Journal of Computational Physics 229 (2010) 2287–2313

Contents lists available at ScienceDirect

Journal of Computational Physics

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

Spectrally-accurate algorithm for moving boundary problems for the Navier–Stokes equations

S.Z. Husain, J.M. Floryan *

Department of Mechanical and Material Engineering The University of Western Ontario London, Ontario, Canada N6A 5B9

ARTICLE INFO

Article history:
Received 18 April 2009
Received in revised form 15 September 2009
Accepted 26 November 2009
Available online 11 December 2009

Keywords:
Spectral methods
Moving boundary problems
Immersed boundary conditions

ABSTRACT

A spectral algorithm based on the immersed boundary conditions (IBC) concept is developed for simulations of viscous flows with moving boundaries. The algorithm uses a fixed computational domain with flow domain immersed inside the computational domain. Boundary conditions along the edges of the time-dependent flow domain enter the algorithm in the form of internal constraints. Spectral spatial discretization uses Fourier expansions in the stream-wise direction and Chebyshev expansions in the normal-to-the-wall direction. Up to fourth-order implicit temporal discretization methods have been implemented. It has been demonstrated that the algorithm delivers the theoretically predicted accuracy in both time and space. Performances of various linear solvers employed in the solution process have been evaluated and a new class of solver that takes advantage of the structure of the coefficient matrix has been proposed. The new solver results in a significant acceleration of computations as well as in a substantial reduction in memory requirements.

1. Introduction

"Moving boundary problems" refer to situations where the boundaries of the flow domain change locations as a function of time in a known and well prescribed manner. Such problems attracted attention of researchers from various fields for many years and the relevant algorithms have been pursued with considerable interest particularly in the field of biomedical engineering. Practical examples include peristaltic and pulsatile flows that define the flows in the esophagus and flows through the vasculatures due to cardiac actions. Various available algorithms can be classified mainly as Lagrangian and Eulerian [1]. Mixed methods, that are combinations of the Lagrangian and Eulerian techniques, have also been pursued [1].

In algorithms based on the Lagrangian concepts, each fluid element is followed individually resulting in a need for a coordinate system that moves with the fluid. Mesh tangling and associated loss of numerical accuracy poses significant restrictions on the overall applicability of these methods [1].

The Eulerian algorithms rely on coordinate systems that are stationary in a laboratory frame of reference or may move in a prescribed manner. Such algorithms can be divided for convenience into fixed grid, adaptive grid and various mapping methods.

In the fixed grid methods, the grid is fixed in the solution domain and the locations of the moving boundaries are tracked using either surface [2] or volume tracking procedures [3]. The surface tracking relies on a set of points whose motion is tracked during the solution process allowing precise identification of the boundary locations; these boundaries are represented as a set of interpolated curves [3,4]. The volume tracking algorithms on the other hand work by reconstructing

* Corresponding author. Tel.: +1 519 661 2130; fax: +1 519 661 3020.
E-mail address: mfloryan@eng.uwo.ca (J.M. Floryan).

0021-9991/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2009.11.035

the boundary whenever necessary instead of storing the boundary locations. The presence of a convenient marker within a computational cell and its quantity form the basis of the various reconstruction methodologies. Different versions of volume tracking algorithms exist, e.g., VOF (Volume of Fluid) [5], MAC (Marker and Cell) [6] and Level Set [7,8] methods. These methods are based on the standard spatial discretization schemes with low-order of accuracy for the field equations, which are consistent with the diffused boundary locations resulting from the boundary reconstruction processes.

The adaptive grid methods use numerical mappings to adjust the grids at each time step so that one of the grid lines always overlaps with the boundary location. The computational costs of these methods are very high due to the requirement of grid reconstruction at each time step. For example, the grid construction process contributed to about 75% of the total computational cost for the problem discussed in [9]. The choice of spatial discretization technique has a smaller effect on the overall computational costs. The requirement of high accuracy in solution may lead to numerous challenges as the total error has contributions from the error in the grid generation as well as from the error due to spatial and temporal discretizations of the field equations.

Analytical mapping of the irregular physical domain into a rectangular computational domain can help in improving the accuracy at the cost of increased complexity of the field equations [10,18]. However, such mappings are available only for a limited class of geometries [1] and reconstruction of the coefficient matrix during each time step can add to the overall computational cost by a substantial margin [19].

Mixed Lagrangian–Eulerian methods rely on the combination of the concepts described above [1].

The increase of accuracy while maintaining computational efficiency poses a significant challenge for any algorithm to be developed for moving boundary problems. One of the new concepts involves the use of the immersed or fictitious boundaries. This concept was first proposed by Peskin [11] in the context of cardiac dynamics and its various variants have been reviewed in [12,13]. The common limitation is the spatial accuracy, as most of these methods are based on the low-order finite-difference, finite-volume or finite-element technique [13–16]. The second, less known limitation is associated with the use of the local fictitious forces required to enforce the no-slip and no-penetration conditions. These forces locally affect the flow physics and this may lead to the incorrect estimates of derivatives of flow quantities, i.e., misrepresentation of the local wall shear. It is difficult to estimate the potential error associated with such procedures, but it is known that in the case of hydrodynamic instabilities the second derivative of mean flow plays a very strong role in determining the flow response, and this derivative may not be predicted with sufficient accuracy using a procedure that changes flow physics (even locally). This problem is likely to be more pronounced in the case of methods with higher spatial accuracy. Sharp interface method as presented in [17] also uses Cartesian grid to enforce boundary conditions along the immersed boundaries that may be either stationary or moving. The method relies on the level set description of the immersed interface along with finite-difference discretization and special treatment for points close to the interface resulting in a sharp resolution of the interface. This method requires modest programming effort and retains low spatial accuracy, similar to other immersed boundary methods discussed earlier.

Spectral methods provide the lowest error for spatial discretization of the field equations but are generally limited to solution domains with regular geometries. The first spectrally accurate implementation of the immersed boundary concept was developed by Szumbarski and Floryan in the context of a fixed boundary problem [18] and is referred to as the immersed boundary conditions (IBC) method in the rest of this discussion. This method does not use any fictitious boundaries or fictitious forces but relies on a purely formal construction of boundary constraints in order to generate the required closing relations. The method is analogous to the fixed grid Eulerian methods. Spatial discretization relies on the spectral expansions based on the Fourier and Chebyshev expansions in the stream-wise and normal-to-the-wall directions, respectively, and thus provides ability to reach machine level accuracy. The construction of boundary constraints relies on the representation of the physical boundaries in the spectral space and nullifying the relevant Fourier modes. Such implementation is limited to geometries that can be represented by Fourier expansions but results in a gridless algorithm as all possible variations of boundary geometries are described in terms of the Fourier coefficients only. The programming effort associated with modeling the changes of geometry has been essentially removed as the only information required for specifying the new geometry is reduced to a set of Fourier coefficients provided as an input to the code. The additional attractiveness of this concept is associated with the precise mathematical formalism, high accuracy and sharp identification of the location of time-dependent physical boundaries. The method has been implemented to study problems involving hydrodynamic instabilities induced by surface roughness [19,20] and has been successfully extended to two-dimensional unsteady problems [21] as well as moving boundary problems involving Laplace and biharmonic operators [22,23]. Extension of the IBC algorithm to three-dimensional problems is fairly simple. The flow-field needs to be assumed to be periodic in two spatial dimensions, which are discretized using Fourier expansions, while the remaining aperiodic dimension is discretized using Chebyshev expansions. The boundary geometries/motions can be modeled using Fourier expansions in the periodic dimensions and the enforcement of the boundary conditions would follow the concept of the IBC algorithm as presented in the case of two-dimensional problems.

The present work has two goals. The first one deals with the extension of the IBC algorithm to moving boundary problems described by the two-dimensional Navier–Stokes equations. The second one deals with the question of efficient implementation of the algorithm.

The paper is organized as follows: Section 2 provides description of the model problem used for presentation of the algorithm. Section 3 describes numerical implementation of the flow boundary conditions. Section 4 discusses performance of the algorithm using carefully selected test problems. In particular, Section 4.1 describes peristaltic flow problem while

Section 4.2 considers pulsatile flow problem. Section 4.3 provides a brief discussion of issues associated with different temporal discretization schemes. Section 4.4 discusses issues related to the efficient implementation of the algorithm. Section 5 provides a short summary of the main conclusions.

2. Problem formulation

The problem to be investigated consists of an incompressible unsteady viscous flow in a conduit bounded by moving walls (see Fig. 1) whose shapes and motions are described by the following relations

$$y_L(x,t) = -1 + \sum_{n=-\infty}^{n=+\infty} H_L^{(n)}(t) e^{in\alpha x}, \quad y_U(x,t) = 1 + \sum_{n=-\infty}^{n=+\infty} H_U^{(n)}(t) e^{in\alpha x}, \tag{2.1a, b}$$

where L and U stand for lower and upper walls, respectively. Here $H_L^{(n)} = H_L^{(-n)*}$ and $H_U^{(n)} = H_U^{(-n)*}$ are known and stars denote complex conjugates. At any instant of time, the conduit extends to $\pm\infty$ in the x-direction and its geometry remain periodic with the wavelength $\lambda = 2\pi/\alpha$.

The reference flow, i.e., steady flow through a straight conduit, is the Poiseuille flow with the velocity and pressure fields in the form

$$u_0(x,y) \equiv u_0(y) = 1 - y^2, \quad v_0(x,y) = 0, \quad p_0(x,y) \equiv p_0(x) = -2x/Re, \tag{2.1}$$

where the motion of the fluid is in the positive x-direction, the Reynolds number Re is based on the half of the original conduit height L and the maximum velocity in the x-direction U_{max}, and the flow is driven by a constant negative pressure gradient. The pressure and time scales have the form $\rho\upsilon U_{max}/L$ and L^2/υ, respectively, where υ stands for the kinematic viscosity and ρ denotes the density of the fluid. Introduction of the wall motions induces flow modifications and thus the total flow quantities can be expressed as

$$u(x,y,t) = u_0(y) + u_1(x,y,t), \quad v(x,y,t) = v_1(x,y,t), \quad p(x,y,t) = p_0(x) + p_1(x,y,t), \tag{2.3}$$

where u, v and p denote the total velocities and pressure, and u_1, v_1 and p_1 denote velocity and pressure modifications induced by the boundary motions. Substitution of (2.3) into the Navier–Stokes and continuity equations result in the following form of governing equations

$$\partial_t u_1 + Re(u_0\partial_x u_1 + u_1\partial_x u_1 + v_1 D u_0 + v_1\partial_y u_1) = -\partial_x p_1 + \nabla^2 u_1, \tag{2.4a}$$

$$\partial_t v_1 + Re(u_0\partial_x v_1 + u_1\partial_x v_1 + v_1\partial_y v_1) = -\partial_y p_1 + \nabla^2 v_1, \tag{2.4b}$$

$$\partial_x u_1 + \partial_y v_1 = 0, \tag{2.4c}$$

where $D \equiv d/dy$, $\nabla^2 = \partial_{xx} + \partial_{yy}$ is the Laplacian and the symbol ∂ denotes partial differentiation with subscripts x, y and t denoting the argument of the differentiation. The flow problem can be posed either for the complete flow quantities (u, v, p) or for the flow modifications (u_1, v_1, p_1). The latter approach was selected for the present work as it results in numerically smaller nonlinear terms thereby providing faster convergence when iterative solution processes are employed.

The problem formulation needs to be supplemented with suitable initial and boundary conditions. The initial conditions are taken to be in the form

$$u(x,y,0) = u_i(x,y), \quad v(x,y,0) = v_i(x,y), \quad y_L(x,0) = y_{Li}(x), \quad y_U(x,0) = y_{Ui}(x), \tag{2.5a-b}$$

where $u_i(x,y)$, $v_i(x,y)$, $y_{Li}(x)$, $y_{Ui}(x)$ are considered to be known, and the boundary conditions at the solid walls are given by

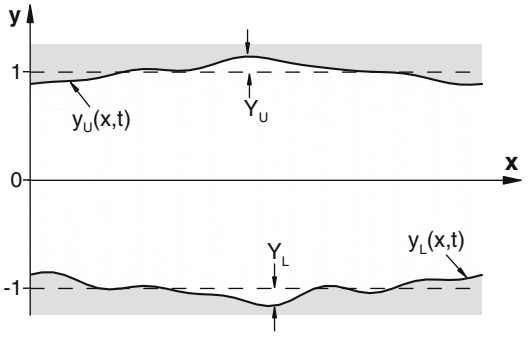

Fig. 1. Sketch of the instantaneous form of the flow domain.

$$u_0(y_L(x,t)) + u_1(x,y_L(x,t),t) = u_L(x,t) = 0, \tag{2.6a}$$
$$u_0(y_U(x,t)) + u_1(x,y_U(x,t),t) = u_U(x,t) = 0, \tag{2.6b}$$
$$v_1(x,y_L(x,t),t) = v_L(x,t) = (y_L)' = \sum_{n=-\infty}^{n=+\infty} (H_L^{(n)})' e^{in\alpha x}, \tag{2.6c}$$
$$v_1(x,y_U(x,t),t) = v_U(x,t) = (y_U)' = \sum_{n=-\infty}^{n=+\infty} (H_U^{(n)})' e^{in\alpha x}, \tag{2.6d}$$

where the '′' sign denotes derivative with respect to time.

Introduction of the stream function Ψ and elimination of pressure lead to a single field equation in terms of the unknown modification of stream function Ψ_1 of the form

$$Re^{-1}\partial_t[\nabla^2\Psi_1] - Re^{-1}\nabla^2(\nabla^2\Psi_1) + [u_0\partial_x(\nabla^2) - D^2u_0\partial_x]\Psi_1 = -\partial_y[\partial_x(u_1u_1) + \partial_y(u_1v_1)] + \partial_x[\partial_x(u_1v_1) + \partial_y(v_1v_1)], \tag{2.7}$$

where

$$u(x,y,t) = u_0(y) + u_1(x,y,t) = D\Psi_0 + \partial_y\Psi_1 = \partial_y\Psi_T, \tag{2.8a}$$
$$v(x,y,t) = v_1(x,y,t) = -\partial_x\Psi_1 = -\partial_x\Psi_T \tag{2.8b}$$

where $\Psi_0 = -y^3/3 + y + 2/3$ denotes the stream function of this flow and Ψ_T stand for the stream function of the complete flow (i.e., the total stream function). The boundary conditions in terms of the stream function thus take the following form

$$\partial_y\Psi_1(x,y_L(x,t),t) = -D\Psi_0(y_L(x,t)), \tag{2.9a}$$
$$\partial_y\Psi_1(x,y_U(x,t),t) = -D\Psi_0(y_U(x,t)), \tag{2.9b}$$
$$\partial_x\Psi_1(x,y_L(x,t),t) = -\sum_{n=-\infty}^{n=+\infty} (H_L^{(n)})' e^{in\alpha x}, \tag{2.9c}$$
$$\partial_x\Psi_1(x,y_U(x,t),t) = -\sum_{n=-\infty}^{n=+\infty} (H_U^{(n)})' e^{in\alpha x}. \tag{2.9d}$$

We are interested in the determination of solution of the flow problem described by Eqs. (2.7)–(2.9) with the spectral accuracy in space and the desired accuracy in time. The main difficulty associated with the implementation of the spectral discretization in the spatial dimensions arises due to the irregularity and time-dependence of the solution domain.

