\left(\frac{\partial}{\partial y}\right)^m u(x,y,t), \tag{33}$$

where c_{nm} are constant coefficients. The task of spatial discretization is to provide discrete approximations to these differential operators. In the DSC algorithm, the differentiations are approximated as

$$\mathcal{D}_{ij} = \sum_{n=0}\sum_{m=0} c_{nm}\sum_{k=i-M_x}^{i+M_x}\sum_{l=j-M_y}^{j+M_y}\delta^{(n)}_{\alpha,\sigma}(x_i - x_k)\delta^{(m)}_{\alpha,\sigma}(y_j - y_l)u_{ij}, \tag{34}$$

where $u_{ij} = u(x_i, y_j, t)$, M_x and M_y are the half length of supports in the x- and y-directions, respectively. We choose a simple example of the DSC kernel, the regularized Shannon's kernel (RSK)

$$\delta_{\alpha,\sigma}(x) = \frac{\sin(\pi/\Delta)(x - x_k)}{(\pi/\Delta)(x - x_k)}\exp\left[-\frac{(x - x_k)^2}{2\sigma^2}\right] \tag{35}$$

to carry our study, although many other excellent DSC kernels can also be used [23]. The derivatives of the DSC kernels are directly given by

$$\begin{aligned}\delta^{(n)}_{\alpha,\sigma}(x_i - x_j) &= \left[\frac{\mathrm{d}^n}{\mathrm{d}x^n}\delta_{\alpha,\sigma}(x - x_j)\right]_{x=x_i},\\ \delta^{(n)}_{\alpha,\sigma}(y_i - y_j) &= \left[\frac{\mathrm{d}^n}{\mathrm{d}y^n}\delta_{\alpha,\sigma}(y - y_j)\right]_{y=y_i},\end{aligned} \tag{36}$$

where $\delta_{\alpha,\sigma}(x - x_j)$ is given by Eq. (35).

2.2. Fourier pseudospectral method

In this subsection, we present a brief description of the Fourier pseudospectral method by Sanders et al. [38]. In general, spectral methods refer to all those global methods that can be interpreted as projection methods over a finite-dimensional space of polynomials with respect to an appropriate inner product. Three most important spectral formulations are the *Galerkin*, *tau*, and *collocation* methods. The spectral method employed in the present work is the collocation scheme with the discrete Fourier basis as the trial functions and is referred as the Fourier pseudospectral (FPS) method [6,38,39].

In general, the Fourier pseudospectral method is implemented with the use of the periodic boundary conditions. A spatial domain, $[0, L_x] \times [0, L_y]$, is discretized into N_x and N_y equidistance points in the x- and y-directions with grid spacings $\Delta x = L_x/N_x$ and $\Delta y = L_y/N_y$. The forward discrete Fourier transform (DFT) of a function $u(x, y, t)$ is defined as

$$\begin{aligned} &\hat{u}_{l,m} = \sum_{j=0}^{N_x-1} \sum_{k=0}^{N_y-1} u_{j,k} \mathrm{e}^{2\pi \mathrm{i}(lj/N_x + mk/N_y)}, \\ &\quad l = -N_x/2, -N_x/2+1, \ldots, N_x/2-1; \\ &\quad m = -N_y/2, -N_y/2+1, \ldots, N_y/2-1 \end{aligned} \tag{37}$$

and its inverse discrete Fourier transform is given by

$$\begin{aligned} &u_{j,k} = \frac{1}{N_x N_y} \sum_{l=-N_x/2}^{N_x/2-1} \sum_{m=-N_y/2}^{N_y/2-1} \hat{u}_{l,m} \mathrm{e}^{-2\pi \mathrm{i}(jl/N_x + km/N_y)}, \\ &\quad j = 0, 1, \ldots, N_x - 1; \quad k = 0, 1, \ldots, N_y - 1. \end{aligned} \tag{38}$$

In the present periodic boundary condition, what is actually needed is not the Fourier expansion coefficients, but a fast way to compute convolutions. The fast Fourier transform (FFT) is the natural solution and is thus used in the present work. As such, the number of the grid points N_x and N_y are taken to be the power of 2.

What is required for solving PDEs is the approximation of the differential operators. In the FPS methods, such an approximation is given by algebraic multiplications. If $u(x, y, t)$ is a sufficiently smooth function of its variables, its spatial derivatives can be evaluated as

$$\begin{aligned} \left[\frac{\mathrm{d}^n u}{\mathrm{d}x^n}\right]_{j,k} &= \frac{1}{N_x N_y} \sum_{l=-N_x/2}^{N_x/2-1} \sum_{m=-N_y/2}^{N_y/2-1} \left(\frac{2\pi \mathrm{i} l}{L_x}\right)^n \hat{u}_{l,m} \mathrm{e}^{-2\pi \mathrm{i}(jl/N_x + km/N_y)}, \\ \left[\frac{\mathrm{d}^n u}{\mathrm{d}y^n}\right]_{j,k} &= \frac{1}{N_x N_y} \sum_{l=-N_x/2}^{N_x/2-1} \sum_{m=-N_y/2}^{N_y/2-1} \left(\frac{2\pi \mathrm{i} m}{L_y}\right)^n \hat{u}_{l,m} \mathrm{e}^{-2\pi \mathrm{i}(jl/N_x + km/N_y)}. \end{aligned} \tag{39}$$

This global approximation is very efficient as long as N_x and N_y are not too large.

2.3. Temporal discretization

As the fourth-order Runge–Kutta (RK4) scheme is a robust approach for the temporal discretization, it is employed for both the heat equation and the wave equation. However, the Navier–Stokes equation with the incompressible condition is an ill-posed problem and a Poisson equation for pressure is derived by using the incompressible condition. The resulting equations of velocity and pressure are updated by using the Crank–Nicolson scheme and details are given in the next section. Here, we describe the RK4 scheme.

For the time discretization of the heat and wave equations, we rewrite the spatially discretized Eq. (32) in its matrix form as

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = \mathbf{F}(\mathbf{u}, t), \tag{40}$$

where $\mathbf{u}$ is a vector having components $[u_{11}, u_{21}, \ldots, u_{N_x 1}, u_{12}, \ldots, u_{N_x N_y}]^{\mathrm{T}}$. The fourth-order Runge–Kutta scheme (RK4) is used to solve for u at each time step $n+1$,

$$\begin{aligned}
&\mathbf{u}^{n+1} = \mathbf{u}^n + \tfrac{1}{6}(\mathbf{r}_1 + 2\mathbf{r}_2 + 2\mathbf{r}_3 + \mathbf{r}_4),\\
&\mathbf{r}_1 = \Delta t\mathbf{F}(\mathbf{u}^n, t_n),\\
&\mathbf{r}_2 = \Delta t\mathbf{F}\big(\mathbf{u}^n + \tfrac{1}{2}\mathbf{r}_1, t_n + \tfrac{1}{2}\Delta t\big),\\
&\mathbf{r}_3 = \Delta t\mathbf{F}\big(\mathbf{u}^n + \tfrac{1}{2}\mathbf{r}_2, t_n + \tfrac{1}{2}\Delta t\big),\\
&\mathbf{r}_4 = \Delta t\mathbf{F}\big(\mathbf{u}^n + \mathbf{r}_3, t_n + \Delta t\big).
\end{aligned} \tag{41}$$

Except for specified cases, the RK4 is used in association with both the DSC algorithm and the FPS method in the present work.

2.4. Discrete Fourier analysis of the DSC kernels

In general, discrete Fourier basis is *exact* for time-independent problems whose solutions are given by bandlimited, periodic L^2 functions. The use of discrete Fourier analysis for characterizing the errors of difference approximations has been extensively described [40–42]. Indeed, the Fourier analysis characterizes the *Fourier* resolution of an interpolation or differential scheme applied to a class of bandlimited periodic functions. In this subsection, we discuss the discrete Fourier analysis of DSC kernels in association with the approximation of bandlimited periodic functions and their derivatives. Such an analysis is to project the DSC kernel into the discrete Fourier basis. The result provides us an idea about accuracy of the DSC kernels used for bandlimited periodic functions. It is point out that such a result does not hold when the function is not bandlimited periodic. The analysis in this subsection is carried out in one-dimensional spatial coordinate.

Using discrete Fourier analysis, the DSC kernel function, i.e. the RSK (35), and its nth derivative $\{\delta^{(n)}_{\alpha,\sigma}(x_j)\}_{j=0}^{N-1}$ can be transformed into the Fourier representation $\hat{\delta}^{(n)}_{\alpha,\sigma}(l)$

$$\begin{aligned}
&\hat{\delta}^{(n)}_{\alpha,\sigma}(l) = \sum_{j=0}^{N-1} \delta^{(n)}_{\alpha,\sigma}(x_j)\mathrm{e}^{\mathrm{i}2\pi lj/N},\\
&\quad l = -N/2, -N/2+1, \ldots, N/2-1.
\end{aligned} \tag{42}$$

Here, $\hat{\delta}_{\alpha,\sigma}(l)$ are a set of low pass filter coefficients which are distributed over the discrete wave numbers $\{2\pi l/L\}_{l=-N/2}^{N/2-1}$, i.e. the wave numbers that can be resolved range from $-\pi/\Delta$ to π/Δ. The quantity π/Δ is the *Nyquist frequency*, which is the maximum wave number that can be distinguished by the discretization with mesh size Δ. Similarly, $\hat{\delta}^{(1)}_{\alpha,\sigma}(l)$, $\hat{\delta}^{(2)}_{\alpha,\sigma}(l), \ldots$ are sets of high pass filter coefficients. For convenience, we introduce a scaled wavenumber $\omega_l = 2\pi l\Delta/L = 2\pi l/N$ and a scaled coordinate $s_j = x_j/\Delta$. The Fourier modes in terms of these new definitions are simply $\exp(\mathrm{i}\omega_l s_j)$. Therefore, the domain of the scaled wavenumber is converted into $[-\pi, \pi]$. As such, the Dirichlet kernel generated from the discrete Fourier basis will be an ideal low pass filter in the domain $[-\pi, \pi]$, i.e. it is exact for periodic L^2 functions bandlimited to π. It is noted that all other spectral bases cannot give exact interpolation for periodic L^2 functions bandlimited to π.

For the first order derivative, the FPS provides the exact response function $\mathrm{i}\omega$ within the frequency band $[-\pi, \pi]$. In Fig. 1, the response functions of $\hat{\delta}^{(1)}_{\sigma,\alpha}(l)$ are plotted for a few different M and $r = \sigma/\Delta$ values. When $M = 16$,

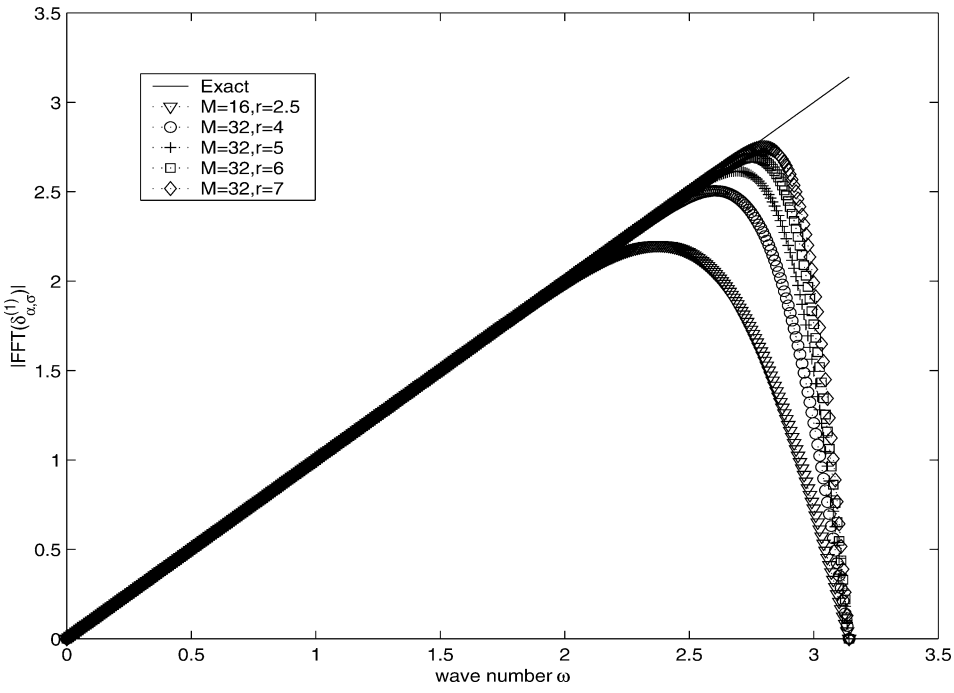

Fig. 1. The Fourier resolutions of the first order derivative approximated by using a few kernels.

the DSC kernel gives accurate response at relatively low frequency. As the frequency increases, the kernel response decreases gradually to zero. At a larger M value ($M = 32$), the DSC kernel gives accurate response up to higher frequency components. The responses obtained at a few different r values are depicted in Fig. 1. It appears that for a sufficiently large M, the larger the r is, the better the response is. However, one has to be very careful since errors that are smaller than 1% cannot be noticed from the plot. A more detailed plot, Fig. 2, indicates that if the r is too large, the accuracy of the DSC kernel might degrade. It is found that there exists some optimal range of r, depending on the value of M. In Fig. 2, the differences between the DSC responses at $M = 32$ and the exact response are plotted for the parameter r being 4, 5, 6 and 7. As shown in the plot, the larger the value of r, the better the approximation to high frequency will be. Therefore, it implies we can achieve a better approximation to the high frequency range when we use a large enough r. However, there is a trade off between the accuracy for the high frequency part and that for the low frequency part. As seen from Fig. 2, the larger the value of r, the larger the errors in the low frequency responses. Therefore, from Figs. 1 and 2, it can be concluded that the optimal value r should be chosen depending on the nature of problems to be solved, and a compromise between the high frequency resolution and the low frequency resolution should be achieved.

Accuracy is one of the most important factors in choosing a computational method for a practical application. In the Fourier space, the FPS basis provides the most accurate (i.e. exact) representation. However, the same FPS basis is not the best basis in a polynomial space, where the finite difference approximation can be proved to be the most accurate basis. Depending on the nature of problems under study, there exist different numerical methods which are the most accurate. Although the DSC algorithm is not as accurate as the FPS method for approximating bandlimited periodic functions, it can be more accurate than the FPS method for many other problems.

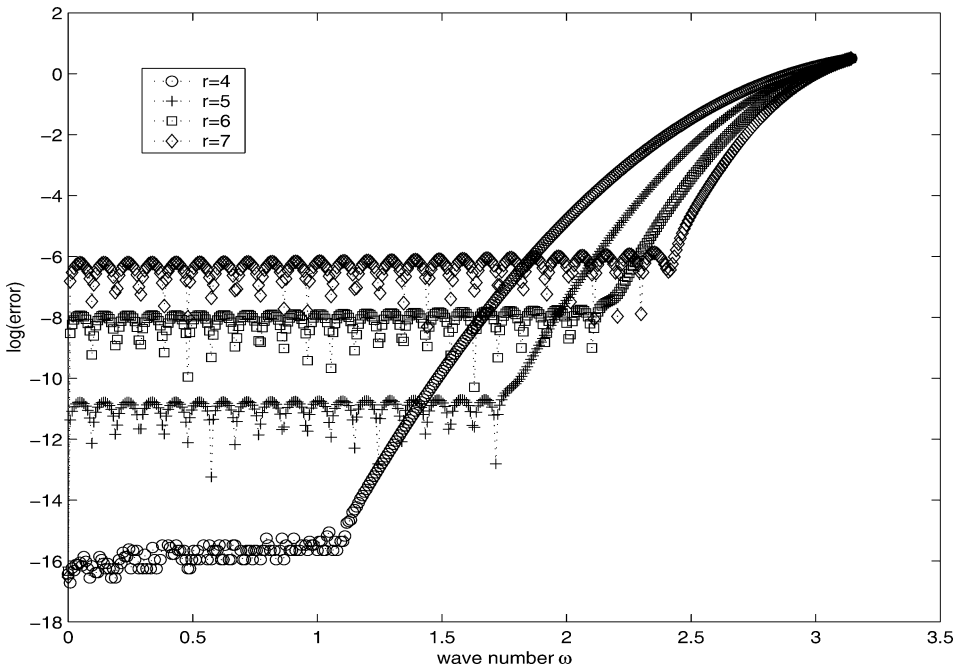

Fig. 2. The errors of a few kernels in approximating the first order derivative.