In order to overcome problems associated with the spatial discretization, we select fixed rectangular computational domain extending over one period in the x-direction and extending sufficiently far in the y-direction so that the flow domain always remains immersed inside the computational domain during the time interval of interest. If we denote the locations of extremities of the walls as Y_U and Y_L within the time interval under investigation, then the y-extent of the computational domain is set as $(-1 - Y_L, 1 + Y_U)$ without loss of generality. The spatial discretization is based on the use of Fourier series in the x-direction due to periodicity of the geometry, and on expansions in terms of the Chebyshev polynomials in the y-direction. We shall use standard definition of the Chebyshev polynomials and thus the y-extent of the computational domain needs to be mapped onto $(-1,1)$ (see Fig. 1) before calculations can proceed. The required mapping has the form

$$\hat{y} = [y - (1 + Y_U)]\Gamma + 1, \tag{2.10}$$

where $\hat{y} \in \langle -1, 1\rangle$ and $\Gamma = 2/(2 + Y_U + Y_L)$ is a constant. Application of (2.10) transforms the governing equation into

$$Re^{-1}\partial_t[\hat{\nabla}^2\Psi_1] - Re^{-1}\hat{\nabla}^4\Psi_1 + [u_0\partial_x(\hat{\nabla}^2) - \Gamma^2\widehat{D}^2u_0\partial_x]\Psi_1 = -\Gamma\partial_{\hat{y}}[\partial_x(u_1u_1) + \partial_{\hat{y}}(u_1v_1)] + \partial_x[\partial_x(u_1v_1) + \Gamma\partial_{\hat{y}}(v_1v_1)], \tag{2.11}$$

where $\hat{\nabla}^2 = \partial_{xx} + \Gamma^2\partial_{\hat{y}\hat{y}}$ and $\widehat{D} = d/d\hat{y}$. Locations of the walls in the $(x,\hat{y})$ plane are given as

$$\hat{y}_L(x,t) = \sum_{n=-\infty}^{n=+\infty} A_L^{(n)}(t)e^{in\alpha x}, \quad \hat{y}_U(x,t) = \sum_{n=-\infty}^{n=+\infty} A_U^{(n)}(t)e^{in\alpha x}, \tag{2.12a,b}$$

where $A_L^{(0)}(t) = 1 + \Gamma[-2 - Y_U + H_L^{(0)}(t)]$, $A_L^{(n)}(t) = \Gamma H_L^{(n)}(t)$ for $n\neq 0$, $A_U^{(0)}(t) = 1 + \Gamma[-Y_U + H_U^{(0)}(t)]$, $A_U^{(n)}(t) = \Gamma H_U^{(n)}(t)$ for $n\neq 0$. The boundary conditions at the transformed boundaries become

$$\partial_{\hat{y}}\Psi_1(x,\hat{y}_L(x,t),t) = -\widehat{D}\Psi_0(\hat{y}_L(x,t)), \tag{2.13a}$$
$$\partial_{\hat{y}}\Psi_1(x,\hat{y}_U(x,t),t) = -\widehat{D}\Psi_0(\hat{y}_U(x,t)), \tag{2.13b}$$
$$\Psi_1(x,\hat{y}_L(x,t),t) = -\sum_{n=-\infty,n\neq 0}^{n=+\infty} (in\alpha\Gamma)^{-1}(A_L^{(n)})' e^{in\alpha x} + C_L(t), \tag{2.13c}$$
$$\Psi_1(x,\hat{y}_U(x,t),t) = -\sum_{n=-\infty,n\neq 0}^{n=+\infty} (in\alpha\Gamma)^{-1}(A_U^{(n)})' e^{in\alpha x} + C_U(t), \tag{2.13d}$$

where $C_U(t)$ and $C_L(t)$ are time-dependent constants resulting from integration of Eqs. (2.9c) and (2.9d), respectively.

The solution can be represented in the form of Fourier expansion

$$\Psi_1(x,\hat{y},t)=\sum_{n=-\infty}^{n=+\infty}\Phi^{(n)}(\hat{y},t)e^{in\alpha x}\approx\sum_{n=-N_M}^{n=+N_M}\Phi^{(n)}(\hat{y},t)e^{in\alpha x}, \tag{2.14}$$

where $\Phi^{(n)}(\hat{y},t)=\Phi^{(-n)^*}(\hat{y},t)$. Substitution of (2.14) into Eq. (2.11) and separation of Fourier components lead to a system of partial differential equations for $\Phi^{(n)},\ n\in\langle 0,N_M\rangle$, of the type

$$\begin{aligned}&Re^{-1}\partial_t[\Gamma^2\widehat{D}^2-(n\alpha)^2]\Phi^{(n)}-Re^{-1}[\Gamma^4\widehat{D}^4-2\Gamma^2(n\alpha)^2\widehat{D}^2+(n\alpha)^4]\Phi^{(n)}+[in\alpha\Gamma^2u_0\widehat{D}^2-i(n\alpha)^3u_0-in\alpha\Gamma^2\widehat{D}^2u_0]\Phi^{(n)}\\&\quad=-in\alpha\Gamma\widehat{D}(R_1)^{(n)}-\Gamma^2\widehat{D}^2(R_2)^{(n)}-(n\alpha)^2(R_2)^{(n)}+in\alpha\Gamma\widehat{D}(R_3)^{(n)},\end{aligned} \tag{2.15}$$

where the nonlinear velocity products are periodic and thus are expressed in terms of the relevant Fourier expansion in the form

$$[(u_1u_1),(u_1v_1),(v_1v_1)](x,\hat{y},t)=\sum_{n=-N_M}^{n=+N_M}[(R_1)^{(n)},(R_2)^{(n)},(R_3)^{(n)}](\hat{y},t)e^{in\alpha x}. \tag{2.16}$$

Four types of temporal discretizations have been used. The third-order, implicit, backward-difference method results in the following relations

$$\begin{aligned}&\Big[-Re^{-1}\Gamma^4\widehat{D}^4+\Big(11\Gamma^2Re^{-1}\Delta t^{-1}/6+2n^2\alpha^2\Gamma^2Re^{-1}+in\alpha\Gamma^2u_0\Big)\widehat{D}^2\\&\quad+\Big(-n^4\alpha^4Re^{-1}-11n^2\alpha^2\Gamma^2\Delta t^{-1}Re^{-1}/6-in\alpha\Gamma^2\widehat{D}^2u_0-in^3\alpha^3u_0\Big)\Big]\Big(\Phi^{(n)}_{\tau+1}\Big)^{J+1}\\&\quad=\Big[-in\alpha\Gamma\widehat{D}(R_1)^{(n)}_{\tau+1}-\Gamma^2\widehat{D}^2(R_2)^{(n)}_{\tau+1}-(n\alpha)^2(R_2)^{(n)}_{\tau+1}+in\alpha\Gamma\widehat{D}(R_3)^{(n)}_{\tau+1}\Big]^J\\&\quad+3Re^{-1}\Delta t^{-1}\Big[\Gamma^2\widehat{D}^2-(n\alpha)^2\Big]\Phi^{(n)}_{\tau}-(3/2)Re^{-1}\Delta t^{-1}\Big[\Gamma^2\widehat{D}^2-(n\alpha)^2\Big]\Phi^{(n)}_{\tau-1}\\&\quad+(1/3)Re^{-1}\Delta t^{-1}\Big[\Gamma^2\widehat{D}^2-(n\alpha)^2\Big]\Phi^{(n)}_{\tau-2},\qquad n\in\langle 0,N_M\rangle,\end{aligned} \tag{2.17}$$

while similar relations resulting from the first-, second- and fourth-order implicit methods are shown in Appendix A. In the above, the subscript τ denotes the time step and Δt stands for the (constant) length of the time step. The solution is obtained in an iterative manner during each time step with the superscript J denoting the iteration number. Relation (2.17) has the form of an inhomogeneous ordinary differential equation for $\Phi^{(n)}_{\tau+1}$. The following discussion will be carried out in the context of the third-order, implicit, backward-difference scheme while the relevant relations for the other schemes can be readily deduced. A detailed discussion of issues associated with the numerical implementation and performance of different temporal schemes is given in Section 4.3.

The unknown function $\Phi^{(n)}_{\tau+1}$ can be represented in terms of expansions based on the Chebyshev polynomials in the form

$$\Phi^{(n)}_{\tau+1}(\hat{y})=\sum_{k=0}^{k=\infty}Z^{(n)}_{k,\tau+1}T_k(\hat{y})\approx\sum_{k=0}^{k=N_T}Z^{(n)}_{k,\tau+1}T_k(\hat{y}), \tag{2.18}$$

where T_k denotes the Chebyshev polynomial of kth order and $Z^{(n)}_{k,\tau+1}$ denotes the unknown coefficients of the expansion. The modal functions associated with the Fourier expansions of the nonlinear terms can also be expressed in terms of expansions based on the Chebyshev polynomials in the form

$$\Big[(R_1)^{(n)}_{\tau+1},(R_2)^{(n)}_{\tau+1},(R_3)^{(n)}_{\tau+1}\Big](\hat{y})=\sum_{k=0}^{k=N_T}\Big[(\widetilde{R}_1)^{(n)}_{k,\tau+1},(\widetilde{R}_2)^{(n)}_{k,\tau+1},(\widetilde{R}_3)^{(n)}_{k,\tau+1}\Big]T_k(\hat{y}). \tag{2.19}$$

Relations (2.18 and (2.19) are substituted into (2.17) and Galerkin procedure [23] is used to develop a set of algebraic equations for the unknown coefficients $Z^{(n)}_{k,\tau+1}$ in the form

$$\begin{aligned}&\sum_{k=0}^{k=N_T}\Big[-Re^{-1}\Gamma^4\Big\langle T_j,\widehat{D}^4T_k\Big\rangle+\Big(11\Gamma^2Re^{-1}\Delta t^{-1}/6+2n^2\alpha^2\Gamma^2Re^{-1}\Big)\Big\langle T_j,\widehat{D}^2T_k\Big\rangle\\&\quad+in\alpha\Gamma^2\Big\langle T_j,u_0\widehat{D}^2T_k\Big\rangle+\Big(-n^4\alpha^4Re^{-1}-11n^2\alpha^2\Gamma^2\Delta t^{-1}Re^{-1}/6\Big)\langle T_j,T_k\rangle-in\alpha\Gamma^2\Big\langle T_j,\widehat{D}^2u_0T_k\Big\rangle-in^3\alpha^3\langle T_j,u_0T_k\rangle\Big]\Big(Z^{(n)}_{k,\tau+1}\Big)^{J+1}\\&\quad=\sum_{k=0}^{k=N_T}\Big[-in\alpha\Gamma\Big\langle T_j,\widehat{D}T_k\Big\rangle(\widetilde{R}_1)^{(n)}_{k,\tau+1}-\Gamma^2\Big\langle T_j,\widehat{D}^2T_k\Big\rangle(\widetilde{R}_2)^{(n)}_{k,\tau+1}-(n\alpha)^2\langle T_j,T_k\rangle(\widetilde{R}_2)^{(n)}_{k,\tau+1}+in\alpha\Gamma\Big\langle T_j,\widehat{D}T_k\Big\rangle(\widetilde{R}_3)^{(n)}_{k,\tau+1}\Big]^J\\&\quad+\sum_{k=0}^{k=N_T}(Re\Delta t)^{-1}\Big[\Gamma^2\Big\langle T_j,\widehat{D}^2T_k\Big\rangle-(n\alpha)^2\langle T_j,T_k\rangle\Big]\Big[3Z^{(n)}_{k,\tau}-(3/2)Z^{(n)}_{k,\tau-1}+(1/3)Z^{(n)}_{k,\tau-2}\Big],\quad n\in\langle 0,N_M\rangle,\ j\in\langle 0,N_T\rangle,\end{aligned} \tag{2.20}$$

where the inner product is defined as $\langle f_j(\hat{y}), g_k(\hat{y})\rangle = \int_{-1}^{1} f_j(\hat{y}) g_k(\hat{y}) \hat{\omega}(\hat{y}) d\hat{y}$ and $\hat{\omega} = 1/\sqrt{1-\hat{y}^2}$ denotes the weight function. The inner products are evaluated using the orthogonality properties of the polynomials. Eq. (2.20) leads to $N_T + 1$ algebraic equations for each Fourier mode; these equations are coupled through the nonlinear terms. Since the Chebyshev polynomials do not individually satisfy the boundary conditions, special construction is required to ensure that the global solution (2.14) satisfies the boundary conditions. These conditions will be accommodated in the Tau-like manner [24] where the four highest projection equations for each Fourier mode are dropped in order to make space for the relations that ensure that the discretized boundary conditions are satisfied. The numerical treatment and discretization of boundary conditions are discussed in the next Section and show additional source of coupling among different Fourier modes.

The evaluation of the nonlinear modal functions $R_1^{(n)}$, $R_2^{(n)}$ and $R_3^{(n)}$ involves determination of velocity components u_1 and v_1 in the physical space, evaluation of velocity products in the physical space, evaluation of the Fourier transforms of the velocity products and, finally, evaluation of the coefficients of Chebyshev expansions representing modal functions of the velocity products. Aliasing errors are controlled through the use of the padding method [24]. The padding method requires evaluation of the velocity components u_1 and v_1 in $(2M_M + 1)$ locations in the x-direction for each y-location where $M_M \geqslant 1.5N_M$. In order to take advantage of the properties of the FFT algorithm, it is recommended to select values of M_M equal to a power of 2. The FFT algorithm generates $(2M_M + 1)$ modes but only modes $\langle -N_M, N_M\rangle$ are retained. The cost of evaluation of the nonlinear terms is one of the dominant factors that affect the total computational cost per iteration. The iterations continue until the change in the magnitude of the Chebyshev coefficients for two consecutive iterations is less than the convergence criteria which, for all results presented in this paper, were set to 10^{-14}.

3. Numerical treatment of boundary conditions

The flow boundary conditions are imposed using the immersed boundary conditions (IBC) concept. Transformation (2.10) ensures that the boundary extremities at any given time are contained within the computational domain $\hat{y} \in \langle -1, 1\rangle$. The flow boundary conditions are to be enforced along the lines $\hat{y}_L(x, \tau + 1)$ and $\hat{y}_U(x, \tau + 1)$ that define the physical boundaries in the transformed coordinates $(x, \hat{y})$, where the locations of these lines are represented in the form

$$\hat{y}_L(x, \tau + 1) \approx \sum_{n=-N_A}^{n=+N_A} A_{L,\tau+1}^{(n)} e^{in\alpha x}, \quad \hat{y}_U(x, \tau + 1) \approx \sum_{n=-N_A}^{n=+N_A} A_{U,\tau+1}^{(n)} e^{in\alpha x}. \tag{3.1a,b}$$

The following discussion will be carried out in the context of the upper wall with the developments for the lower wall being analogous. In order to enforce the boundary conditions (2.13a,d) at time $\tau + 1$ one needs to evaluate $(\partial_{\hat{y}} \Psi_1)_{U,\tau+1} \equiv \partial_{\hat{y}} \Psi_1(x, \hat{y}_U(x, \tau + 1), \tau + 1)$ and $(\Psi_1)_{U,\tau+1} \equiv \Psi_1(x, \hat{y}_U(x, \tau + 1), \tau + 1)$ along the time-dependent line $\hat{y}_U(x, \tau + 1)$ defining the upper wall of the channel.

The terms $(\partial_{\hat{y}} \Psi_1)_{U,\tau+1}$ and $(\Psi_1)_{U,\tau+1}$ are periodic in x with the period $\lambda = 2\pi/\alpha$ and thus can be expressed in terms of Fourier series as

$$(\partial_{\hat{y}} \Psi_1)_{U,\tau+1} = \sum_{n=-N_U}^{n=+N_U} U_{\tau+1}^{(n)} e^{in\alpha x}, \quad (\Psi_1)_{U,\tau+1} = \sum_{n=-N_U}^{n=+N_U} V_{\tau+1}^{(n)} e^{in\alpha x}, \tag{3.2a,b}$$

where the value of N_U that needs to be used will be discussed later in the text. Since the flow representation used in the computations is limited to $N_M + 1$ modes (see Eq. (2.14)), even if $N_U > N_M$ only the first $(N_M + 1)$ terms in (3.2) can be accounted for. The components in (3.2) can also be evaluated along the wall at time $\tau + 1$ using the discretized form of the solution, i.e.,

$$(\partial_{\hat{y}} \Psi_1)_{U,\tau+1} = \sum_{n=-N_M}^{n=+N_M} \sum_{k=0}^{k=N_T} Z_{k,\tau+1}^{(n)} \widehat{D}T_k(\hat{y}_U(x, \tau + 1)) e^{in\alpha x}, \tag{3.3a}$$

$$(\Psi_1)_{U,\tau+1} = \sum_{n=-N_M}^{n=+N_M} \sum_{k=0}^{k=N_T} Z_{k,\tau+1}^{(n)} T_k(\hat{y}_U(x, \tau + 1)) e^{in\alpha x}. \tag{3.3b}$$