3. Results and discussion

In this section, we employ three two-dimensional (2D) problems, the heat equation, the wave equation, and the Navier–Stokes equation, to make a detailed comparison between the Fourier pseudospectral (FPS) method and the DSC algorithm. The DSC algorithm is realized by using the regularized Shannon's kernel (RSK) (35). The spatial computational support in the DSC is set to $2M + 1$, i.e. each local spatial grid makes use of the values of its $2M$ nearest neighbors and the point itself. We take three values for M, 8, 16 and 32, to compare the accuracy and stability of the DSC and the FPS method. The FFT is utilized for the FPS method. The fourth-order Runge–Kutta (RK4) scheme is used for time integrations in solving the heat and the wave equations. For simplicity, we take a square domain ($L_x = L_y$) with the same number of grid points in each direction ($N = N_x = N_y$) in our computations. For a fair comparison, both methods are tested with the same grid spacings $\Delta = \Delta x = \Delta y$ in the square domain.

To compare the accuracy of both methods, we calculate the errors of numerical approximations $\bar{u}$ with respect to the exact analytical solutions u of each test problem. Two kinds of numerical error measures, i.e. L_∞ and L_2 errors, are used in this work. These error measures are defined as $L_\infty = \max|\bar{u}_{ij} - u_{ij}|$ and $L_2 = (\Delta^2 \sum_{ij}(\bar{u}_{ij} - u_{ij})^2)^{1/2}$, respectively. Some brief discussions are given based on the detailed comparison of both methods in terms of accuracy and stability.

3.1. The heat equation

We consider the 2D heat equation or diffusion equation

$$\frac{\partial u(x,y,t)}{\partial t} = a^2\left(\frac{\partial^2 u(x,y,t)}{\partial x^2} + \frac{\partial^2 u(x,y,t)}{\partial y^2}\right). \tag{43}$$

The heat equation is a good test problem because it admits an exact solution of the form

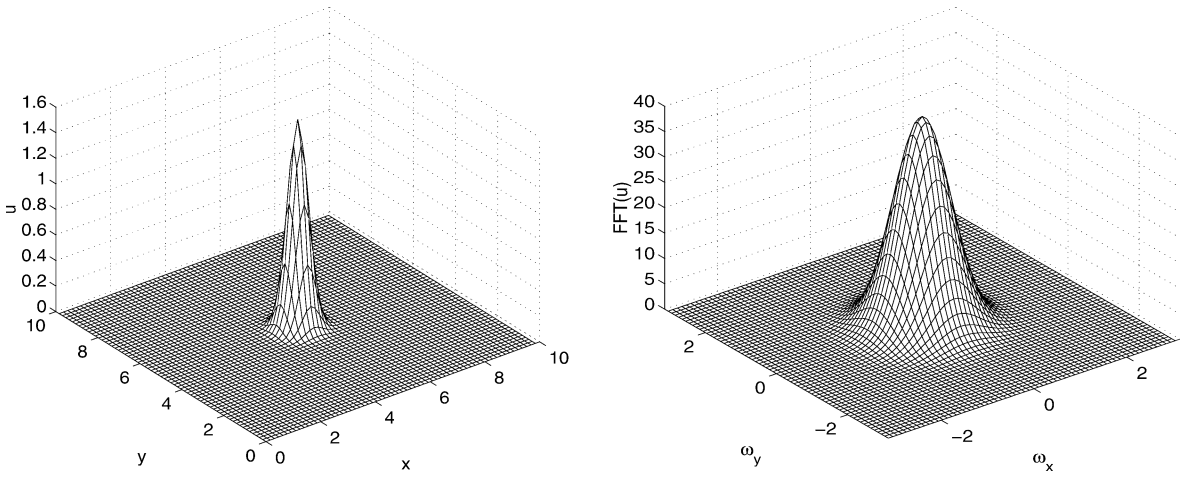

Fig. 3. Plot of solution of the heat equation at time $t = 0.1$ and $a = 0.7$. The figure on the right is the Fourier Transform of the left one.

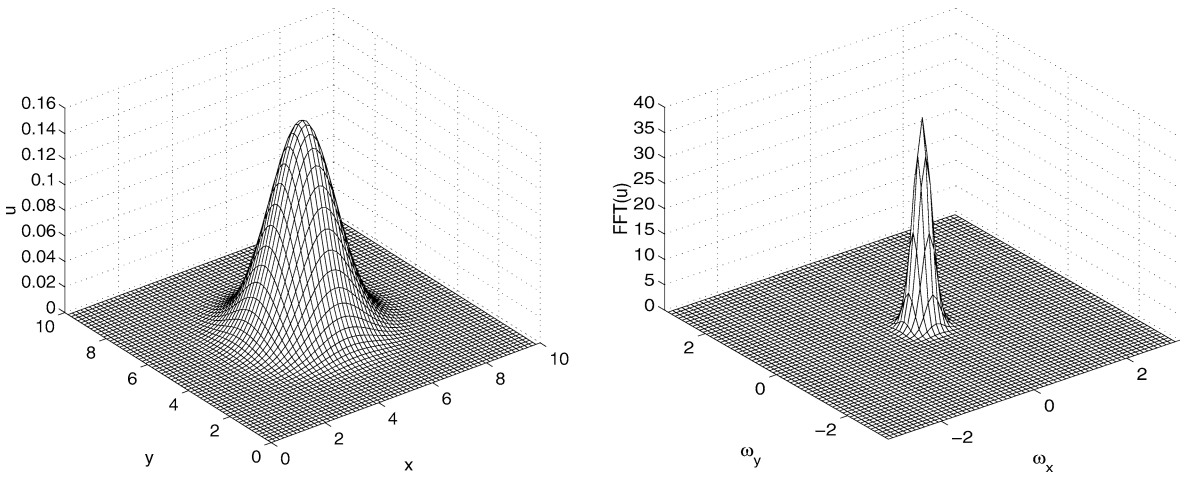

Fig. 4. Plot of solution of the heat equation at time $t = 1$ and $a = 0.7$. The figure on the right is the Fourier Transform of the left one.

$$u(x, y, t) = \frac{1}{4a^2\pi t} \mathrm{e}^{-(x^2+y^2)/(4a^2t)}. \tag{44}$$

Note that, the Fourier transform of Eq. (44) is still a Gaussian, i.e.

$$\hat{u}(\omega_x, \omega_y, t) = \frac{1}{\sqrt{2\pi}} \int\int_{-\infty}^{\infty} u(x, y, t) \mathrm{e}^{\mathrm{i}x\omega_x + \mathrm{i}y\omega_y} \,\mathrm{d}x \,\mathrm{d}y = \frac{1}{2\pi} \mathrm{e}^{-a^2t(\omega_x^2+\omega_y^2)}. \tag{45}$$

To utilize the FPS method for this problem, we have to cast the problem to a large computational domain so that the periodic boundary condition is applicable. We consider the heat equation in a 2D square domain $[0, 10] \times [0, 10]$. The parameter a in the heat equation (43) is set to 0.7 in all computations. Since it is difficult to accurately represent the initial condition, a delta pulse on a grid, we choose the initial value as the Gaussian wave at $t = 0.1$

$$u_0(x, y) = \frac{1}{4a^2\pi t} \mathrm{e}^{-[(x-x_0)^2+(y-y_0)^2]/(4a^2t)}, \tag{46}$$

Table 1
Numerical solutions of the heat equation ($\Delta t = 0.001$)