Chebyshev polynomials and their derivatives evaluated at the wall, i.e., $T_k(\hat{y}_U(x, \tau + 1)$ and $\widehat{D}T_k(\hat{y}_U(x, \tau + 1)$, are periodic functions of x and thus can be expressed in terms of Fourier expansion as follows

$$T_k(\hat{y}_U(x, \tau + 1)) = \sum_{m=-N_S}^{m=+N_S} (w_U)_{k,\tau+1}^{(m)} e^{im\alpha x}, \quad \widehat{D}T_k(\hat{y}_U(x, \tau + 1)) = \sum_{m=-N_S}^{m=+N_S} (d_U)_{k,\tau+1}^{(m)} e^{im\alpha x}, \tag{3.4a,b}$$

where $\max(N_S) = N_T N_A$. The method for evaluation of coefficients $(w_U)_{k,\tau+1}^{(m)}$ and $(d_U)_{k,\tau+1}^{(m)}$ is explained in Appendix B. Substitution of (3.4b) into (3.3a) gives

$$(\partial_{\hat{y}} \Psi_1)_{U,\tau+1} = \sum_{n=-N_M}^{n=+N_M} \sum_{m=-N_S}^{m=+N_S} \sum_{k=0}^{k=N_T} Z_{k,\tau+1}^{(n)} (d_U)_{k,\tau+1}^{(m)} e^{i(n+m)\alpha x} = \sum_{h=-N_U}^{h=+N_U} \sum_{n=-N_M}^{n=+N_M} \sum_{k=0}^{k=N_T} Z_{k,\tau+1}^{(n)} (d_U)_{k,\tau+1}^{(h-n)} e^{ih\alpha x}, \tag{3.5}$$

where $h = n + m$ and $N_U = N_T N_A + N_M$. Comparison of (3.2a) with (3.5a) along with change of indices in (3.5a), i.e., $n \to m$ and $h \to n$, give

$$U^{(n)}_{\tau+1} = \sum_{m=-N_M}^{m=+N_M} \sum_{k=0}^{k=N_T} Z^{(m)}_{k,\tau+1} (d_U)^{(n-m)}_{k,\tau+1}. \tag{3.6a}$$

Substitution of (3.4a) into (3.3b), followed by similar operations and comparison of the results with (3.2b), give

$$V^{(n)}_{\tau+1} = \sum_{m=-N_M}^{m=+N_M} \sum_{k=0}^{k=N_T} Z^{(m)}_{k,\tau+1} (w_U)^{(n-m)}_{k,\tau+1}. \tag{3.6b}$$

The function $(\widehat{D}\Psi_0)_{U,\tau+1}$ is known from the reference flow and can be expressed as

$$(\widehat{D}\Psi_0)_{U,\tau+1} = \sum_{n=-\infty}^{n=+\infty} F^{(n)}_{U,\tau+1} e^{in\alpha x}. \tag{3.7}$$

Substitution of (3.2), (3.6) and (3.7) into (2.13b,d) leads to boundary conditions along the upper wall while those for the lower wall can be derived following similar steps. The discretized boundary constraints for the both walls take the form

$$\sum_{m=-N_M}^{m=+N_M} \sum_{k=0}^{k=N_T} Z^{(m)}_{k,\tau+1} (d_U)^{(n-m)}_{k,\tau+1} = -F^{(n)}_{U,\tau+1}, \quad |n| \geqslant 0, \tag{3.8a}$$

$$\sum_{m=-N_M}^{m=+N_M} \sum_{k=0}^{k=N_T} Z^{(m)}_{k,\tau+1} (d_L)^{(n-m)}_{k,\tau+1} = -F^{(n)}_{L,\tau+1}, \quad |n| \geqslant 0, \tag{3.8b}$$

$$\sum_{m=-N_M}^{m=+N_M} \sum_{k=0}^{k=N_T} Z^{(m)}_{k,\tau+1} (w_U)^{(n-m)}_{k,\tau+1} = -(in\alpha\Gamma)^{-1} (A^{(n)}_U)'_{\tau+1}, \quad |n| \geqslant 1, \tag{3.8c}$$

$$\sum_{m=-N_M}^{m=+N_M} \sum_{k=0}^{k=N_T} Z^{(m)}_{k,\tau+1} (w_L)^{(n-m)}_{k,\tau+1} = -(in\alpha\Gamma)^{-1} (A^{(n)}_L)'_{\tau+1}, \quad |n| \geqslant 1, \tag{3.8d}$$

$$\sum_{m=-N_M}^{m=+N_M} \sum_{k=0}^{k=N_T} Z^{(m)}_{k,\tau+1} (w_U)^{(-m)}_{k,\tau+1} = C_U(\tau+1), \tag{3.8e}$$

$$\sum_{m=-N_M}^{m=+N_M} \sum_{k=0}^{k=N_T} Z^{(m)}_{k,\tau+1} (w_L)^{(-m)}_{k,\tau+1} = C_L(\tau+1), \tag{3.8f}$$

that are enforced for $|n| \leqslant N_m$. Relations obtained for $|n| > N_m$ can be used as a measure of error. The form of constants $C_L(\tau+1)$ and $C_U(\tau+1)$ that appear in Eq. (3.8e-f) can be determined using either the constant volume flux or the constant pressure gradient constraint. For convenience, we have selected the former one for this presentation, i.e., we assumed that the flow rates in the smooth reference conduit and in the deforming conduit remained the same.

Volume flux Q along the conduit can be evaluated by integrating the x-velocity component across the conduit, i.e.,

$$Q(x,\tau+1) = \int_{\hat{y}_L}^{\hat{y}_U} \left(\frac{\partial \Psi_T}{\partial \hat{y}}\right)_{\tau+1} d\hat{y} = \Psi_T(x,\hat{y}_U(x,\tau+1),\tau+1) - \Psi_T(x,\hat{y}_L(x,\tau+1),\tau+1). \tag{3.9}$$

where according to the definition of the total stream function

$$\Psi_T(x,\hat{y}_L(x,\tau+1),\tau+1) = \Psi_0(x,\hat{y}_L(x,\tau+1)) + \Psi_1(x,\hat{y}_L(x,\tau+1),\tau+1), \tag{3.10a}$$

$$\Psi_T(x,\hat{y}_U(x,\tau+1),\tau+1) = \Psi_0(x,\hat{y}_U(x,\tau+1)) + \Psi_1(x,\hat{y}_U(x,\tau+1),\tau+1). \tag{3.10b}$$

Values of Ψ_0 evaluated along the lower and upper walls represent known functions of t and x that are periodic in x and can be expressed as

$$\Psi_0(\hat{y}_L(x,\tau+1)) = \sum_{n=-N_M}^{n=+N_M} \left(\Xi^{(n)}_L\right)_{\tau+1} e^{in\alpha x}, \quad \Psi_0(\hat{y}_U(x,\tau+1)) = \sum_{n=-N_M}^{n=+N_M} \left(\Xi^{(n)}_U\right)_{\tau+1} e^{in\alpha x}. \tag{3.11}$$

The volume flux represents an x-periodic function that can be written in the form of a Fourier expansion

$$Q(x,\tau+1) = \sum_{n=-N_M}^{n=+N_M} q^{(n)}(\tau+1) e^{in\alpha x}, \tag{3.12}$$

where the zero term, i.e., $q^{(0)}$, represents the net mass flux along the conduit. The value of $q^{(0)}(\tau+1)$ is assumed in this analysis to be known and independent of time, and equal to the flow rate of the reference flow, i.e., $q^{(0)} = \frac{4}{3}$.

Substitution of (2.13), (3.10)–(3.12) into Eq. (3.9) and extraction of mode zero results in

$$C_U(\tau+1) = Q^{(0)} + C_L(\tau+1) + \Xi_L^{(0)} - \Xi_U^{(0)}. \tag{3.13}$$

One of the constants, either C_U or C_L, can be selected arbitrarily and the other one follows from (3.13). In the description given below the latter one has been selected arbitrarily by introducing condition $\Psi_T = 0$ at a conveniently selected point $x = x_0$ at the lower wall resulting in

$$C_L(\tau+1) = -\Xi_L^{(0)} + \sum_{n=-N_A, n\neq 0}^{n=+N_A} (in\alpha\Gamma)^{-1}(A_L^{(n)})'_{\tau+1} e^{in\alpha x_0}. \tag{3.14}$$

Substitution of (3.14) into (3.13) gives

$$C_U(\tau+1) = -\Xi_U^{(0)} + \sum_{n=-N_A, n\neq 0}^{n=+N_A} (in\alpha\Gamma)^{-1}(A_L^{(n)})'_{\tau+1} e^{in\alpha x_0} + q^{(0)}. \tag{3.15}$$

Substitution of (3.14) and (3.15) into (3.8e–f) results in the form of the closing conditions useful for numerical implementation, i.e.,

$$\sum_{m=-N_M}^{m=+N_M}\sum_{k=0}^{k=N_T} Z_{k,\tau+1}^{(m)}\left(w_{k,\tau+1}^{(m)^*}\right)_L = -\Xi_L^{(0)} + \sum_{n=-N_A, n\neq 0}^{n=+N_A} (in\alpha\Gamma)^{-1}(A_L^{(n)})'_{\tau+1} e^{in\alpha x_0}, \tag{3.16a}$$

$$\sum_{m=-N_M}^{m=+N_M}\sum_{k=0}^{k=N_T} Z_{k,\tau+1}^{(m)}\left(w_{k,\tau+1}^{(m)^*}\right)_U = -\Xi_U^{(0)} + \sum_{n=-N_A, n\neq 0}^{n=+N_A} (in\alpha\Gamma)^{-1}(A_L^{(n)})'_{\tau+1} e^{in\alpha x_0} + q^{(0)}. \tag{3.16b}$$

Eqs. (2.20), (3.8a–d) and (3.16) form a complete set of algebraic equations for the unknown coefficients $Z_{k,\tau+1}^{(n)}$, $k = 0,\ldots,N_T$, $n = 0,\ldots,N_M$. A solution of this set moves calculations forward by one time step. The performance of the algorithm and various methods of solution of the algebraic equations are discussed in the next section.

4. Performance of the algorithm

Two different problems have been selected to test and characterize the performance of the algorithm. The test problems involve (i) peristaltic flow and (ii) pulsatile flow. The selection of these test problems enables us to show the applicability of the proposed algorithm in solving flow problems of considerable practical interest, particularly in the field of biomedical engineering.

4.1. Peristaltic flow

4.1.1. Problem prototype

A general form of peristaltic flow can be modelled by considering peristaltic wave traveling along the conduit walls, whose locations can be described as

$$y_L(x,t) = -1 + \sum_{n=-N_m, n\neq 0}^{n=+N_M} H_L^{(n)} e^{in\alpha(x-ct)}, \quad y_U(x,t) = 1 + \sum_{n=-N_m, n\neq 0}^{n=+N_M} H_U^{(n)} e^{in\alpha(x-ct)}, \tag{4.1a,b}$$

where c denotes the phase speed of the wave and α stands for its wave number. The simplest situation corresponds to the wave profiles described by only one Fourier mode, resulting in the wall motions described by

$$y_L(x,t) = -1 + S\cos[\alpha(x-ct)] = -1 + (0.5Se^{i\alpha(x-ct)} + CC), \tag{4.2a}$$

$$y_U(x,t) = 1 - S\cos[\alpha(x-ct)] = 1 - (0.5Se^{i\alpha(x-ct)} + CC), \tag{4.2b}$$

where S stands for the amplitude of the wave and CC denotes complex conjugate. Fig. 2 illustrates changes in the location of the walls as a function of time.

Peristaltic flow represents a convenient test problem because it can be converted into a steady, fixed boundary problem in the moving frame of reference (X, y) using Galileo transformation in the form

$$X = x - ct. \tag{4.3}$$

This is an easier problem and its solution provides a convenient comparison problem for the direct numerical solution of the time-dependent, moving boundary problem in the fixed frame of reference. In addition, this problem provides a convenient test for characterization of the accuracy of spatial discretization.

The full problem in the moving frame of reference (X, y) takes the form

$$-Re^{-1}\nabla^4\Psi_1 + [(u_0 - c)\partial_X(\nabla^2) - D^2u_0\partial_X]\Psi_1 = -\partial_y[\partial_X(u_1u_1) + \partial_y(u_1v_1)] + \partial_X[\partial_X(u_1v_1) + \partial_y(v_1v_1)] \tag{4.4}$$

where $\nabla^2 = \partial_{XX} + \partial_{yy}$ and the boundary shapes take the forms

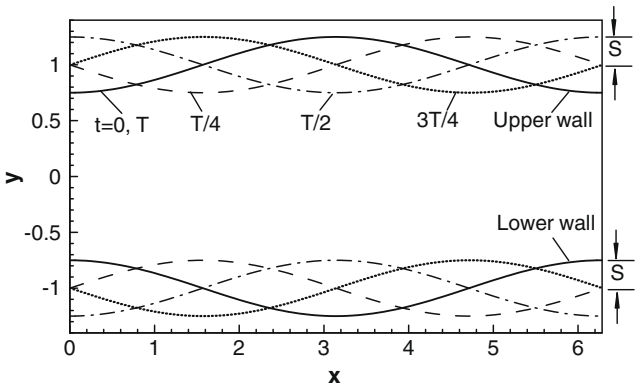

Fig. 2. Locations of the conduit's walls deformed by peristaltic wave described by Eq. (4.2) with the amplitude $S = 0.25$, the wave number $\alpha = 1.0$ and the phase speed $c = \pi$ at times $t = 0$, $T/4$, $T/2$, $3T/4$ and T, where T denotes one time period.

$$y_L(X) = -1 + (0.5Se^{i\alpha X} + CC), \quad y_U(X) = 1 - (0.5Se^{i\alpha X} + CC). \tag{4.5a, b}$$

The boundary conditions in the moving frame of reference (X, y) have the form

$$u(X, y_{L,U}(X)) = 0, \quad v(y_{L,U}(X)) = -c \sum_{n=-N_M, n\neq 0}^{n=+N_M} in\alpha H_{L,U}^{(n)} e^{in\alpha x}. \tag{4.6a, b}$$

The stationary problem (4.4)–(4.6) shall be referred to as the model problem #1, and will be solved using the steady version of the technique discussed in the previous Sections.

The same problem expressed in the fixed frame of reference (x, y) consists of the field Eq. (2.7), boundary conditions (2.9) with wall motions given by (4.2). This problem can be solved directly as a time-dependent, moving boundary problem – we shall refer to this problem as the model problem #2. The solution of the model problem #1 is used as initial condition for the model problem #2 in various tests discussed later in the text, unless explicitly stated otherwise.

4.1.2. Solution of the model problem #1

Model problem #1 provides an opportunity to demonstrate the spectral accuracy of the spatial discretization. In order to have a meaningful discussion of error, we have produced a machine-accurate reference solution for this problem using the Domain Transformation (DT) method where the irregular flow domain is analytically mapped into a regular computational domain resulting in the classical treatment of boundary conditions. We have used the same spatial discretization as in the case of the IBC method, i.e., Fourier expansions in the X-direction and Chebyshev expansions in the y-direction. Details of the DT method can be found in Appendix C.

We define error in the evaluation of the u- and v-velocity components at any location (X, y) as

$$u_{er}(X, y) = u_{\mathrm{IBC}}(X, y) - u_{\mathrm{DT}}(X, y), \quad v_{er}(X, y) = v_{\mathrm{IBC}}(X, y) - v_{\mathrm{DT}}(X, y), \tag{4.7a, b}$$

where the subscript 'IBC' denotes solution obtained using the IBC algorithm and the subscript 'DT' denotes the machine-accurate reference solution determined using the DT method. We shall use the L_∞ norms defined as

$$\|u_{er}\|_\infty = \sup_{\substack{0\leqslant X\leqslant 2\pi/\alpha \\ y_L(X)\leqslant X\leqslant y_U(X)}} |u_{er}(X, y)|, \quad \|v_{er}\|_\infty = \sup_{\substack{0\leqslant X\leqslant 2\pi/\alpha \\ y_L(X)\leqslant X\leqslant y_U(X)}} |v_{er}(X, y)|, \tag{4.8a, b}$$

as a measure of error over the whole flow domain. Error in the enforcement of flow boundary conditions is of special interest and is measured using the L_∞ norms defined as

$$\|u_{er,BC}(X)\|_\infty = \sup_{0\leqslant X\leqslant 2\pi/\alpha} \{|(u_{er,BC}(X))_L|, |(u_{er,BC}(X))_U|\}, \tag{4.9a}$$

$$\|v_{er,BC}(X)\|_\infty = \sup_{0\leqslant X\leqslant 2\pi/\alpha} \{|(v_{er,BC}(X))_L|, |(v_{er,BC}(X))_U|\}, \tag{4.9b}$$

where

$$(u_{er,BC}(X))_{L,U} = u(X, y_{L,U}(X)), \qquad (v_{er,BC}(X))_{L,U} = v(X, y_{L,U}(X)) - (v_{wall}(X))_{L,U}. \tag{4.10a, b}$$

Here the terms $(v_{wall}(X))_L$ and $(v_{wall}(X))_U$ stand for the actual velocities of the lower and upper walls, respectively. We shall focus further discussion on the error in the u-velocity component, i.e., u_{er}. This error is related to the error of the y-derivative of the stream function which needs to be evaluated numerically and thus evaluation of u is potentially less accurate than evaluation of v.

The Chebyshev expansions (2.18) with coefficients calculated using the Galerkin procedure are expected to deliver spectral accuracy in the y-direction with the increasing number of terms N_T. The validation of this fact is illustrated in Fig. 3. This figure also shows that the use of a single norm $\|u_{er,BC}\|_\infty$ as an error measure could be misleading when the number of Chebyshev polynomials employed in the computations is not adequate. The reason behind this property can be explained with the aid results displayed in Fig. 4, which shows formation of boundary layers in the distribution of modal functions in the zones around the walls. Although the discretization in the y-direction using the Chebyshev expansions is fairly standard, the existence of boundary layers creates special problems. Inadequate number of polynomials can lead to spurious oscillations in the distribution of the modal functions and increase the error in the overall solution although the boundary conditions may still be satisfied with very high accuracy (see Fig. 3). In most cases, 60 Chebyshev polynomials provide machine accuracy. However, when the wavelengths of the peristaltic wave become shorter (α increases), higher Fourier modes begin to play important role and one needs to increase the number of Chebyshev polynomials in order to resolve the wall boundary layers with reasonable accuracy. Fig. 4 shows that these layers become extremely thin for larger values of α and for higher Fourier modes. Modal functions change very rapidly inside these layers while they are nearly zero in the rest of the domain.

The convergence of the truncated Fourier series is pertinent to the second aspect of the spatial discretization, i.e., accuracy of discretization in the x-direction. Chebyshev norm defined as

$$\|D\Phi^{(n)}\|_{\hat{\omega}} = \sqrt{I^2 \int_{-1}^{1} D\Phi^{(n)}(\hat{y},t) D\Phi^{(n)^*}(\hat{y},t)\hat{\omega}(\hat{y})d\hat{y}}, \quad \hat{\omega} = 1/\sqrt{1-\hat{y}^2} \tag{4.11}$$

is used as a measure of the magnitude of the derivative of the modal function $\Phi^{(n)}$ (i.e., the u-velocity component). Results displayed in Fig. 5 demonstrate that this norm decreases exponentially as a function of the mode number n.

The next question of interest is the identification of the number of Fourier modes N_M required for a desired level of accuracy of the solution. Results displayed in Fig. 6 demonstrate that when a sufficient number of Chebyshev polynomials is used (see Fig. 3), the maximum error over the computational domain becomes equal to the maximum error in the enforcement of flow boundary conditions, i.e.,

$$\|u_{er}\|_\infty = \|u_{er,BC}\|_\infty \quad \text{and} \quad \|v_{er}\|_\infty = \|v_{er,BC}\|_\infty \quad \text{when} \quad \frac{d}{dN_T}(\|u_{er}\|_\infty, \|v_{er}\|_\infty) \approx 0. \tag{4.12}$$

Therefore, the number of Fourier modes N_M used in the computations contributes only to the error in the enforcement of flow boundary conditions while the magnitude of the error inside the solution domain is determined by the number of Chebyshev polynomials. This fact is further substantiated through the results displayed in Fig. 7, which demonstrate that the error reaches maximum around the wall and decreases rapidly as one moves away from the wall. Thus, one can use the norm $\|u_{er,BC}\|_\infty$ to quantify the maximum error for any number of Fourier modes N_M assuming that a sufficient number of Chebyshev polynomials N_T have been used.

Fig. 8 shows distribution of error in the enforcement of flow boundary conditions for the u- and v-velocity components along the upper wall. It can be seen that the errors oscillate along the conduit and that the locations of the maxima of the amplitudes overlap with the location where the peristaltic wave maximizes conduit opening. This figure also illustrates that the magnitude of the error in the u-velocity component is slightly higher than that in the v-velocity component, as had already been pointed out in the preceding discussion.

Variations of the error $\|u_{er}\|_\infty$ as a function of the amplitude S of the peristaltic wave for fixed values of its wave number α, and as a function of its wave number α for fixed amplitudes S are illustrated in Figs. 9 and 10, respectively. The results shown

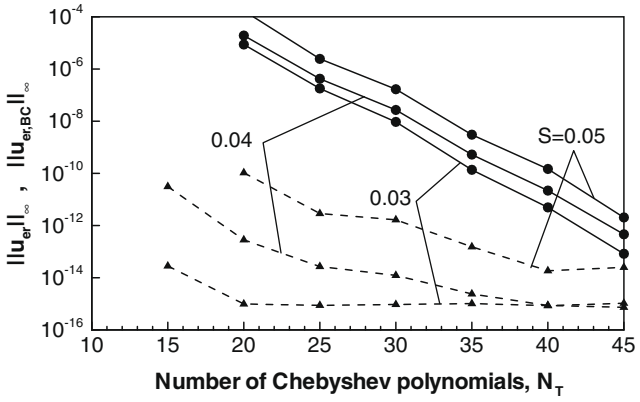

Fig. 3. Variations of the norms $\|u_{er}\|_\infty$ (solid lines) and $\|u_{er,BC}\|_\infty$ (dashed lines) as a function of the number of Chebyshev polynomials N_T used in the computations for the model problem #1 (see Eqs. (4.4)–(4.6)) for the peristaltic wave with the wave number $\alpha = 1$, the phase speed $c = \pi$ and different amplitudes S. Computations have been carried out for the flow Reynolds number $Re = 100$ using $N_M = 15$ Fourier modes.

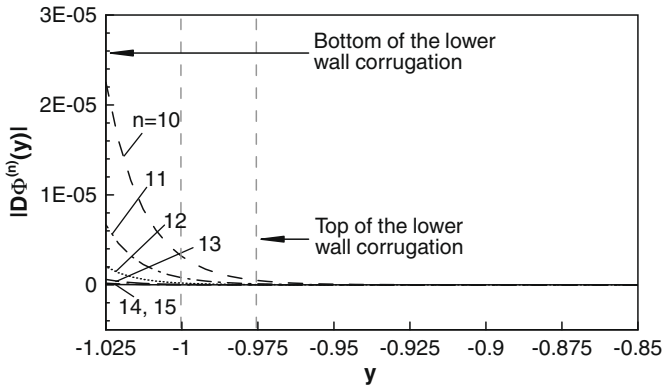

Fig. 4. Distribution of $|D\Phi^{(n)}|$ for higher modes ($n \geqslant 10$) in the vicinity of the lower wall for the model problem #1 (see Eqs. (4.4)–(4.6)) for the peristaltic wave with the wave number $\alpha = 5$, the amplitude $S = 0.025$ and the phase speed $c = \pi$. Computations have bee carried out for the flow Reynolds number $Re = 100$ using $N_M = 15$ Fourier modes and $N_T = 100$ Chebyshev polynomials.

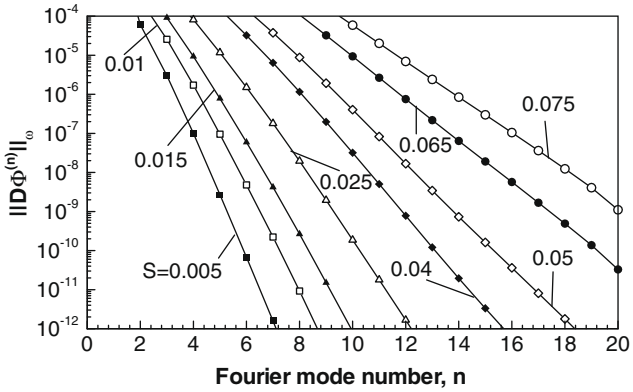

Fig. 5. Variations of the Chebyshev norm of $D\Phi^{(n)}$ (see Eq. (4.5)) as a function of the Fourier mode number n for the model problem #1 (see Eqs. (4.4)–(4.6)) for different amplitudes S of the peristaltic wave with the wave number $\alpha = 1.0$ and the phase speed $c = \pi$. Computations have been carried out for the flow Reynolds number $Re = 100$ using $N_M = 20$ Fourier modes and $N_T = 80$ Chebyshev polynomials.

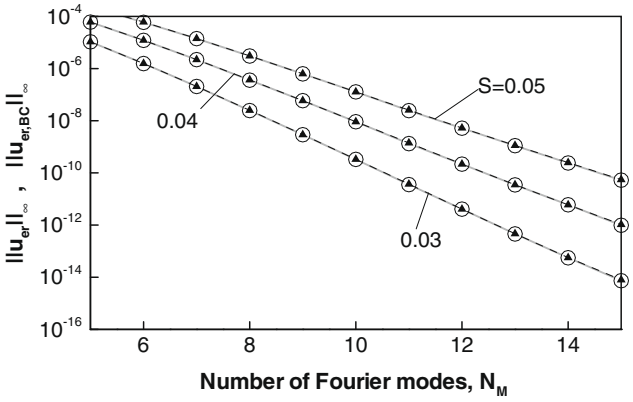

Fig. 6. Variations of the norms $\|u_{er}\|_\infty$ (circles) and $\|u_{er,BC}\|_\infty$ (triangles) as a function of the number of Fourier modes N_M used in the computations for the model problem #1 (see Eqs. (4.4)–(4.6)) for the peristaltic wave with the wave number $\alpha = 1$, the phase velocity $c = \pi$ and selected values of the amplitudes S. Computations have been carried out for the flow Reynolds number $Re = 100$ using $N_T = 80$ Chebyshev polynomials.

demonstrate that for certain combinations of α and S, the accuracy can be maintained at the machine level. However, the error increases almost exponentially when values of α and S reach certain critical threshold. The error can be controlled by increasing the number of Fourier modes used in the computations and/or by using the over-constraint formulation

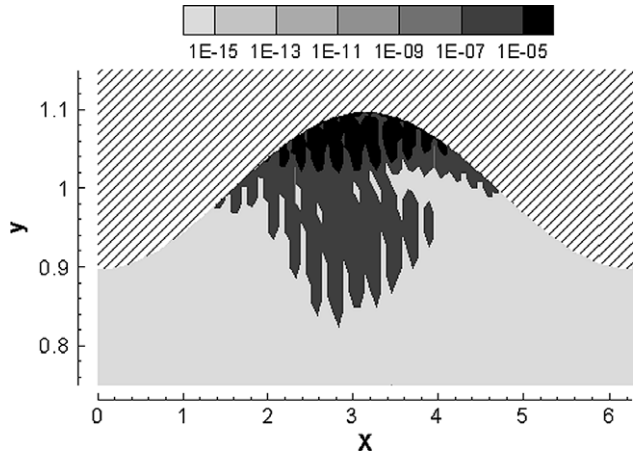

Fig. 7. Distribution of the absolute value of the error $u_{er}(X,y)$ (see Eq. (4.7)) around the upper wall for the model problem #1 (see Eqs. (4.4)–(4.6)) for the peristaltic wave with the wave number $\alpha = 1.0$, the amplitude $S = 0.1$ and the phase speed $c = \pi$. Computations have been carried out for the flow Reynolds number $Re = 100$ using $N_M = 20$ Fourier modes and $N_T = 100$ Chebyshev polynomials.

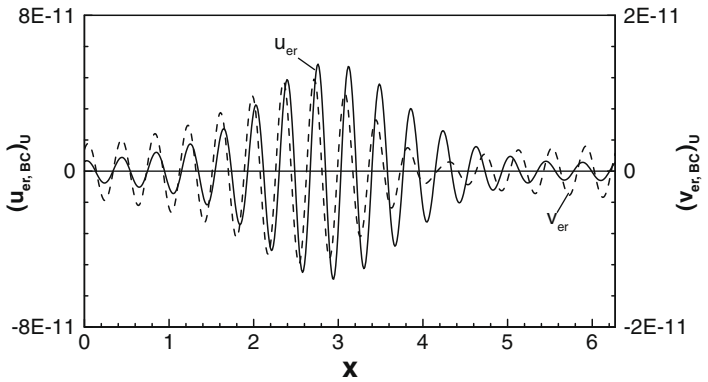

Fig. 8. Distribution of the boundary errors $(u_{er,BC})_U$ (see Eq. (4.10a), solid line) and $(v_{er,BC})_U$ (see Eq.(4.10b), dash line) for the model problem #1 (see Eqs. (4.4)–(4.6)) for the peristaltic wave with the phase speed $c = \pi$, the amplitude $S = 0.05$ and the wave number $\alpha = 1$. Computations have been carried out for the flow Reynolds number $Re = 100$ using $N_M = 15$ Fourier modes and $N_T = 80$ Chebyshev polynomials.

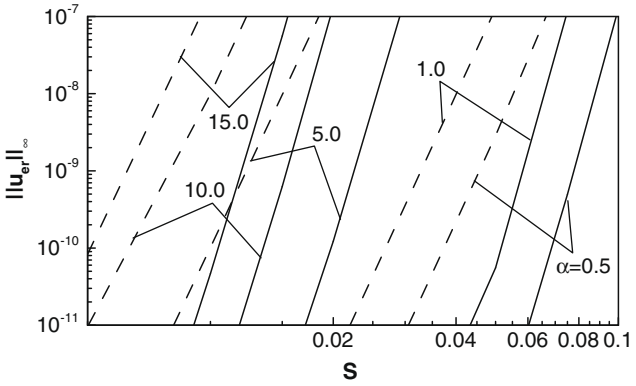

Fig. 9. Variations of the $\|u_{er}(X)\|_\infty$ norm (see Eq. (4.8)) as a function of the amplitude S of the peristaltic wave with the phase speed $c = \pi$ and with selected values of the wave number α in the case of flow with the Reynolds number $Re = 100$ (model problem#1, see Eqs. (4.4)–(4.6)). The dashed and solid lines correspond to results obtained with the $N_M = 10$ and 15 Fourier modes, respectively. $N_T = 80$ Chebyshev polynomials were used in the computations.

[25], which provides ability to simulate dynamics of peristaltic waves with more complex profiles. Both techniques do not however affect the qualitative character of the error variations; they merely increase the threshold of α and S that leads to a rapid error increase.

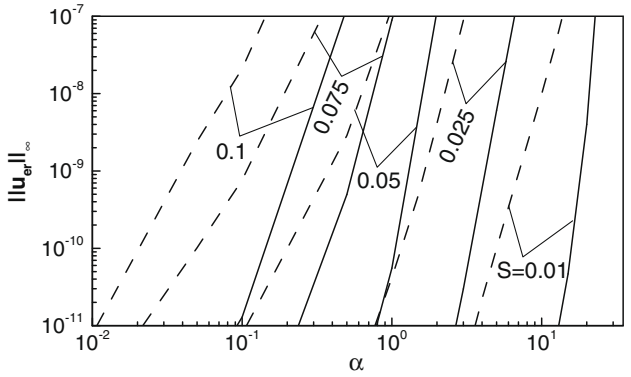

Fig. 10. Variations of the $\|u_{er}(X)\|_\infty$ norm (see Eq. (4.8)) as a function of the wave number α of the peristaltic wave with the wave speed $c = \pi$ and with selected values of the amplitude S. Other parameters are as in Fig. 9.

4.1.3. Solution of the model problem #2

We shall now focus our discussion on the model problem #2, i.e., a moving boundary problem expressed in a fixed frame of reference (x, y). Solution of the model problem #1 provides initial conditions via the reversed Galileo transformation. We shall use the L_∞ norms as quantitative measures of error associated with the enforcement of flow boundary conditions, i.e.,

$$\|u_{er,BC}(x,t)\|_\infty = \sup_{0\leqslant X\leqslant 2\pi/\alpha}\left\{\left(\left|u_{er,BC}(x,t)\right|\right)_L,\ \left(\left|u_{er,BC}(x,t)\right|\right)_U\right\}, \tag{4.13a}$$

$$\|v_{er,BC}(x,t)\|_\infty = \sup_{0\leqslant X\leqslant 2\pi/\alpha}\left\{\left(\left|v_{er,BC}(x,t)\right|\right)_L,\ \left(\left|v_{er,BC}(x,t)\right|\right)_U\right\}, \tag{4.13b}$$

where

$$(u_{er,BC}(x,t))_{L,U} = u(x, y_{L,U}(x,t), t), \quad (v_{er,BC}(x,t))_{L,U} = v(x, y_{L,U}(x,t), t) - (v_{wall})_{L,U}(x,t). \tag{4.14a, b}$$

The test problem is solved using different temporal discretizations discussed in Section 2. Fig. 11 illustrates the time history of $\|u_{er}(x,t)\|_\infty$ and demonstrates that the location of the maximum error follows the location of the maximum conduit opening as it moves in the positive x-direction, similarly as in the case of solution obtained in the moving frame of reference (see Fig. 8). The magnitude of this error remains approximately constant while several waves pass through the computational box. Variations of the u-velocity component at a few test points are displayed in Fig. 12. The test points have the same y-coordinates but are spaced apart in the x-direction by a distance of $\lambda/4$, where λ is the wavelength of the peristaltic wave. The results show the expected phase differences associated with different locations of the test points. Both figures, i.e., Figs. 11 and 12, demonstrate the expected periodic variations in time of the computed quantities.