N	t			Periodical b.c.		Dirichlet b.c.	
			FPS	RSK-8	RSK-16	RSK-8	RSK-16
16	0.5	L_2	2.69E−2	2.36E−2	2.61E−2	2.36E−2	2.61E−2
		L_∞	2.45E−2	1.55E−2	2.15E−2	1.55E−2	2.15E−2
	1.0	L_2	1.36E−2	1.32E−2	1.36E−2	1.32E−2	1.36E−2
		L_∞	7.62E−3	6.50E−3	7.48E−3	6.50E−3	7.48E−3
	1.5	L_2	1.00E−2	9.94E−3	1.00E−2	9.94E−3	1.00E−2
		L_∞	4.30E−3	4.03E−3	4.29E−3	4.03E−3	4.29E−3
	2.0	L_2	8.29E−3	8.24E−3	8.29E−3	8.24E−3	8.29E−3
		L_∞	2.97E−3	2.87E−3	2.97E−3	2.87E−3	2.97E−3
			FPS	RSK-16	RSK-32	RSK-16	RSK-32
32	0.5	L_2	1.93E−7	8.09E−6	2.36E−7	8.09E−6	2.36E−7
		L_∞	1.70E−7	5.86E−6	9.81E−8	5.86E−6	9.81E−8
	1.0	L_2	9.12E−7	2.68E−7	2.45E−7	1.37E−7	7.96E−8
		L_∞	4.69E−7	9.06E−8	9.06E−8	5.55E−8	2.89E−8
	1.5	L_2	5.08E−5	2.27E−5	2.26E−5	1.02E−5	1.01E−5
		L_∞	2.20E−5	2.33E−6	7.34E−6	3.38E−6	3.34E−6
	2.0	L_2	3.72E−4	2.10E−4	2.10E−4	1.14E−4	1.13E−4
		L_∞	1.38E−4	6.06E−5	6.06E−5	3.36E−5	3.33E−5

where the point (x_0, y_0) is the center of the 2D square domain. This initial value is depicted in Fig. 3 ($a = 0.7$). The Fourier transform of the Gaussian initial wave is also shown in Fig. 3. The corresponding plots of $u(x, y, t = 1)$ in both the coordinate space and the Fourier space are shown in Fig. 4. As can be seen from these plots, the Gaussian wave grows wider as time increases, whereas the corresponding Fourier transform of the Gaussian wave becomes narrower as time increases.

We use both the DSC and FPS methods to approximate the 2nd order derivatives in Eq. (43). The DSC discretization (34) of the heat equation is given as

$$\frac{\mathrm{d}u_{ij}}{\mathrm{d}t} = a^2\left[\sum_{k=i-M}^{i+M}\delta^{(2)}_{\alpha,\sigma}(x_i - x_k) + \sum_{l=j-M}^{j+M}\delta^{(2)}_{\alpha,\sigma}(y_j - y_l)\right]u_{i,j}. \tag{47}$$

The DSC algorithm is implemented with both periodic and Dirichlet boundary conditions (b.c.). Here, the Dirichlet boundary condition assumes that the function values outside the boundaries of 2D domain $[0, 10] \times [0, 10]$ vanish. We use the RK4 scheme (41) for the time integration of above (47) to solve for a set of u_{ij} at each time step. Time increment used in our computations is usually very small so that most errors are due to spatial discretizations. The time integration cannot be too long for this problem so as to avoid significant boundary effects.

The computational errors with respect to the exact solution are listed in Tables 1 and 2. The FPS method is compared with the DSC algorithm with both the periodic and Dirichlet boundary conditions. Both the time

Table 2
Numerical solutions of the heat equation ($\Delta t = 0.01$)

N	t			Periodical b.c.		Dirichlet b.c.	
			FPS	RSK-8	RSK-16	RSK-8	RSK-16
16	0.5	L_2	2.69E–2	2.36E–2	2.61E–2	2.36E–2	2.61E–2
		L_∞	2.44E–2	1.55E–2	2.15E–2	1.55E–2	2.15E–2
	1.0	L_2	1.36E–2	1.32E–2	1.36E–2	1.32E–2	1.36E–2
		L_∞	7.62E–3	6.50E–3	7.48E–3	6.50E–3	7.48E–3
	1.5	L_2	1.00E–2	9.94E–3	1.00E–2	9.94E–3	1.00E–2
		L_∞	4.13E–3	4.03–E3	4.29E–3	4.03–E3	4.29E–3
	2.0	L_2	8.29E–3	8.24E–3	8.28E–3	8.24E–3	8.28E–3
		L_∞	2.97E–3	2.87E–3	2.97E–3	2.87E–3	2.97E–3
			FPS	RSK-16	RSK-32	RSK-16	RSK-32
32	0.5	L_2	2.35E–7	8.06E–6	2.64E–7	8.05E–6	2.64E–7
		L_∞	2.16E–7	5.82E–6	1.05E–7	5.82E–6	1.05E–7
	1.0	L_2	9.12E–7	2.67E–7	2.45E–7	1.35E–7	8.00E–7
		L_∞	4.69E–7	9.06E–8	9.06E–8	5.40E–8	2.89E–8
	1.5	L_2	5.08E–5	2.26E–5	2.26E–5	1.02E–5	1.01E–5
		L_∞	3.72E–4	2.10E–4	2.10E–4	1.22E–4	1.21E–4
	2.0	L_2	1.49E–3	1.14E–4	1.13E–4	6.66E–4	6.62E–4
		L_∞	1.38E–4	6.06E–5	6.06E–5	3.36E–5	3.33E–5

increment Δt and the number of grid points N are varied to test the performance of both methods. We employ an equal spatial discretization along x- and y-directions, i.e. $N = N_x = N_y$. Although N can take any integer value in the DSC algorithm, in order to compare with FPS in which N is required to be powers of 2, we choose $N = 16$ and $N = 32$ for our computations. In the DSC scheme, we have one more parameter M which determines the support size of the DSC kernels. The value of M is usually set smaller than N, therefore we choose $M = 8$ and $M = 16$. The corresponding DSC methods are referred to as RSK-8 and RSK-16, respectively.

From Tables 1 and 2, we observe that the DSC with the periodic boundary condition gives the same order of accuracy as the FPS (see the RSK-16 with periodic b.c. and the FPS in Table 2). The comparison has been done for two time steps $\Delta t = 0.001$ and $\Delta t = 0.01$ with two spatial discretizations $N = 16$ and $N = 32$. The results indicate that, DSC algorithm is competitive with the FPS method.

Moreover, a careful comparison indicates that the boundary condition used in the DSC algorithm affects the accuracy of the method. As in Tables 1 and 2, all the RSK-8, RSK-16 and RSK-32 with the Dirichlet boundary condition are similar to the corresponding DSC schemes with the periodical boundary condition, and more accurate than the FPS method with the same N value. The importance of boundary condition is most obvious for a larger N value. Note that, for a large enough $N = 32$, the support size M has less effect on the accuracy of the DSC algorithm in this problem. The dominant factor that affects the accuracy of the DSC algorithm is the boundary condition at $N = 32$.

Obviously, as a local method, the DSC algorithm is more robust than the FPS method in choosing the number of grid points, and in dealing with complex geometry and boundary condition. However, it is not so obvious why the DSC algorithm is even more accurate than the FPS method for the heat equation. In general, global methods should be more accurate than local ones. Indeed, we have shown in the last section that the discrete Fourier basis is more accurate than the DSC kernels. In the present case, the solution of the heat equation is the Gaussian which is invariant with respect to the Fourier transform, i.e. it is also a Gaussian in the Fourier domain as shown in Eq. (45). As a result, the Gaussian is not bandlimited, nor is it periodic. In such a case, the FPS method does not provide one of the best approximations for solving the heat equation. In contrast, the DSC algorithm is indeed not as accurate as the FPS method for approximating the bandlimited periodic function as illustrated by the earlier discrete Fourier analysis. However, for some non-bandlimited functions, the DSC algorithm turns out to be more accurate.

3.2. The wave equation

Propagation of electromagnetic wave is governed by Maxwell's equations. In a charge-free homogeneous medium, the wave equation can be used to describe the motion of an electric or magnetic field. For simplicity, we consider the 2D wave equation of the form

$$\frac{\partial^2 u(x,y,t)}{\partial t^2} = v^2\left(\frac{\partial^2 u(x,y,t)}{\partial x^2} + \frac{\partial^2 u(x,y,t)}{\partial y^2}\right), \tag{48}$$

where v is the velocity of the wave. A general solution to Eq. (48) has form of $f(x \pm y \pm vt)$.

The wave equation is different from the heat equation (43) due to the second order derivative in time. To use the RK4 scheme (41) for the time evolution of Eq. (48), we rewrite the wave equation as two coupled PDEs

$$\begin{aligned} \frac{\partial w(x,y,t)}{\partial t} &= v^2\left(\frac{\partial^2 u(x,y,t)}{\partial x^2} + \frac{\partial^2 u(x,y,t)}{\partial y^2}\right), \\ \frac{\partial u(x,y,t)}{\partial t} &= w(x,y,t). \end{aligned} \tag{49}$$

The spatial discretizations using both the FPS method and the DSC algorithm are described in Section 2.

In our computation, we set the velocity $v = 1.0$ in all test cases. The computational domain is set to $[0, 2\pi] \times [0, 2\pi]$, on which the periodic boundary condition is imposed. We choose the initial condition as $u(x, y, 0) = \cos(x + y)$ and thus, the exact solution of the wave equation (48) is given by

$$u(x,y,t) = \cos(x + y + vt). \tag{50}$$

Both the DSC and FPS methods are employed to compute the numerical solution of the wave equation (48) and their performance is compared. In Fig. 5, the waveform of the solution is plotted. Two types of spatial discretizations ($N = 32$ and $N = 64$) and time increments ($\Delta t = 0.0005$ and $\Delta t = 0.01$) are considered in our numerical tests. The computational errors with respect to the exact solution (50) are listed in Table 3.

The results in the first part of Table 3 are obtained with a very small time increment ($\Delta t = 0.005$) so that all errors are mainly originated from the spatial discretization. The comparison is done for the time t up to 100. It is seen that when $M = 32$, both the DSC and the FPS give the same order of accuracy and both methods can achieve the machine precision. Next, we consider a slightly larger time increment ($\Delta t = 0.01$). We still observe that the accuracy of both the DSC and the FPS is of the same order as shown in second part of Table 3.

Another important parameter is the support size M in the DSC kernel. The accuracy of the DSC algorithm is controllable by adjusting the M value. It is seen that the DSC algorithm is very accurate when $M = 16$. In fact, it achieves machine precision with a large support size $M = 32$ at a small time increment ($\Delta t = 0.0005$). However, at a large time increment ($\Delta t = 0.01$), all the three schemes, the FPS, the RSK-16 and the RSK-32, give same order of accuracy, and the choice of M does not affect the accuracy of the DSC algorithm. This is due to the fact that, for a large Δt, the main errors come from the time discretization.

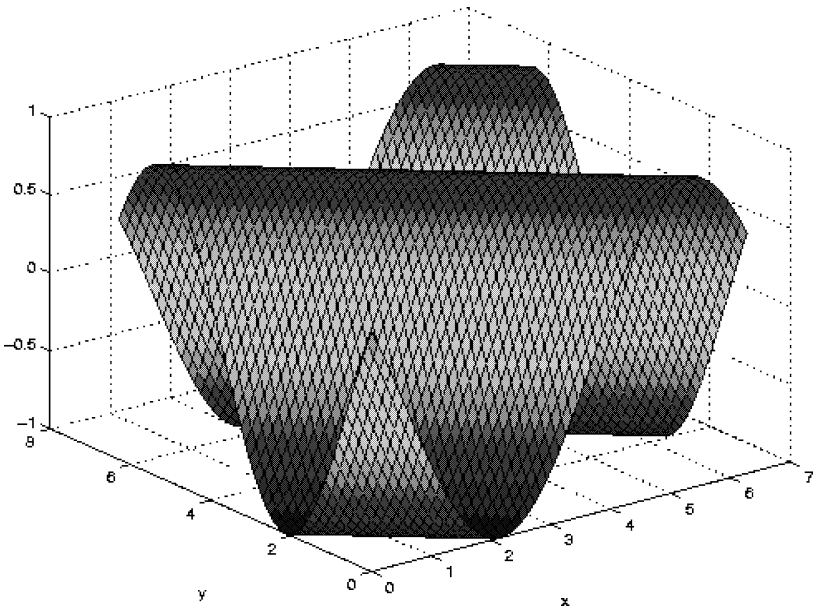

Fig. 5. The solution of the wave equation.

3.3. The Navier–Stokes equation

The Navier–Stokes (NS) equation is the governing equation for aerodynamics and fluid dynamics and has applications in various problems, including weather forecasting, turbulence simulation, and chemical reaction, etc. Some of these problems invoke complicated boundary conditions, which cause difficulties for global methods, including the FPS method. There is no general analytical solution to the Navier–Stokes equation, except for a few simple cases. Recently, the DSC algorithm has been employed to solve the Navier–Stokes equation with simple [23] and complex geometries [25]. In this work, we consider the Taylor problem, i.e. the Navier–Stokes equation with periodic boundary conditions, to make a comparison of the DSC and FPS methods. This problem admits an analytical solution and can be solved by using both the FPS and the DSC, and thus, the performance of both methods can be compared in detail. Both approaches are formulated in association with the Crank–Nicolson scheme.

3.3.1. The problem and solution scheme

We consider the Navier–Stokes equation in its primitive variable, describing an incompressible fluid flow

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u} = -\nabla p + \frac{1}{\mathrm{Re}}\nabla^2\mathbf{u}, \tag{51}$$

$$\nabla\cdot\mathbf{u} = 0, \tag{52}$$

where $\mathbf{u} = (u, v)$ is the velocity vector having its x- and y-components $u(x, y, t)$ and $v(x, y, t)$, respectively. Here, p is the pressure. We are interested in the Navier–Stokes equation with periodic boundary conditions, i.e. the Taylor problem, in a 2D square domain $[0, 2\pi] \times [0, 2\pi]$. The initial values for the velocity are taken as

$$\begin{aligned} u(x, y, 0) &= -\cos(kx)\sin(ky), \\ v(x, y, 0) &= \sin(kx)\cos(ky), \end{aligned} \tag{53}$$

Table 3
Errors of the numerical solutions of the wave equation

$\Delta t = 0.0005$							
N	t	L_2			L_∞		
		RSK-16	RSK-32	FPS	RSK-16	RSK-32	FPS
64	0.1	3.02E–11	3.02E–15	2.79E–15	6.73E–12	2.11E–15	1.78E–15
	1	2.72E–09	2.30E–13	2.44E–13	6.02E–10	5.31E–14	5.69E–14
	10	2.90E–08	3.59E–12	3.32E–12	6.43E–09	8.01E–13	7.43E–13
	100	3.01E–07	1.34E–09	1.34E–09	6.73E–08	2.96E–10	2.95E–10
32	0.1	3.96E–11	3.08E–15	1.87E–15	8.63E–12	1.67E–15	7.49E–16
	1	3.54E–09	2.15E–13	2.42E–13	7.72E–10	4.77E–14	5.60E–14
	10	3.78E–08	4.07E–12	3.27E–12	8.24E–09	8.93E–13	7.38E–13
	100	3.97E–07	1.36E–09	1.31E–09	8.65E–08	2.97E–10	2.95E–10
$\Delta t = 0.01$							
64	0.1	4.69E–11	3.76E–11	3.70E–11	1.04E–11	8.33E–12	8.33E–12
	1	2.51E–09	3.76E–10	3.70E–10	5.56E–10	8.33E–11	8.33E–11
	10	2.53E–08	3.76E–09	3.70E–09	5.60E–09	8.33E–10	8.33E–10
	100	2.68E–07	3.77E–08	3.71E–08	5.93E–08	8.35E–09	8.34E–09
	1000	2.67E–06	3.73E–07	3.67E–07	5.91E–07	8.26E–08	8.26E–08
32	0.1	5.34E–11	3.82E–11	3.70E–11	1.16E–11	8.33E–12	8.30E–12
	1	3.33E–09	3.82E–10	3.70E–10	7.25E–10	8.33E–11	8.33E–11
	10	3.40E–08	3.82E–09	3.70E–09	7.41E–09	8.33E–10	8.33E–10
	100	3.60E–07	3.83E–08	3.71E–08	7.84E–08	8.35E–09	8.34E–09
	1000	3.58E–06	3.79E–07	3.67E–07	7.81E–07	8.26E–08	8.26E–08

where k is the wavenumber taking an integer value. The case of $k = 1$ was treated by using the DSC algorithm [23]. In this work, we study the cases for $k \geqslant 1$ which involve higher frequency. The corresponding numerical solution is compared with the exact one

$$\begin{aligned} u(x, y, t) &= -\cos(kx)\sin(ky)\mathrm{e}^{-2k^2t/\mathrm{Re}}, \\ v(x, y, t) &= \sin(kx)\cos(ky)\mathrm{e}^{-2k^2t/\mathrm{Re}}, \\ p(x, y, t) &= -\tfrac{1}{4}\big[\cos(2kx) + \cos(2ky)\big]\mathrm{e}^{-4k^2t/\mathrm{Re}}. \end{aligned} \tag{54}$$