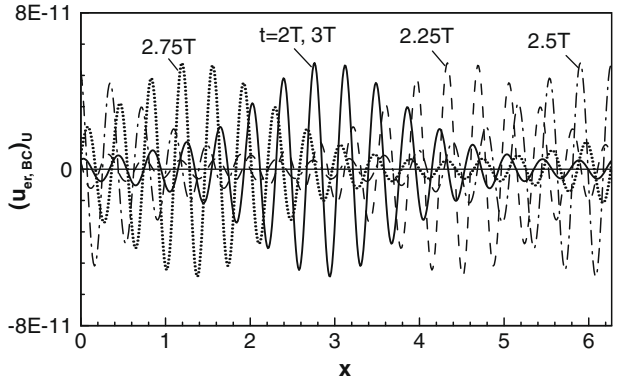

Fig. 11. Distribution of the boundary error $(u_{er,BC})_U$ (see Eq. (4.14)) at times $t = 2T, 2.25T, 2.5T, 2.75T$ and $3T$, where T stand for one time period, for the model problem #2 with the amplitude of the peristaltic wave $S = 0.05$, the wave number $\alpha = 1.0$ and the phase speed $c = \pi$. Computations have been carried out for the flow Reynolds number $Re = 100$ using $N_M = 15$ Fourier modes, $N_T = 80$ Chebyshev polynomials and the third-order implicit temporal discretization scheme with the time step $\Delta t = 0.001$. Solution of the model problem #1 was used as the initial condition.

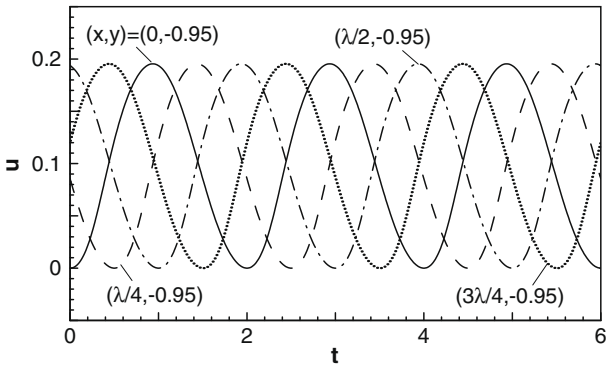

Fig. 12. Variations of the u-velocity component over three time periods at four test points for the model problem #2 solved directly as a moving boundary problem in the fixed reference frame.$\lambda = 2\pi/\alpha$ stands for the wavelength of the peristaltic wave. Other conditions are as in Fig. 11.

4.2. Pulsatile flow

4.2.1. Problem prototype

Pulsatile flows can be modelled by replacing the flat conduit walls with elastic standing waves, which can be expressed as

$$y_L(x,t) = -1 + \sum_{n=-N_M, n\neq 0}^{n=+N_M} H_L^{(n)}(t) e^{in\alpha x}, \quad y_U(x,t) = 1 + \sum_{n=-N_M, n\neq 0}^{n=+N_M} H_L^{(n)}(t) e^{in\alpha x} \tag{4.15a,b}$$

and, in the simplest case of a sinusoidal wave, the shape and motion of the walls can be described as

$$y_L(x,t) = -1 + S\cos(\omega t)\cos(\alpha x) = -1 + (0.5S\cos(\omega t)e^{i\alpha x} + CC), \tag{4.16a}$$

$$y_U(x,t) = 1 - S\cos(\omega t)\cos(\alpha x) = 1 - (0.5S\cos(\omega t)e^{i\alpha x} + CC), \tag{4.16b}$$

where α denotes the wave number of the standing wave, S stands for its amplitude and ω denotes its frequency. In the above, CC implies complex conjugate. The character of motion of the walls in this case is illustrated in Fig. 13.

The complete test problem consists of the discretized field Eq. (2.20), the boundary constraints (3.8a-d, 3.16), the boundary motions described by (4.16) and suitable initial conditions consistent with the boundary shapes. We shall refer to this problem as the model problem #3. For convenience, we shall use the L_∞ norms defined by Eq. (4.13) as quantitative measures of error associated with the enforcement of flow boundary conditions.

4.2.2. Solution of the model problem #3

Fig. 14 illustrates variations of the maximum error in the enforcement of the flow boundary condition for the u-velocity as a function of time over two time periods. It can be seen that the magnitude of the error changes periodically in time with frequency equal to double the frequency of the wave. The occurrence of the maximum error has certain phase lag with respect to the occurrence of the maximum conduit opening, with the phase lag increasing with an increase of Re. Results displayed in Fig. 15 demonstrate that the time lag does not depend on the wave amplitude, which implies that this lag is a

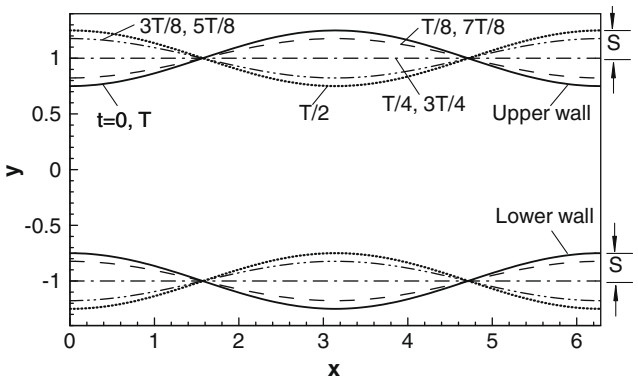

Fig. 13. Locations of the conduit's walls deformed by an elastic standing wave with the wave number $\alpha = 1.0$, the amplitude $S = 0.25$ and the frequency $\omega = \pi$ at times $t = 0$, $T/8$, $T/4$, $3T/8$, $T/2$, $5T/8$, $3T/4$, $7T/8$ and T, where T denotes one time period.

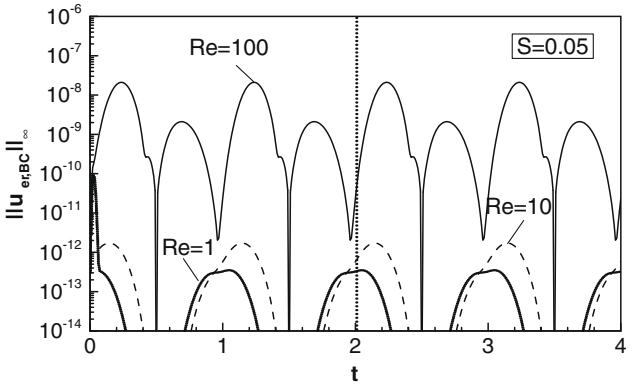

Fig. 14. Variations of the $\|u_{er,BC}(x,t)\|_\infty$ norm (see Eq. (4.13)) as a function of time over two time periods for the standing wave problem (model problem #3) with the wave number $\alpha = 1$, the amplitude $S = 0.05$ and the frequency $\omega = \pi$ for three Reynolds numbers, i.e., $Re = 1$, 10 and 100. Computations have been carried out using $N_M = 15$ Fourier modes, $N_T = 80$ Chebyshev polynomials and the third-order implicit temporal discretization scheme with the time step $\Delta t = 0.001$. Solution of the steady, fixed boundary problem with the shape of the walls given by Eq. (4.16) was used as the initial condition.

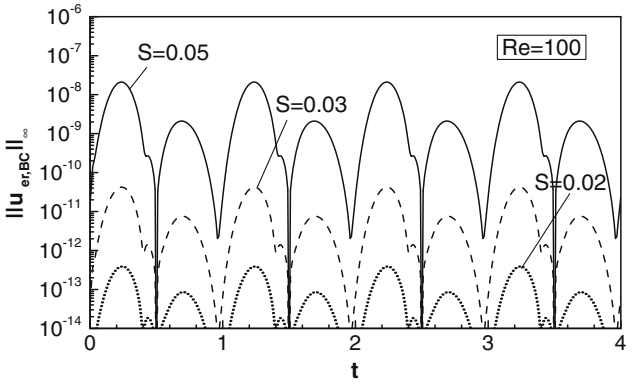

Fig. 15. Variations of the $\|u_{er,BC}(x,t)\|_\infty$ norm (see Eq. (4.13)) as a function of time over two time periods for the standing wave problem (model problem #3) with the wave number $\alpha = 1$ and the frequency $\omega = \pi$ for three values of the wave amplitude, i.e., $S = 0.02$, 0.03 and 0.05. Other conditions are as in Fig. 14.

function of the Reynolds number and the wave frequency only. Results shown in Fig. 16 demonstrate that the magnitude of the error can be reduced by increasing the number of Fourier modes used in the computations but its qualitative character remains unchanged.

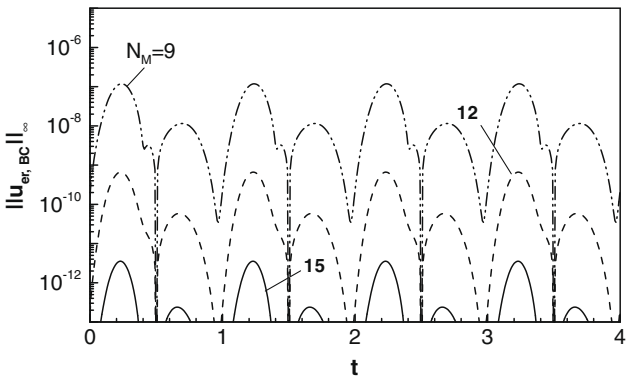

Fig. 16. Variations of the $\|u_{er,BC}(x,t)\|_\infty$ norm (see Eq. (4.13)) as a function of time over three time periods for the standing wave problem (model problem #3) with the wave number $\alpha = 1$, the amplitude $S = 0.025$ and the frequency $\omega = \pi$. Computations have been carried out for the flow Reynolds number $Re = 100$ using $N_M = 9$, 12 and 15 Fourier modes. Other conditions are as in Fig. 14.

The spatial distribution of the error is illustrated in Fig. 17 after 2.125 and 2.625 cycles of the wave motion, i.e., when the error is largest (see Fig. 16). It can be seen that the maximum of the error occurs at a location corresponding to the maximum conduit opening at a given instant of time and the magnitudes of the maxima at the two time levels are essentially identical. The reader should note that the maximum of this error as a function of time occurs after the walls have already moved passed the position corresponding to the maximum possible conduit opening, as discussed in the previous paragraph.

The Fourier spectra of the error in the enforcement of the u-velocity boundary conditions is given by the following relation

$$(u_{er,BC}(x,t))_{L,U} = \sum_{n=-\infty}^{n=+\infty} \left(U^{(n)}_{er,BC}(t)\right)_{L,U} e^{in\alpha x}. \tag{4.17}$$

The computed spectra shown in Fig. 18 demonstrate that the first N_M Fourier modes have been eliminated, according to the construction of the boundary constraints described in Section 3. The largest error is associated with the first Fourier mode omitted in the enforcement of the flow boundary conditions; the error associated with the following modes rapidly decreases as the mode number increases. Fig. 18 also displays results of tests carried out in order to check if the method produces any spurious spatial oscillations. Three cases were considered, i.e., in case (A) the wave was represented by the principal Fourier mode and the calculations had been carried out with $N_M = 5$ Fourier modes, in case (B) the wave was represented by the second Fourier mode (the principal mode had the wave number $\alpha = 0.5$), and in case (C) the wave was represented by the third Fourier mode (the principal mode had the wave number $\alpha = 1/3$). In order to have fully equivalent representations, the number of Fourier modes used in cases (B) and (C) were $N_M = 10$ and $N_M = 15$, respectively. The problem set up admitted sub-harmonics of the 1/2 type in case (B) and 1/3 type in case (C). The Fourier spectra shown in Fig. 18 demonstrate the equivalency of the results in all three cases. No sub-harmonics had been produced during the solution

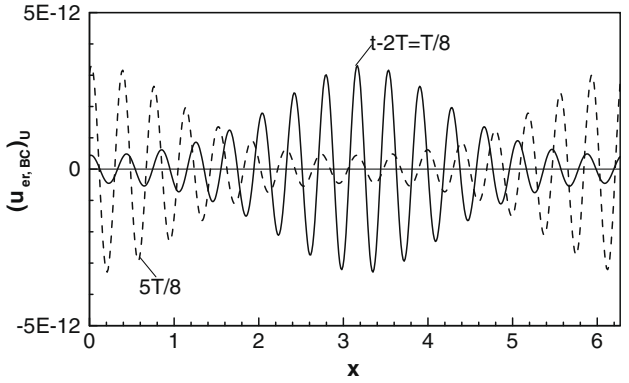

Fig. 17. Spatial distribution of the error $(u_{er,BC})_U$ (see Eq. (4.14)) at two different instances of time for the same standing wave problem as described in Fig. 16. Other conditions are as in Fig. 14.

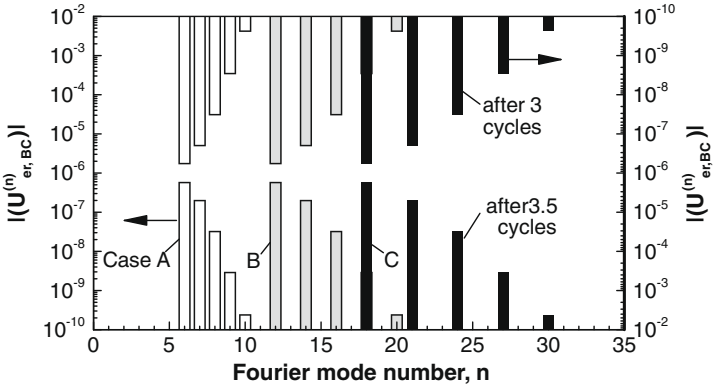

Fig. 18. Fourier spectra of distribution of the boundary error $(u_{er,BC})_U$ (see Eq. (4.17)) for the standing wave problem (model problem #3) with the wave amplitude $S = 0.025$ and the frequency $\omega = \pi$. Three different forms of Fourier expansions were considered, i.e., case A: $\alpha = 1.0$, $N_M = 5$; case B: $\alpha = 0.5$, $N_M = 10$ and case C: $\alpha = 1/3$, $N_M = 15$. Computations have been carried out using $N_T = 80$ Chebyshev polynomials and the third-order implicit temporal discretization scheme with the time step $\Delta t = 0.001$. Solution of the steady, fixed boundary problem with the shape of the walls given by Eq. (4.16) was used as the initial condition.

process and the modes expected to produce zero contributions in cases (B) and (C) behaved as expected. Temporal variations of the u- and v-velocity components at a test point displayed in Fig. 19 demonstrate the expected periodic variations in time, with the error of u being larger than error of v, as had already been discussed.

4.3. Temporal discretization

In the course of the present work we have tested four different fully-implicit temporal discretization schemes, which can deliver first-, second-, third- and fourth-order accuracy. The schemes treat all terms implicitly resulting in a nonlinear system of algebraic equations that has to be solved iteratively at each time step. The general form of the modal field Eq. (2.15) can be written as

$$\partial_t[\mathcal{L}_1\Phi^{(n)}] = \mathcal{L}_2\Phi^{(n)} + \mathcal{N}\Phi^{(n)}, \tag{4.18}$$

where $\mathcal{L}_1$, $\mathcal{L}_2$ denote linear and $\mathcal{N}$ stands for nonlinear differential operators acting on the unknown modal functions $\Phi^{(n)}$. The expression for advancing from $\Phi^{(n)}_\tau$ at time t to $\Phi^{(n)}_{\tau+1}$ at time $t+\Delta t$ using the first-, the second-, the third- and the fourth-order backward-difference schemes have the forms

$$[\Delta t^{-1}\mathcal{L}_1 - \mathcal{L}_2]\left\{\Phi^{(n)}_{\tau+1}\right\}^{J+1} = \mathcal{N}\left\{\Phi^{(n)}_{\tau+1}\right\}^{J} + \Delta t^{-1}\mathcal{L}_1\Phi^{(n)}_\tau, \tag{4.19a}$$

$$[(3/2)\Delta t^{-1}\mathcal{L}_1 - \mathcal{L}_2]\left\{\Phi^{(n)}_{\tau+1}\right\}^{J+1} = \mathcal{N}\left\{\Phi^{(n)}_{\tau+1}\right\}^{J} + \Delta t^{-1}\mathcal{L}_1[2\Phi^{(n)}_\tau - (1/2)\Phi^{(n)}_{\tau-1}], \tag{4.19b}$$

$$[(11/6)\Delta t^{-1}\mathcal{L}_1 - \mathcal{L}_2]\left\{\Phi^{(n)}_{\tau+1}\right\}^{J+1} = \mathcal{N}\left\{\Phi^{(n)}_{\tau+1}\right\}^{J} + \Delta t^{-1}\mathcal{L}_1[3\Phi^{(n)}_\tau - (3/2)\Phi^{(n)}_{\tau-1} + (1/3)\Phi^{(n)}_{\tau-2}], \tag{4.19c}$$

$$[(25/12)\Delta t^{-1}\mathcal{L}_1 - \mathcal{L}_2]\left\{\Phi^{(n)}_{\tau+1}\right\}^{J+1} = \mathcal{N}\left\{\Phi^{(n)}_{\tau+1}\right\}^{J} + \Delta t^{-1}\mathcal{L}_1[4\Phi^{(n)}_\tau - 3\Phi^{(n)}_{\tau-1} + (4/3)\Phi^{(n)}_{\tau-2} - (1/4)\Phi^{(n)}_{\tau-3}], \tag{4.19d}$$

respectively, where the superscript $J+1$ denotes the current iteration and J denotes the previous iteration The values of the modal functions $\Phi^{(n)}_{\tau+1}$ on the right hand side of Eqs. (4.19a)–(4.19d) associated with the nonlinear terms are initialized with the values of $\Phi^{(n)}_\tau$, i.e., with the solution from the previous time step. The first-order implicit scheme is self-starting, and is used to start the second-, the third- and the fourth-order methods. All methods were found to be numerically stable.