We adopt the Adams–Bashforth–Crank–Nicolson (ABCN) scheme [43] for the time discretization and treatment of pressure

$$\begin{aligned} &\frac{1}{2\mathrm{Re}}\nabla^2\mathbf{u}^{n+1} - \frac{1}{\Delta t}\mathbf{u}^{n+1} = \nabla p^{n+1/2} + \mathbf{S}^n, \\ &\nabla^2 p^{n+1/2} + \nabla\cdot\mathbf{S}^n = 0, \end{aligned} \tag{55}$$

where the source vector $\mathbf{S}^n$ is given by

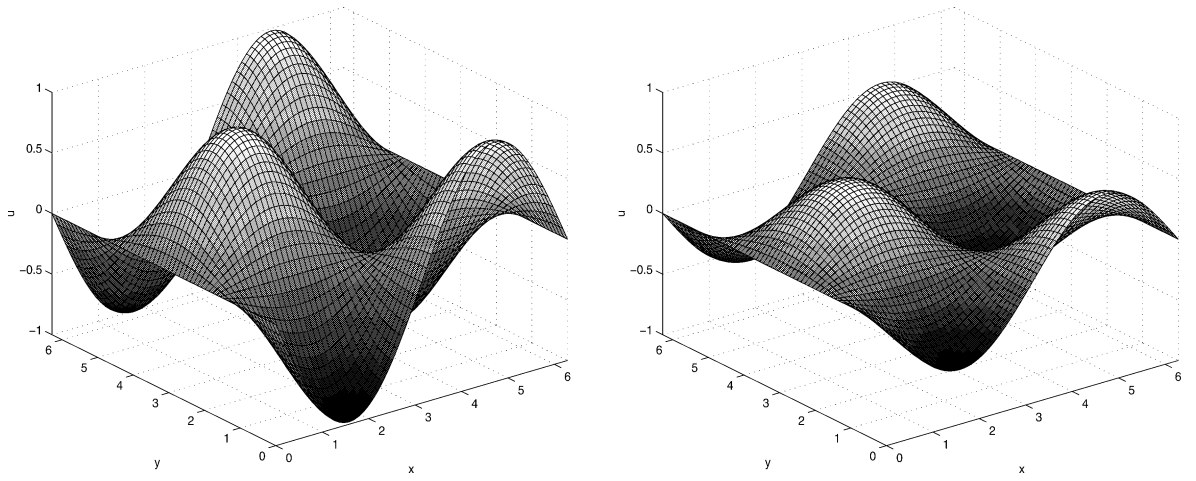

Fig. 6. The solutions of the NS equation ($k = 1$ and $\mathrm{Re} = 100$). Left: $t = 1$; right $t = 30$.

$$\mathbf{S}^n = -\frac{\mathbf{u}^n}{\Delta t} + \frac{3(\mathbf{u}^n \cdot \nabla)\mathbf{u}^n - (\mathbf{u}^{n-1} \cdot \nabla)\mathbf{u}^{n-1}}{2} - \frac{1}{2\mathrm{Re}}\nabla^2 \mathbf{u}^n. \tag{56}$$

The aspect of ensuring divergence-free velocity field in the scheme was discussed by Le Queéré and Alziary de Roquefort [44]. The error expression, L_∞, is calculated to evaluate the accuracy of both numerical methods as in the previous two numerical problems. Our calculations are conducted for a variety of Reynolds number Re, k and time increment.

3.3.2. Numerical experiments

Two algorithms, the DSC and the FPS, are applied to solve the NS equation for different values of k from 1 to 5. In each case, Reynolds number Re varying from 100 to 10^{10} is used in computation. We also compare the two algorithms for different time increment Δt and spatial discretization N. Here, for simplicity, we set equal spatial discretization in x- and y-directions. Although N can be set to any integer values for the DSC algorithm, for the purpose of comparing with the FPS in which N is required to be powers of 2, we choose $N = 32$ in our computations. Numerical solutions to the NS equation are obtained for a long time. In the DSC algorithm, we have one more parameter M which determines the support size of DSC kernels. The value of M is usually smaller than N, therefore we choose $M = 16$ or 32. The corresponding DSC algorithm are denoted by the RSK-16 and RSK-32, respectively.

The solutions of the present problem are of $\sin(kx)$ and $\cos(kx)$ type, which are the basis functions of Fourier pseudospectral methods. The periodic boundary conditions are also suitable for the FPS method. Therefore, it is expected that the FPS should achieve the highest possible accuracy in this case.

We first study the case of $k = 1$ in Eq. (55). The solutions $u(x, y, t)$ (55) of the NS equation at time $t = 1$ and $t = 30$ are depicted in Fig. 6. The L_∞ errors of the numerical results obtained using the DSC and FPS methods for time up to 30 are listed in Table 4. The number of grid points is set as $N = 32$ in each dimension, and the results obtained by using the FPS is compared with those of the RSK-16 and RSK-32.

For the case of $\mathrm{Re} = 10^{10}$, it is observed in Table 4 that the accuracy of the DSC algorithm depends on the choice of support size M, as the DSC algorithm is a local method. We have shown again that the accuracy of the DSC algorithm is controllable. The DSC algorithm becomes more accurate as the support size M increases from 16 to 32. It is interesting to note that the RSK-32 is even more accurate than the FPS method for both time increments. In the same table, the computations are shown for smaller Reynolds numbers. For two intermediate Reynolds numbers, the FPS method performs slightly better than the DSC algorithm, as in the same table. In

Table 4
L_∞ error of the numerical solutions for the NS equation with $k = 1$

Re	t	$\Delta t = 0.001$			$\Delta t = 0.01$		
		RSK-16	RSK-32	FPS	RSK-16	RSK-32	FPS
10^2	1	1.27E–07	1.80E–12	1.95E–10	1.26E–07	6.53E–11	1.95E–08
	10	1.06E–06	1.03E–11	1.52E–10	1.01E–06	5.46E–10	1.58E–08
	20	1.74E–06	1.42E–11	1.15E–10	1.64E–06	8.94E–10	1.25E–08
	30	2.11E–06	1.54E–11	8.68E–11	1.99E–06	1.08E–09	1.01E–08
10^4	1	2.91E–09	1.37E–12	7.91E–13	3.51E–08	2.27E–13	1.93E–12
	10	2.02E–08	2.84E–11	7.59E–12	2.67E–07	5.13E–12	1.30E–12
	20	4.01E–08	1.11E–10	1.48E–11	5.31E–07	2.38E–11	6.87E–13
	30	5.77E–08	2.05E–10	2.23E–11	7.65E–07	7.39E–11	5.68E–13
10^6	1	3.58E–09	7.38E–13	9.02E–13	3.57E–08	1.53E–13	8.89E–14
	10	2.75E–08	1.18E–11	8.65E–12	2.76E–07	4.00E–12	8.46E–13
	20	5.51E–08	5.85E–11	1.70E–11	5.52E–07	3.80E–11	1.66E–12
	30	7.99E–08	1.38E–10	2.56E–11	8.01E–07	1.70E–10	2.46E–12
10^{10}	1	3.59E–09	2.96E–13	9.9-E–13	3.57E–08	2.75E–14	9.90E–14
	10	2.76E–08	2.18E–12	9.46E–12	2.76E–07	2.49E–13	9.42E–13
	20	5.52E–08	4.47E–12	1.86E–11	5.52E–07	5.68E–13	1.83E–12
	30	8.01E–08	6.50E–12	2.79E–11	8.28E–07	9.28E–13	2.70E–12

these three cases, the results obtained from $\Delta t = 0.01$ are generally better than those from $\Delta t = 0.001$ for both the RSK-32 and the FPS. This phenomenon follows naturally from the fact that at a high Reynolds number, the solution does not vary much with time. Therefore, the numerical solution is not so sensitive to the size of the time increment Δt. A larger time increment implies a smaller number of iterations to reach a given total time, which leads to higher accuracy. When the Reynolds number is reduced to 100, as expected, a smaller time increment yields higher accuracy for both methods. In this case, the DSC algorithm outperforms the FPS method again. It is obvious that both methods perform extremely well for this case.

We next consider the case of $k = 2$. The solutions $u(x, y, t)$ (55) of the NS equation at time $t = 1$ and $t = 30$ are plotted in Fig. 7. The L_∞ errors obtained by both the FPS and DSC are given in Table 5. Similar to the previous case, the numerical study is performed at two different time increments $\Delta t = 0.001$ and $\Delta t = 0.01$ for four Reynolds numbers $\mathrm{Re} = 10^{10}$, 10^6, 10^4 and 100.

We observe that, for small value of t ($t < 10$), both the RSK-32 and the FPS give similar order of accuracy. However, when t is large and approaches 30, the DSC scheme gives much better results than the FPS does. This happens for both time increments $\Delta t = 0.001$ and 0.01. For example, The DSC algorithm performs about a 1000 times better than the FPS at $t = 30$. At a low Reynolds number, $\mathrm{Re} = 100$, the DSC algorithm still performs 100 times better than the FPS method at early times. However, both the FPS and the DSC give the same level of accuracy at a late time $t = 30$. It is noted that the factor $\mathrm{e}^{-2k^2 t/\mathrm{Re}}$ in the analytical solution (55) becomes very small for large k and/or t, and small Re. Therefore, the small errors in both methods are in part due to the small magnitude of the exact solutions of the NS equation.

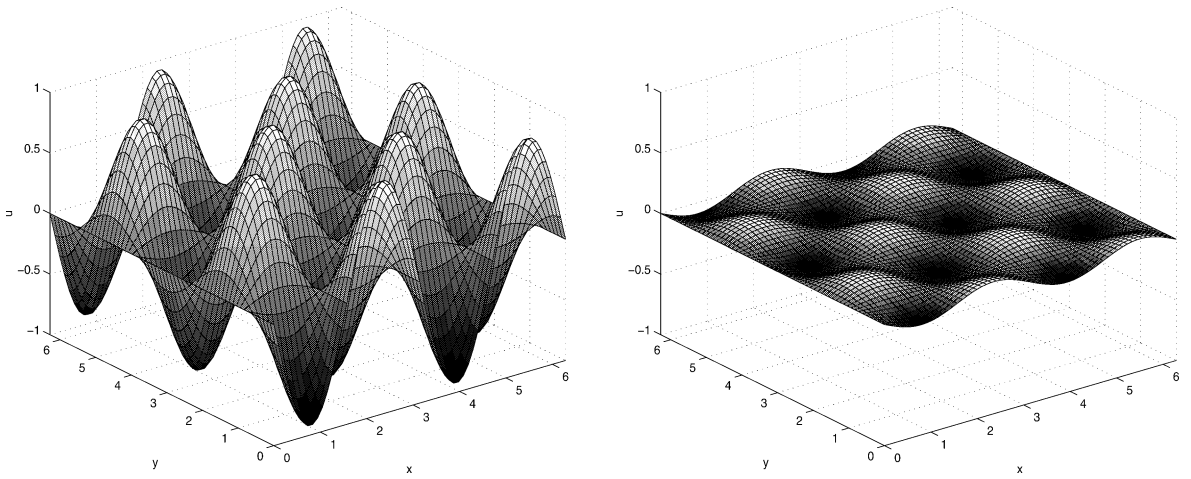

Fig. 7. The solutions of the NS equation with $k = 2$ and $\mathrm{Re} = 100$. The left plot is the solution at time $t = 1$. The right plot is the one at time $t = 30$.

Table 5
L_∞ error of the numerical solutions for the NS equation with $k = 2$

Re	t	$\Delta t = 0.001$		$\Delta t = 0.01$	
		RSK-32	FPS	RSK-32	FPS
10^2	1	4.01E–11	2.91E–09	3.93E–09	2.91E–07
	10	1.93E–10	1.24E–09	1.91E–08	1.25E–07
	20	1.74E–10	4.72E–10	1.72E–08	4.74E–08
	30	1.24E–10	1.73E–10	1.21E–08	1.74E–08
10^4	1	9.32E–13	3.33E–13	1.82E–13	3.19E–11
	10	4.51E–11	6.64E–12	6.78E–12	3.13E–11
	20	1.39E–09	4.41E–09	4.50E–10	5.91E–10
	30	5.99E–08	4.07E–06	3.12E–08	9.55E–07
10^6	1	5.07E–13	5.28E–13	8.59E–14	5.01E–14
	10	1.76E–11	7.36E–12	5.36E–12	1.34E–12
	20	3.99E–10	3.80E–09	3.45E–10	1.38E–09
	30	1.71E–08	4.99E–06	2.38E–08	3.17E–06
10^{10}	1	4.20E–13	5.94E–13	8.63E–14	3.97E–14
	10	5.09E–12	6.65E–12	1.02E–12	1.14E–12
	20	6.41E–11	4.58E–09	1.61E–11	1.05E–09
	30	2.33E–09	6.99E–06	1.00E–09	1.42E–06

Finally, we consider the case of $k = 5$. Obviously, this case requires very high spatial accuracy (or Fourier resolution). In all the previous cases, the parameter $\sigma = 3.2\Delta$ is used as it gives very accurate approximations. However, in the present case, a large σ value is required to take better care of the high frequency component.

Table 6
L_∞ error of the numerical solutions for the NS equation with $k = 5$

Re	t	$\Delta t = 0.001$		$\Delta t = 0.01$	
		RSK-32	FPS	RSK-32	FPS
10^2	1	6.53E–09	6.95E–08	6.32E–07	6.93E–06
	10	7.25E–10	1.40E–10	7.02E–08	1.38E–08
	20	9.79E–12	3.79E–12	9.46E–10	3.80E–10
10^4	1	4.25E–11	1.18E–11	4.01E–10	1.24E–09
	10	1.18E–06	1.38E–06	1.41E–06	1.41E–07
10^6	1	4.10E–11	1.15E–12	4.06E–10	7.32E–14
	10	7.37E–07	1.86E–06	2.16E–06	5.83E–07
10^{10}	1	4.07E–11	1.02E–12	4.06E–10	7.32E–14
	10	4.95E–07	2.49E–06	2.11E–06	5.83E–07

Therefore, we choose $\sigma = 5.4\Delta$. When $\sigma = 5.4\Delta$, though the DSC algorithm approximates higher frequency part better, it is at the sacrifice of the accuracy for approximating low frequency components as seen from Fig. 2. In Table 6, we compare the accuracy of the RSK-32 and the FPS at $k = 5$. At high Reynolds numbers, the system is dominated by the nonlinear advection. The accuracy of both methods degrades quite fast. It is seen that the FPS method performs slightly better than the DSC algorithm. It also can be observed that, for small Reynolds number, the DSC algorithm can compete with the FPS method.

As a short summary of this section, the numerical study on the NS equation with periodic boundary condition using both the FPS and DSC methods shows that the DSC can compete with the FPS method in terms of accuracy, though the problem is most suitable for the implementation of the FPS method. This point has been illustrated by using various choices of time increment Δt, Reynolds number Re and wavenumber k for arbitrarily long time evolution t. Moreover, we have demonstrated that the accuracy of the DSC algorithm is controllable.