Computational cost of the implicit schemes cannot be predicted beforehand, as it is dominated by the number of iterations required in order to solve the nonlinear algebraic equations with the specified accuracy. We have used the convergence criteria based on the absolute difference between two subsequent solutions, and this difference had been set to 10^{-14} in all tests reported in this paper. The required number of iterations is influenced by several factors including wave profiles, (e.g., their wave number, amplitude, phase speed, frequency), the number of Fourier modes and Chebyshev polynomials used in the spatial discretization, the type of temporal discretization and the time step-size.

Results of the temporal grid convergence studies for the model problem #2 are reported using the error defined as

$$u_{er} = \sup_{\substack{0\leqslant x\leqslant 2\pi/\alpha \\ y_L(x,t)\leqslant y\leqslant y_U(x,t)}} |u_{\text{difference}}(x,y,t)|, \tag{4.20}$$

where

$$u_{\text{difference}}(x,y,t) = u_{MP\#2}(x,y,t) - u_{MP\#1}(x,y,t). \tag{4.21}$$

Here the term $u_{MP\#2}$ refers to the solution obtained through the direct solution of the moving boundary problem in the fixed frame of reference and the term $u_{MP\#1}$ denotes solution of the corresponding fixed boundary problem in the moving frame of

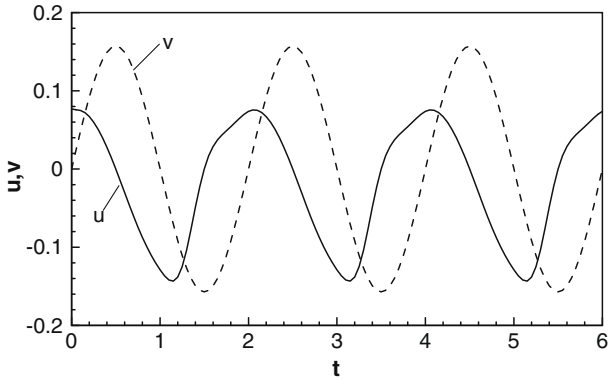

Fig. 19. The evolution of the u- (solid line) and v- (dashed line) velocity components at a test point $(x,y) = (\lambda/4, -0.9)$ during the first four cycles of the standing wave (model problem #3). All other conditions are as in Fig. 17.

reference and converted into the fixed frame of reference using the reversed Galileo transformation. The results shown in Fig. 20 corresponds to time $t = 1.0$ and demonstrate that all the temporal schemes deliver the theoretically predicted accuracy.

4.4. Computational efficiency and effectiveness

All implicit algorithms require solution of a large system of nonlinear algebraic equations at each time step. The nonlinear system is solved in an iterative manner, i.e., the nonlinear terms are moved to the right hand side and the resulting linear system is solved repetitively with the right hand side being updated after each iteration. The solution process requires evaluation of the inverse (or equivalent) of the coefficient matrix only once for the first iteration, with subsequent iterations reduced to multiplication of the inverse (or equivalent) with the recomputed right hand side vector. Because of the size of the system, one needs to evaluate the performance of various possible solution strategies of the relevant linear system.

The linear system has the form

$$\mathbf{Lx} = \mathbf{R}, \tag{4.22}$$

where $\mathbf{L}$ is a $p \times p$ coefficient matrix with $p = (2N_M + 1)(N_T + 1)$, $\mathbf{x}$ is a p-dimensional column vector of unknowns and $\mathbf{R}$ is a p-dimensional column vector representing information contained in the nonlinear terms and taken either from the previous time step or from the previous iteration. The system is organized by grouping entries corresponding to the linear part of the field equations in matrix $\mathbf{H}$ of size $q \times p$, $q = (2N_M + 1)(N_T - 3)$, and entries corresponding to boundary relations in matrix $\mathbf{K}$ of size $(p - q) \times p$, resulting in the structure of coefficient matrix $\mathbf{L}$ illustrated in Fig. 21. Matrix $\mathbf{H}$ has the block-diagonal structure with each block corresponding to a different modal equation and having the size $(N_T + 1) \times (N_T - 3)$. This matrix needs to be computed only ones as it does not depend on boundary motions. Matrix $\mathbf{K}$ is full as it provides coupling between different modes and it needs to be recomputed at each time step in order to capture boundary motions. Structure of matrix $\mathbf{L}$ illustrates advantage of the IBC algorithm. Algorithms based on the dynamic grid adjustments and/or mappings require an effort equivalent to evaluation of the complete coefficient matrix $\mathbf{L}$ at each time step while the IBC algorithm requires evaluation of only matrix $\mathbf{K}$ which represents a small portion of the complete matrix $\mathbf{L}$. The additional cost associated with the evaluation of the complete matrix can be illustrated using the domain transformation (DM) method discussed in the context of the test problem #1 as a representative of the methods requiring grid adjustment. Fig. 22 shows the ratio of time required to construct the coefficient matrix for the DT and the IBC methods using different numbers of Fourier modes N_M and Chebyshev polynomials N_T. The IBC method requires approximately 30 times less time for the smaller values of N_M and N_T and the savings increase almost linearly with the increase of either N_M or N_T. The cost of the matrix construction for the dynamic grid methods further significantly increases (relative to the IBC method) if the grid adjustment needs to rely on the numerical grid generation.

Eq. (4.22) can be solved directly using various algorithms, with potential efficiency gains associated with taking advantage of the sparse character of the coefficient matrix. We have tested four different methods in an effort to find an efficient solution strategy.

(i) *Method A:* This method relies on the LU decomposition. The lower and upper triangular matrices are computed at the beginning of each time step and the subsequent iterations are reduced to a simple backward eliminations and forward substitutions in a repetitive fashion.

(ii) *Method B:* The solution process for this method is also based on LU decomposition, however, it takes advantage of the sparse structure of the coefficient matrix $\mathbf{L}$ and uses specialized solver for computing LU factors of sparse matrices.

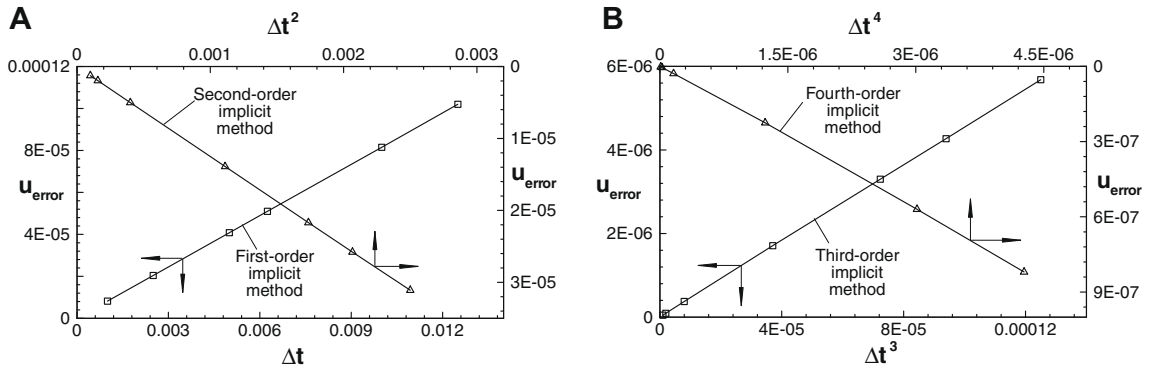

Fig. 20. Variations of the error u_{er} (see Eq. (4.20)) as a function of the time step Δt. Results shown were obtained for the travelling wave problem (model problem #2) with the wave amplitude $S = 0.05$, the wave number $\alpha = 1.0$ and the phase speed $c = \pi$. Computations have been carried out for the Reynolds number $Re = 100$ using $N_M = 10$ number of Fourier modes and $N_T = 100$ number of Chebyshev polynomials. Solution of the model problem #1 was taken as the initial condition. (A) First- and second-order methods and (B) third- and fourth-order methods.

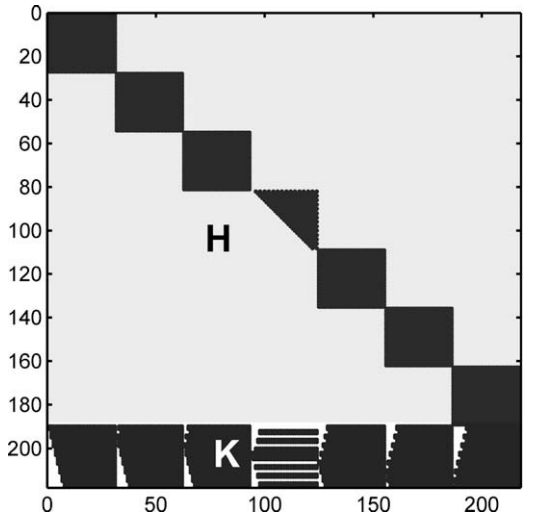

Fig. 21. Structure of the coefficient matrix for the model problem #2 constructed using $N_M = 3$ Fourier modes and $N_T = 30$ Chebyshev polynomials. Non-zero entries are marked in black.

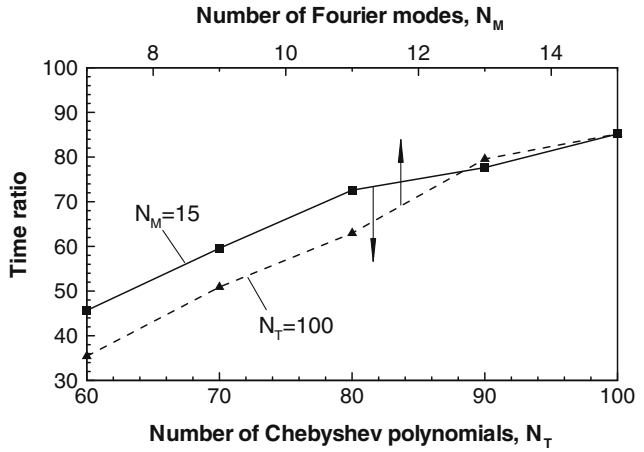

Fig. 22. Variations of the ratio of time required to construct the coefficient matrix using the DT method as compared with the IBC method for the model problem #1 as a function of the number of Fourier modes N_M (dash line, N_T=100) and as a function of the number of Chebyshev polynomials N_T (solid line, $N_M = 15$) used in the computations.

(iii) *Method C*: The inverse of the coefficient matrix **L** is computed once at the beginning of each time step and the subsequent iterations are reduced to the multiplication of the inverted matrix with the updated right hand side vector.
(iv) *Method D:* Similar to method C this method computes the inverse of the coefficient matrix using specialized procedures for sparse matrices.

Method A is used as the reference method in the following discussion. The "performance gain" achieved by other methods can be quantified as the ratio of computational time involved in advancing solution by one time step using method in question to that required by method A. Fig. 23 illustrates performance gains of methods B, C and D. It can be seen that method B is the most efficient while method D is the least efficient. The efficiency of methods B and C depends very little on the problem size, while the efficiency of method B increases almost linearly from ~1.15 to ~1.5 when increasing the number of blocks of the same size (Fig. 23a). The efficiency of this method is almost unchanged when one works with a constant number of blocks of increasing size (Fig. 23b).

When a large number of Fourier modes and/or Chebyshev polynomials are used, the resulting matrix could be very large leading to the computationally unacceptable solution cost, in spite of a minimal cost associated with the generation of matrix **L** at each time step. The solution cost can be reduced using iterative solver based on the mode decoupling concept [22,23]. It had been demonstrated that such solvers lead to a significant acceleration of computations in the case of flows governed by linear operators, with further efficiency gains possible through parallelization as the mode decoupling method

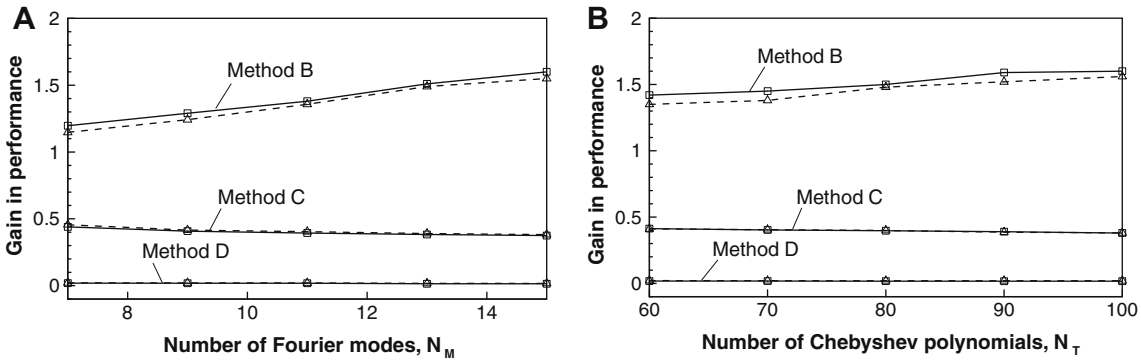

Fig. 23. Variations of the performance gains associated with the use of methods B, C and D for solution of the linear problem as a function of the number of Fourier modes N_M (A, $N_T = 100$) and as a function of the number of Chebyshev polynomials N_T (B, $N_M = 15$) used in the computations. The tests have been carried out for the model problem #2 using the third-order implicit discretization with $\alpha = 1.0$, $S = 0.05$, $Re = 100$ and $c = \pi$. The solid and dashed lines correspond to results obtained using time steps $\Delta t = 0.001$ and $\Delta t = 0.01$, respectively.

is very suitable for applications on distributed processors. On the negative side, the iterative algorithm limits the range of geometries that can be accessed due to convergence problems, as documented in the context of linear operators [22,23]. The convergence problems are more severe in the case of nonlinear operators, which further limit the range of geometries that can be modeled.