4. Conclusion

In this work, we explore the utility, test accuracy and examine the limitation of the discrete singular convolution (DSC) algorithm for solving three types of partial differential equations (PDEs). Particular attention has been given to a detailed comparison of the DSC algorithm and the Fourier pseudospectral (FPS) method for computational accuracy and stability. To this end, three two-dimensional PDEs of different nature, the heat equation, the wave equation and the Navier–Stokes equation are solved by using both methods with standard temporal discretizations, i.e. the fourth-order Runge–Kutta or Crank–Nicolson scheme. The numerical resolution of the DSC algorithm for approximating the spatial derivatives is analyzed by using the discrete Fourier analysis. Extensive numerical experiments are conducted at a variety of time increment, grid spacing, wavenumber, and Reynolds number (for the Navier–Stokes equation) to give a comprehensive comparison between the two methods. The substantial numerical results lead to convincing conclusions which are summarized as follows:

(1) There is no doubt that the FPS method gives the most accurate spatial discretization for problems involving bandlimited periodic L^2 functions. The DSC algorithm is not as accurate as the FPS method for the high frequency component of such functions. However, DSC can perform better than FPS method in many other aspects.

(2) The accuracy of the DSC algorithm is controllable by adjusting its support parameter M. At a large M, the DSC algorithm can be even more accurate than the FPS method for approximating non-bandlimited functions, such as the Gaussian solution to the heat equation, and can be nearly as accurate as the FPS method for bandlimited periodic functions, such as those in the wave equation and the Taylor problem.
(3) The DSC algorithm can be used as both local and global methods depending on the choice of the parameter M. As a result, it can be used for treating complex geometry and boundary conditions, whereas, the FPS method is a global method and is generally restricted to treating periodic boundary conditions. The DSC algorithm has been proved useful in treating complex boundary conditions in structural analysis [22, 29,31] and complex geometries in handling flow simulations [25].
(4) The number of grid points is limited to the power of 2 in the FPS method. There is no such a limitation in the DSC algorithm.
(5) The FPS method in association with the FFT algorithm, is faster than the DSC algorithm when the grid is not very large. For approximating a derivative, the operation of the FFT is proportional to $N \log N$ (here N is the total number of grid points), whereas, the operation for DSC algorithm is $N(2M + 1)$. We found that when $N = 512$, the DSC algorithm becomes faster than the FPS method.

Acknowledgement

This work was supported in part by the National University of Singapore.

References

[1] C. Lanczos, Trigonometric interpolation of empirical and analytical functions, J. Math. Phys. 17 (1938) 123–199.
[2] J.W. Cooley, J.W. Tukey, An algorithm for the machine calculation of complex Fourier series, Math. Comput. 19 (1965) 297–301.
[3] B.A. Finlayson, L.E. Scriven, The method of weighted residuals—a review, Appl. Mech. Rev. 19 (1966) 735–748.
[4] S.A. Orszag, Comparison of pseudospectral and spectral approximations, Stud. Appl. Math. 51 (1972) 253–259.
[5] C. Canuto, M.Y. Hussaini, A. Quarteroni, T.A. Zang, Spectral Methods in Fluid Dynamics, Springer, Berlin, 1988.
[6] B. Fornberg, A Practical Guide to Pseudospectral Methods, Cambridge University Press, Cambridge, 1996.
[7] L.N. Trefethen, Spectral Methods in Matlab, Oxford University, Oxford, England, 2000.
[8] G.E. Forsythe, W.R. Wasow, Finite-Difference Methods for Partial Differential Equations, Wiley, New York, 1960.
[9] E. Isaacson, H.B. Keller, Analysis of Numerical Methods, Wiley, New York, 1966.
[10] O.C. Zienkiewicz, The Finite Element Method in Engineering Science, McGraw-Hill, London, 1971.
[11] C.S. Desai, J.F. Abel, Introduction to the Finite Element Methods, Van Nostrand Reinhold, New York, 1972.
[12] J.T. Oden, The Finite Elements of Nonlinear Continua, McGraw-Hill, New York, 1972.
[13] B. Nath, Fundamentals of Finite Elements for Engineers, Athlone Press, London, 1974.
[14] R.T. Fenner, Finite Element Methods for Engineers, Imperial College Press, London, 1975.
[15] Y.K. Cheung, Finite Strip Methods in Structural Analysis, Pergamon Press, Oxford, 1976.
[16] S.S. Rao, The Finite Element Method in Engineering, Pergamon Press, New York, 1982.
[17] J.N. Reddy, Energy and Variational Methods in Applied Mechanics, John Wiley, New York, 1984.
[18] G.W. Wei, Discrete singular convolution for the Fokker–Planck equation, J. Chem. Phys. 110 (1999) 8930–8942.
[19] G.W. Wei, A unified approach for solving the Fokker–Planck equation, J. Phys. A 33 (2000) 4935–4953.
[20] G.W. Wei, Wavelets generated by using discrete singular convolution kernels, J. Phys. A 33 (2000) 8577–8596.
[21] G.W. Wei, Solving quantum eigenvalue problems by discrete singular convolution, J. Phys. B 33 (2000) 343–352.
[22] G.W. Wei, Vibration analysis by discrete singular convolution, J. Sound Vibration 244 (2001) 535–553.
[23] G.W. Wei, A new algorithm for solving some mechanical problems, Comput. Methods Appl. Mech. Engrg. 190 (2001) 2017–2030.
[24] G.W. Wei, A unified method for computational mechanics, in: C.M. Wang, K.H. Lee, K.K. Ang (Eds.), Computational Mechanics for the Next Millennium, Elsevier, New York, 1999, pp. 1049–1054.
[25] D.C. Wan, G.W. Wei, Discrete singular convolution-finite subdomain method for the solution of incompressible viscous flows, J. Comput. Phys., in press.
[26] S. Guan, C.-H. Lai, G.W. Wei, Bessel–Fourier analysis of patterns in a circular domain, Physica D 151 (2001) 83–98.
[27] G.W. Wei, Synchronization of single-side averaged coupling and its application to shock capturing, Phys. Rev. Lett. 86 (2001) 3542–3545.

[28] G.W. Wei, Discrete singular convolution method for the sine-Gordon equation, Physica D 137 (2000) 247–259.
[29] G.W. Wei, Discrete singular convolution for beam analysis, Engrg. Structures 23 (2001) 1045–1053.
[30] G.W. Wei, Y.B. Zhao, Y. Xiang, The determination of the natural frequencies of rectangular plates with mixed boundary conditions by discrete singular convolution, Int. J. Mech. Sci. 43 (2001) 1731–1746.
[31] Y.B. Zhao, G.W. Wei, Y. Xiang, Plate vibration under irregular internal supports, Int. J. Solids Structures, in press.
[32] G.W. Wei, Y.B. Zhao, Y. Xiang, Discrete singular convolution and its application to the analysis of plates with internal supports. I. Theory and algorithm, Int. J. Numer. Methods Engrg., in press.
[33] L. Schwartz, Theore des Distribution, Hermann, Paris, 1951.
[34] G.W. Wei, Quasi wavelets and quasi interpolating wavelets, Chem. Phys. Lett. 296 (1998) 215–222.
[35] G.G. Walter, J. Blum, Probability density estimation using delta sequences, Ann. Statist. 7 (1979) 328–340.
[36] B.A. Finlayson, The Method of Weighted Residuals and Variational Principles, Academic, New York, 1972.
[37] J.O. Dow, A Unified Approach to the Finite Element Method and Error Analysis Procedures, Academic, San Diego, CA, 1999.
[38] B.F. Sanders, N.O. Katoposes, J.P. Boyd, Spectral modeling of nonlinear dispersive waves, J. Hydraulic. Engrg. ASCE 124 (1998) 2–12.
[39] H.-O. Kreiss, J. Oliger, Comparison of accurate methods for the integration of hyperbolic systems, Tellus 24 (1972) 199–215.
[40] K.V. Roberts, N.O. Weiss, Math. Comput. 20 (1966) 272.
[41] B. Swartz, B. Wendroff, The relation between the Galerkin and collocation methods using smooth splines, SIAM J. Numer. Anal. 11 (1974) 994.
[42] R. Vichnevetsky, J.B. Bowles, Fourier Analysis of Numerical Approximations of Hyperbolic Equations, SIAM, Philadelphia, 1982.
[43] H.H. Yang, B. Shizgal, Chebyshev pseudospectral multi-domain technique for viscous flow calculation, Comput. Methods Appl. Mech. Engrg. 118 (1994) 47.
[44] P. Le Quéré, T. Alziary de Roquefort, Computation of natural convection in two-dimension cavities, J. Comput. Phys. 57 (1985) 210.