Because limitations associated with the iterative solver deemed to be too constraining, a new direct solver has been developed following concepts described in [26] where advantage was taken of the fact that part **H** of the matrix **L** does not change during time advancement. Part of this matrix can be inverted once and the overall solution can be reduced to a solution of a much smaller algebraic system supplemented by multiplication by the inverted matrix. The largest square matrix that can be extracted from **H** has the size $q \times q$. For that purpose, the vector of unknowns has been re-organized by re-locating the first four coefficients of Chebyshev expansions for each Fourier mode to the end of the vector resulting in the structure of coefficient matrix illustrated in Fig. 24. Square matrix **A** of size $q \times q$ has block-diagonal structure with each block of size $(N_T - 3) \times (N_T - 3)$, rectangular matrix **B** of size $q \times (p - q)$ has block-diagonal structure with each block of size $(N_T - 3) \times 4$, full rectangular matrix **C** has size $(p - q) \times q$ and the square matrix **D** has size $(p - q) \times (p - q)$. Matrices **A** and **B** remain unchanged during time advancement while **C** and **D** need to be recomputed. The system can be written as

$$\mathbf{A}\mathbf{x}_1 + \mathbf{B}\mathbf{x}_2 = \mathbf{R}_1, \qquad \mathbf{C}\mathbf{x}_1 + \mathbf{D}\mathbf{x}_2 = \mathbf{R}_2, \tag{4.23a,b}$$

where vector $\mathbf{x}_1$ contains unknowns $Z_k^{(n)}$ for $n \in \langle -N_M, N_M \rangle$, $k \in \langle 4, N_T \rangle$, and $\mathbf{x}_2$ contains unknowns $Z_k^{(n)}$ for $n \in \langle -N_M, N_M \rangle$, $k \in \langle 0, 3 \rangle$. The right hand side vector remains unchanged with $\mathbf{R}_1 + \mathbf{R}_2 = \mathbf{R}$, $\mathbf{R}_1$ having length q and $\mathbf{R}_2$ having

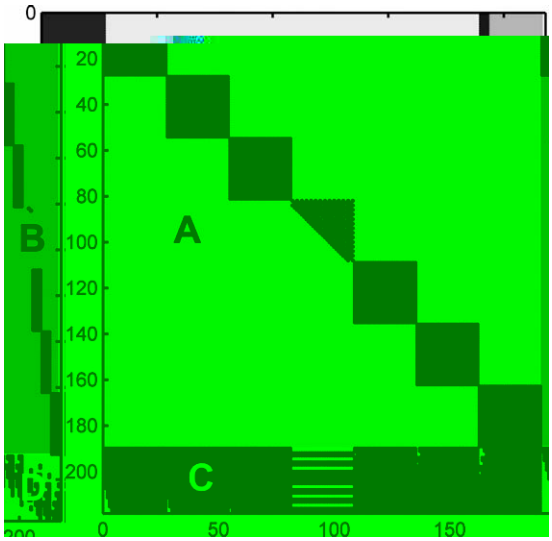

Fig. 24. Structure of the modified coefficient matrix for the model problem #2 constructed using $N_M = 3$ Fourier modes and $N_T = 30$ Chebyshev polynomials (see Eqs. (4.23–4.24)). Non-zero entries are marked in black.

length $(p-q)$. Matrices **B** and **D** contain information associated with $Z_0^{(n)}$, $Z_1^{(n)}$, $Z_2^{(n)}$ and $Z_3^{(n)}$, while matrices **A** and **C** contain the rest. Solution of (4.23) can be written as

$$\mathbf{x_2} = \mathbf{E^{-1}R_2} - \mathbf{E^{-1}CA^{-1}R_1}, \quad \mathbf{x_1} = \mathbf{A^{-1}R_1} - \mathbf{A^{-1}Bx_2}, \tag{4.24a, b}$$

where $\mathbf{E} = (\mathbf{D} - \mathbf{CA^{-1}B})$ has the size $(p-q)\times(p-q)$. Matrices $\mathbf{A}^{-1}$ and $\mathbf{A}^{-1}\mathbf{B}$ need to be computed only once and remain unchanged during time advancement; the main computational effort at each time step is reduced to solving a system of equation of the size $(p-q)\times(p-q)$ resulting in method E.

v) *Method E*: Method based on Eqs. (4.23)–(4.24) where all operations equivalent to evaluation of the inverse matrices are carried out using LU decomposition.

Fig. 25 shows that performance gains vary approximately from 1.5 to 5 depending on the severity of geometry and problem size, and are approximately the same regardless if one uses more smaller blocks (more Fourier modes, Fig. 25a) or works with fewer but larger blocks (more Chebyshev polynomials, Fig. 25b). The reader should note that an increase of severity of boundary motions (larger amplitude S of boundary motions) increases the number of iterations required per time step. Since the largest performance gains are associated with the first iteration, the performance gains for the more extreme wave profiles are less pronounced. In a similar way, use of smaller time steps reduces advantage of this method as it decreases the number of iterations per time step.

Method E can be further improved by taking advantage of the block-diagonal structure of matrix **A**, which results in method F.

vi) *Method F*: The inverses $\mathbf{A}^{-1}$ and $\mathbf{A}^{-1}\mathbf{B}$ are computed by inverting individual blocks (rather than inverting the whole matrices) using the LU decomposition.

Fig. 25 illustrates that performance gains associated with method F vary from 4 to 12 depending on the severity of geometry and problem size, similarly as in the case of method E. These gains increase rapidly with an increasing number of Fourier modes and Chebyshev polynomials, and are approximately the same regardless if one uses more smaller blocks (more Fourier modes, Fig. 25a) or works with fewer but larger blocks (more Chebyshev polynomials, Fig. 25b). Similarly to method E, this method performs slightly better for the less severe geometries and for smaller time step sizes, which result in a fewer iterations per time step.

It is of interest to summarize relative advantages of the IBC method as compared with the methods based on the dynamic grid adjustment represented in this discussion by the DT method. The IBC method requires the construction of the **H** matrix only once while the elements corresponding to the relatively smaller **K** matrix need to be computed at every time step. In contrast, the complete coefficient matrix needs to be computed at each time step for the DT method at an additional cost already discussed at the beginning of this section. Here the reader should note that the results presented in Fig. 22 compare costs of construction of the complete matrices for both methods while in the actual implementation a much smaller matrix K needs to be computed at each time step for the IBC method. Use of method F for solving the resulting linear system further increases the relative gains of the IBC method. The DT method leads to the full coefficient matrix and the cost of solution of the linear system is similar to the cost associated with method A of the IBC implementation; the relative gains of IBC methodology resulting from the use of specialized solver (method F) are illustrated in Fig. 25.

Memory requirement provides a second limitation for the applicability of the algorithm. The most common implementations of the IBC algorithm of the type discussed here rely on the construction of the complete matrix L [21–23] resulting in

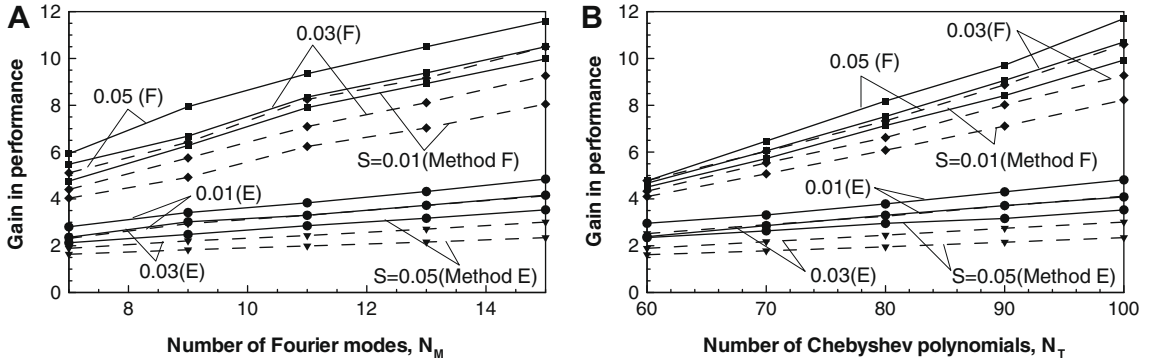

Fig. 25. Variations of the performance gains associated with the methods E (lower curves) and F (upper curves) as a function of the number of Fourier modes N_M (A, $N_T = 100$) and as a function of the number of Chebyshev polynomials N_T (B, $N_M = 15$) used nine in the computations. The tests have been carried out for the model problem #2 using the third-order implicit discretization with $\alpha = 1.0$, $Re = 100$, $c = \pi$, and different values of S. The solid and dashed lines correspond to results obtained using time steps $\Delta t = 0.001$ and $\Delta t = 0.01$, respectively.

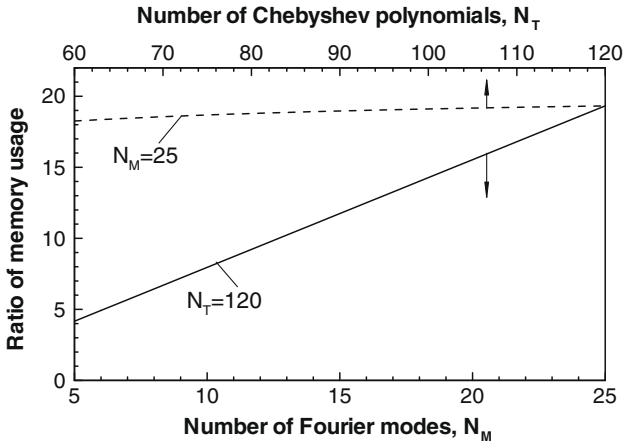

Fig. 26. Variations of the ratio of memory used by methods A and F as a function of the number of Chebyshev polynomials N_T (dash line, $N_M = 25$) and as function of the number of Fourier modes N_M (dash line, $N_T = 120$) used in the computations.

extreme memory requirements when a large number of Fourier modes and Chebyshev polynomials need to be used. Method F requires storage for matrices **B, C, D** and only for the diagonal blocks of matrix **A**. The memory requirement can be further reduced by noting that the information about the diagonal blocks can be overwritten with the information about the inverted blocks. Fig. 26 illustrates variations of the ratio of the memory use by methods A and F. It can be seen that in the case of $N_M = 25$ Fourier modes and $N_T = 120$ Chebyshev polynomials method F uses about 20 times less memory than method A. The memory gains increase almost linearly and very rapidly when the number of Fourier modes N_M increases (resulting in a larger number of blocks). The memory gains change very little as the number of Chebyshev polynomials N_T increases (resulting in larger blocks – see Fig. 26).

The final comment deals with the memory use associated with the methods based on the dynamic grid adjustment represented in this discussion by the DT method. This method leads to the full matrix and offers effectively no potential for memory savings.

5. Conclusions

A highly accurate algorithm to analyze unsteady flow problems associated with the presence of moving boundaries has been presented. The algorithm relies on the concept of immersed boundary conditions where the time-varying flow domain remains completely immersed inside a fixed computational domain during simulation time. The flow boundary conditions are imposed in the form of constraints. The algorithm uses Fourier expansions in the streamwise direction, Chebyshev expansions in the wall–normal direction and fully-implicit time discretization with up to fourth-order accuracy. Various tests demonstrate that the algorithm delivers spectral accuracy in space and the theoretically predicted accuracy in time. The entries in the coefficient matrix corresponding to the field equations remain unchanged during boundary motions due to the use of the immersed boundaries concept. As a result, only a small part of the coefficient matrix corresponding to boundary constraints needs to be recomputed at each time step resulting in a significant saving of computing resources. A very efficient linear solver that takes advantage of the structure of the coefficient matrix has been proposed. The proposed solver results in a significant acceleration of the computations as well as in a substantial reduction of memory requirements.

Acknowledgements

This work has been carried out with support of SHARCNET and NSERC of Canada. SHARCNET of Canada provided the computing resources.

Appendix A. *A.1. First-order implicit temporal discretization*

Temporal discretization of (2.15) using the first-order implicit method results in the following equation

$$\begin{aligned}&\left[\begin{aligned}&-Re^{-1}\Gamma^4\widehat{D}^4+\left(\Gamma^2Re^{-1}\Delta t^{-1}+2n^2\alpha^2\Gamma^2Re^{-1}+in\alpha\Gamma^2u_0\right)\widehat{D}^2\\&+\left(-n^4\alpha^4Re^{-1}-n^2\alpha^2\Gamma^2\Delta t^{-1}Re^{-1}-in\alpha\Gamma^2\widehat{D}^2u_0-in^3\alpha^3u_0\right)\end{aligned}\right]\\&\quad\times\left(\Phi^{(n)}_{\tau+1}\right)^{J+1}=\left[-in\alpha\Gamma\widehat{D}(R_1)^{(n)}_{\tau+1}-\Gamma^2\widehat{D}^2(R_2)^{(n)}_{\tau+1}-(n\alpha)^2(R_2)^{(n)}_{\tau+1}+in\alpha\Gamma\widehat{D}(R_3)^{(n)}_{\tau+1}\right]^J\\&\quad+Re^{-1}\Delta t^{-1}[\Gamma^2\widehat{D}^2-(n\alpha)^2]\Phi^{(n)}_{\tau},\quad n\in\langle 0,N_M\rangle.\end{aligned}\tag{A1}$$

A.2. Second-order implicit temporal discretization

Temporal discretization of (2.15) using the second-order implicit method results in the following equation

$$\begin{aligned}&\left[\begin{aligned}&-Re^{-1}\Gamma^4\widehat{D}^4+\left(3\Gamma^2Re^{-1}\Delta t^{-1}/2+2n^2\alpha^2\Gamma^2Re^{-1}+in\alpha\Gamma^2u_0\right)\widehat{D}^2\\&+\left(-n^4\alpha^4Re^{-1}-3n^2\alpha^2\Gamma^2\Delta t^{-1}Re^{-1}/2-in\alpha\Gamma^2\widehat{D}^2u_0-in^3\alpha^3u_0\right)\end{aligned}\right]\\&\times\left(\Phi^{(n)}_{\tau+1}\right)^{J+1}=\left[-in\alpha\Gamma\widehat{D}(R_1)^{(n)}_{\tau+1}-\Gamma^2\widehat{D}^2(R_2)^{(n)}_{\tau+1}-(n\alpha)^2(R_2)^{(n)}_{\tau+1}+in\alpha\Gamma\widehat{D}(R_3)^{(n)}_{\tau+1}\right]^J\\&+2Re^{-1}\Delta t^{-1}\left[\Gamma^2\widehat{D}^2-(n\alpha)^2\right]\Phi^{(n)}_{\tau}-0.5Re^{-1}\Delta t^{-1}\left[\Gamma^2\widehat{D}^2-(n\alpha)^2\right]\Phi^{(n)}_{\tau-1},\quad n\in\langle 0,N_M\rangle.\end{aligned}\tag{A2}$$

A.3. Fourth-order implicit temporal discretization

Temporal discretization of (2.15) using the fourth-order implicit method results in the following equation

$$\begin{aligned}&\left[\begin{aligned}&-Re^{-1}\Gamma^4\widehat{D}^4+\left(25\Gamma^2Re^{-1}\Delta t^{-1}/12+2n^2\alpha^2\Gamma^2Re^{-1}+in\alpha\Gamma^2u_0\right)\widehat{D}^2\\&+\left(-n^4\alpha^4Re^{-1}-25n^2\alpha^2\Gamma^2\Delta t^{-1}Re^{-1}/12-in\alpha\Gamma^2\widehat{D}^2u_0-in^3\alpha^3u_0\right)\end{aligned}\right]\left(\Phi^{(n)}_{\tau+1}\right)^{J+1}\\&=\left[-in\alpha\Gamma\widehat{D}(R_1)^{(n)}_{\tau+1}-\Gamma^2\widehat{D}^2(R_2)^{(n)}_{\tau+1}-(n\alpha)^2(R_2)^{(n)}_{\tau+1}+in\alpha\Gamma\widehat{D}(R_3)^{(n)}_{\tau+1}\right]^J\\&+4Re^{-1}\Delta t^{-1}[\Gamma^2\widehat{D}^2-(n\alpha)^2]\Phi^{(n)}_{\tau}-3Re^{-1}\Delta t^{-1}[\Gamma^2\widehat{D}^2-(n\alpha)^2]\Phi^{(n)}_{\tau-1}\\&+(4/3)Re^{-1}\Delta t^{-1}[\Gamma^2\widehat{D}^2-(n\alpha)^2]\Phi^{(n)}_{\tau-2}-(1/4)Re^{-1}\Delta t^{-1}\left[\Gamma^2\widehat{D}^2-(n\alpha)^2\right]\Phi^{(n)}_{\tau-2},\quad n\in\langle 0,N_M\rangle.\end{aligned}\tag{A3}$$

Appendix B. Algorithm for evaluation of coefficients $w^{(m)}_{k,\tau+1}$ and $d^{(m)}_{k,\tau+1}$ appearing in Eq. (3.4).

Coefficients in Eq. (3.4a) can be evaluated with the help of the recurrence relation $T_{k+1}(\hat{y})=2\hat{y}T_k(\hat{y})-T_{k-1}(\hat{y})$ that leads to

$$w^{(m)}_{k+1,\tau+1}=2\sum_{n=-\infty}^{n=+\infty}A^{(n)}_{\tau+1}w^{(m-n)}_{k,\tau+1}-w^{(m)}_{k-1,\tau+1}\quad\text{for } k>1,\tag{B1}$$

whose evaluation begins at $k=0$, giving the initial terms in the form

$$w^{(0)}_{0,\tau+1}=1, w^{(m)}_{0,\tau+1}=0\quad\text{for } |m|\geqslant 1;\qquad w^{(m)}_{1,\tau+1}=A^{(m)}_{\tau+1}\quad\text{for } |m|\geqslant 0.\tag{B2}$$

Coefficients in Eq. (3.4b) can be evaluated with the help of the recurrence relation $\mathrm{DT}_{k+1}(\hat{y})=2T_k(\hat{y})+2\hat{y}\mathrm{DT}_k(\hat{y})-\mathrm{DT}_{k-1}(\hat{y})$ that leads to a relation

$$d^{(m)}_{k+1,\tau+1}=2\sum_{n=-\infty}^{\infty}A^{(n)}_{\tau+1}d^{(m-n)}_{k,\tau+1}-d^{(m)}_{k-1,\tau+1}+2w^{(m)}_{k,\tau+1}\quad\text{for } k>2,\tag{B3}$$

whose evaluation begins at $k=0$, giving the initial terms in the form

$$d^{(m)}_{0,\tau+1}=0\quad\text{for } |m|\geqslant 0;\quad d^{(0)}_{1,\tau+1}=1,\quad d^{(m)}_{1,\tau+1}=0\quad\text{for } |m|\geqslant 1;\quad d^{(m)}_{2,\tau+1}=4A^{(m)}_{\tau+1}\quad\text{for } |m|\geqslant 0.\tag{B4}$$

Appendix C. Domain transformation method for the model problem#1 in the moving frame of reference

The irregular conduit geometry in the physical domain (X,y) is mapped into a straight conduit in the computational domain (ξ,η) using transformations in the form

$$\xi=X,\quad \eta=[2y-y_U(X)-y_L(X)]/[y_U(X)-y_L(X)].\tag{c1}$$

Application of transformations given in Eq. (A1) brings Eq. (4.4) into the form

$$\begin{aligned}&\partial_{\eta\eta\eta\eta}\Psi_1+B_1(\xi,\eta)\partial_{\eta\eta\eta}\Psi_1+B_2(\xi,\eta)\partial_{\eta\eta}\Psi_1+B_3(\xi,\eta)\partial_{\eta}\Psi_1+B_4(\xi,\eta)\partial_{\xi\eta\eta\eta}\Psi_1+B_5(\xi,\eta)\partial_{\xi\eta\eta}\Psi_1\\&+B_6(\xi,\eta)\partial_{\xi\eta}\Psi_1+B_7(\xi,\eta)\partial_{\xi\xi\eta}\Psi_1+B_8(\xi,\eta)\partial_{\xi\xi\eta\eta}\Psi_1+B_9(\xi,\eta)\partial_{\xi\xi\xi\eta}\Psi_1+B_{10}(\xi,\eta)\partial_{\xi}\Psi_1\\&+B_{11}(\xi,\eta)\partial_{\xi\xi\xi}\Psi_1+B_{12}(\xi,\eta)\partial_{\xi\xi\xi\xi}\Psi_1=L_1(\xi,\eta)\partial_{\xi\eta}\langle N_1\rangle+L_2(\xi,\eta)\partial_{\eta}\langle N_1\rangle+L_3(\xi,\eta)\partial_{\eta\eta}\langle N_1\rangle\\&+L_4(\xi,\eta)\partial_{\eta\eta}\langle N_2\rangle+L_5(\xi,\eta)\partial_{\eta}\langle N_2\rangle+L_6(\xi,\eta)\partial_{\xi\eta}\langle N_2\rangle+L_5(\xi,\eta)\partial_{\xi\eta}\langle N_2\rangle,\end{aligned}\tag{c2}$$

where

$$\langle N_1\rangle = \langle u_1 u_1\rangle - \langle v_1 v_1\rangle, \quad \langle N_2\rangle = \langle u_1 v_1\rangle, \tag{C3a-b}$$

$$B_0(\xi,\eta) = \eta_X^4 + 2\eta_X^2\eta_y^2 + \eta_y^4, \tag{C4a}$$
$$B_1(\xi,\eta) = \left[6\eta_X^2\eta_{XX} + 2\eta_{XX}\eta_y^2 + 8\eta_X\eta_y\eta_{Xy} + (c-u_0Re)\left(\eta_X^3 - \eta_X\eta_y^2\right)\right]/B_0(\xi,\eta), \tag{C4b}$$
$$B_2(\xi,\eta) = \left[3\eta_{XX}^2 + 4\eta_X\eta_{XXX} + 4\eta_{XXy}\eta_y + 4\eta_{Xy}^2 + (c-u_0Re)\left(3\eta_X\eta_{XX} + 2\eta_{Xy}\eta_y\right)\right]/B_0(\xi,\eta), \tag{C4c}$$
$$B_3(\xi,\eta) = [\eta_{XXXX} + c\eta_{XXX} - Re(u_0\eta_{XXX} + 2\eta_X)]/B_0(\xi,\eta), \tag{C4d}$$
$$B_4(\xi,\eta) = \left(4\eta_X^3 + 4\eta_X\eta_y^2\right)/B_0(\xi,\eta), \tag{C4e}$$
$$B_5(\xi,\eta) = \left[12\eta_X\eta_{XX} + 8\eta_{Xy}\eta_y + (c-u_0Re)\left(3\eta_X^2 + \eta_y^2\right)\right]/B_0(\xi,\eta), \tag{C4f}$$
$$B_6(\xi,\eta) = (4\eta_{XXX} + 3(c-u_0Re)\eta_{XX})/B_0(\xi,\eta), \tag{C4g}$$
$$B_7(\xi,\eta) = (6\eta_{XX} + 3(\mathrm{c}-\mathrm{u_0}Re)\eta_X)/B_0(\xi,\eta), \tag{C4h}$$
$$B_8(\xi,\eta) = \left(6\eta_X^2 + 2\eta_y^2\right)/B_0(\xi,\eta), \tag{C4i}$$
$$B_9(\xi,\eta) = 4\eta_X/B_0(\xi,\eta), \tag{C4j}$$
$$B_{10}(\xi,\eta) = -2Re/B_0(\xi,\eta), \tag{C4k}$$
$$B_{11}(\xi,\eta) = (c-u_0Re)/B_0(\xi,\eta), \tag{C4l}$$
$$B_{12}(\xi,\eta) = 1/B_0(\xi,\eta), \tag{C4m}$$

$$L_1(\xi,\eta) = Re\eta_y/B_0(\xi,\eta), \tag{C5a}$$
$$L_2(\xi,\eta) = Re\eta_{Xy}/B_0(\xi,\eta), \tag{C5b}$$
$$L_3(\xi,\eta) = Re\eta_X\eta_{Xy}/B_0(\xi,\eta), \tag{C5c}$$
$$L_4(\xi,\eta) = Re\left(\eta_y^2 - \eta_X^2\right)/B_0(\xi,\eta), \tag{C5d}$$
$$L_5(\xi,\eta) = -Re\eta_{XX}/B_0(\xi,\eta), \tag{C5e}$$
$$L_6(\xi,\eta) = -2Re\eta_X/B_0(\xi,\eta), \tag{C5f}$$
$$L_7(\xi,\eta) = -Re/B_0(\xi,\eta). \tag{C5g}$$

In the above

$$\eta_X = -\left[(1+\eta)(y_U)_\xi + (1-\eta)(y_L)_\xi\right]/(y_U - y_L), \tag{C6a}$$
$$\eta_{XX} = -\left[2\eta_x\{(y_U)_\xi - (y_L)_\xi\} + (1+\eta)(y_U)_{\xi\xi} + (1-\eta)(y_L)_{\xi\xi}\right]/(y_U - y_L), \tag{C6b}$$
$$\eta_{XXX} = -\left[\begin{array}{l} 3\eta_{xx}\{(y_U)_\xi - (y_L)_\xi\} + 3\eta_x\{(y_U)_{\xi\xi} - (y_L)_{\xi\xi}\} + \\ (1+\eta)(y_U)_{\xi\xi\xi} + (1-\eta)(y_L)_{\xi\xi\xi} \end{array}\right]/(y_U - y_L), \tag{C6c}$$
$$\eta_{XXXX} = -\left[\begin{array}{l} 4\eta_{xxx}\{(y_U)_\xi - (y_L)_\xi\} + 6\eta_{xx}\{(y_U)_{\xi\xi} - (y_L)_{\xi\xi}\} + \\ 4\eta_x\{(y_U)_{\xi\xi\xi} - (y_L)_{\xi\xi\xi}\} + (1+\eta)(y_U)_{\xi\xi\xi\xi} + (1-\eta)(y_L)_{\xi\xi\xi\xi} \end{array}\right]/(y_U - y_L), \tag{C6d}$$
$$\eta_y = 2/(y_U - y_L), \tag{C6e}$$
$$\eta_{Xy} = -\left[\eta_y\{(y_U)_\xi - (y_L)_\xi\}\right]/(y_U - y_L), \tag{C6f}$$
$$\eta_{XXy} = -\left[2\eta_{xy}\{(y_U)_\xi - (y_L)_\xi\} + \eta_y\{(y_U)_{\xi\xi} - (y_L)_{\xi\xi}\}\right]/(y_U - y_L), \tag{C6g}$$
$$\eta_{XXXy} = -\left[3\eta_{xxy}\{(y_U)_\xi - (y_L)_\xi\} + 3\eta_{xy}\{(y_U)_{\xi\xi} - (y_L)_{\xi\xi}\} + \eta_y\{(y_U)_{\xi\xi\xi} - (y_L)_{\xi\xi\xi}\}\right]/(y_U - y_L). \tag{C6h}$$

Definition of the stream function given by Eq. (2.8) remains unchanged and thus the velocity components can be expressed as

$$\begin{aligned} u(\xi,\eta) &= u_0(\xi,\eta) + u_1(\xi,\eta) = u_0(\xi,\eta) + \eta_y\partial_\eta\Psi_1 = \eta_y\partial_\eta\Psi_T, \\ v(\xi,\eta) &= v_1(\xi,\eta) = -\partial_\xi\Psi_1 - \eta_x\partial_\eta\Psi_1 = -\partial_\xi\Psi_T - \eta_x\partial_\eta\Psi_T. \end{aligned} \tag{C7a-b}$$

The boundary conditions for the problem are given by the following relations

$$\begin{aligned} &\eta_y\partial_\eta\Psi_1(\xi,-1) = -u_0(\xi,-1), \quad \eta_y\partial_\eta\Psi_1(\xi,+1) = -u_0(\xi,+1), \\ &\partial_\xi\Psi_1(\xi,-1) + \eta_x\partial_\eta\Psi_1(\xi,-1) = (y_L)' \quad \partial_\xi\Psi_1(\xi,+1) + \eta_x\partial_\eta\Psi_1(\xi,+1) = (y_U)'. \end{aligned} \tag{C8a-b}$$

Solution of (c2) is assumed in terms of a Fourier expansion, i.e.,

$$\Psi_1(\xi,\eta) = \sum_{n=-\infty}^{n=+\infty} \phi^{(n)}(\eta) e^{in\alpha\xi} \approx \sum_{n=-N_M}^{n=+N_M} \phi^{(n)}(\eta) e^{in\alpha\xi} \tag{C9}$$

where $\phi^{(n)} = \phi^{(-n)*}$ and star denotes complex conjugates. The coefficients $B_N(\xi,\eta),\ N = 1,\ldots,12$ and $L_M(\xi,\eta),\ M = 1,\ldots,7$ are be replaced by the Fourier expansions in the form

$$B_N(\xi,\eta) = \sum_{m=-2N_M}^{m=+2N_M} b_N^{(m)}(\eta) e^{im\alpha\xi}, \quad L_M(\xi,\eta) = \sum_{m=-2N_M}^{m=+2N_M} g_M^{(m)}(\eta) e^{im\alpha\xi}. \tag{C10}$$

The nonlinear terms $\langle N_1\rangle$ and $\langle N_2\rangle$ are also expressed as Fourier expansions, i.e.,

$$\langle N_1\rangle(\xi,\eta) = \sum_{n=-N_M}^{n=+N_M} \gamma_1^{(n)}(\eta) e^{in\alpha\xi}, \quad \langle N_2\rangle(\xi,\eta) = \sum_{n=-N_M}^{n=+N_M} \gamma_2^{(n)}(\eta) e^{in\alpha\xi}. \tag{C11}$$

Substitution of C9, C10 and C11 into (c2) and separation of Fourier components lead to the modal equations in the form

$$D^4\phi^{(n)} + \sum_{m=-N_M}^{m=+N_M} \left\{ \begin{array}{l} [b_1^{(n-m)} + (im\alpha)b_4^{(n-m)}]D^3\phi^{(m)} \\ +[b_2^{(n-m)} + (im\alpha)b_5^{(n-m)} - (m\alpha)^2 b_8^{(n-m)}]D^2\phi^{(m)} \\ +[b_3^{(n-m)} + (im\alpha)b_6^{(n-m)} - (m\alpha)^2 b_7^{(n-m)} - i(m\alpha)^3 b_9^{(n-m)}]D\phi^{(m)} \\ +[(im\alpha)b_{10}^{(n-m)} - i(m\alpha)^3 b_{11}^{(n-m)} + (m\alpha)^4 b_{12}^{(n-m)}]\phi^{(m)} \end{array} \right\}$$
$$= \sum_{m=-N_M}^{m=+N_M} \left\{ \begin{array}{l} [g_2^{(n-m)} + (im\alpha)g_1^{(n-m)}]D\gamma_1^{(m)} + [g_3^{(n-m)}]D^2\gamma_1^{(m)} \\ [g_4^{(n-m)}]D^2\gamma_2^{(m)} + [g_5^{(n-m)} + (im\alpha)g_6^{(n-m)}]D\gamma_2^{(m)} \\ +[-(m\alpha)^2 g_7^{(n-m)}]\gamma_2^{(m)} \end{array} \right\}, \tag{C12}$$

where $D = d/d\eta$ and $n = 0,\ldots,N_M$, with each equation requiring four boundary conditions.

Boundary conditions (C8) need to be re-arranged into a more suitable form. The known reference flow at the boundaries $u_0(\xi,\pm1)$ can expressed as Fourier expansions in the form

$$u_0(\xi,-1) = \sum_{n=-N_M}^{n=+N_M} \widehat{F}_L^{(n)} e^{in\alpha\xi}, \quad u_0(\xi,+1) = \sum_{n=-N_M}^{n=+N_M} \widehat{F}_U^{(n)} e^{in\alpha\xi}. \tag{C13a, b}$$

Boundary conditions (C8c–d) can be re-arranged with the help of (C8a–b) into the following form

$$\partial_\xi\Psi(\xi,-1) = (\eta_x/\eta_y)u_0(\xi,-1) - c\sum_{n=-N_M,n\neq0}^{n=+N_M} in\alpha H_L^{(n)} e^{in\alpha\xi},$$
$$\partial_\xi\Psi(\xi,+1) = (\eta_x/\eta_y)u_0(\xi,+1) - c\sum_{n=-N_M,n\neq0}^{n=+N_M} in\alpha H_U^{(n)} e^{in\alpha\xi}. \tag{C14a-b}$$

All the terms in the right hand side of (C14) are known and thus can be expanded using Fourier series with known coefficients as

$$(\eta_x/\eta_y)u_0(\xi,-1) - c\sum_{n=-N_M,n\neq0}^{n=+N_M} in\alpha H_L^{(n)} e^{in\alpha\xi} = \sum_{n=-N_M}^{n=+N_M} \widehat{E}_L^{(n)} e^{in\alpha\xi},$$
$$(\eta_x/\eta_y)u_0(\xi,+1) - c\sum_{n=-N_M,n\neq0}^{n=+N_M} in\alpha H_U^{(n)} e^{in\alpha\xi} = \sum_{n=-N_M}^{n=+N_M} \widehat{E}_U^{(n)} e^{in\alpha\xi}. \tag{C15a-b}$$

The boundary conditions can now be expressed in terms of the unknown modal functions $\phi^{(n)}(\eta)$ in the form

$$D\phi^{(n)}(-1) = -\widehat{F}_L^{(n)} \quad \text{for } |n| \geqslant 0, \quad D\phi^{(n)}(+1) = -\widehat{F}_U^{(n)} \quad \text{for } |n| \geqslant 0,$$
$$\phi^{(n)}(-1) = \widehat{E}_L^{(n)}/(in\alpha) \quad \text{for } n\neq0, \quad \phi^{(n)}(+1) = \widehat{E}_U^{(n)}/(in\alpha) \quad \text{for } n\neq0. \tag{C16a-d}$$

Two more conditions are required to close the problem formulation. One condition is arbitrary and is associated with the definition of the stream function. The constant mass flux constraint has been selected as the second condition for this model problem. Eqs. (C7) and (C8) lead to

$$\partial_\xi\Psi_T(\xi,-1) = c\sum_{n=-N_M,n\neq0}^{n=+N_M} in\alpha H_L^{(n)} e^{in\alpha\xi}, \tag{C17}$$

where $\eta_x \partial_\eta \Psi_T(\xi,-1) = u(\xi,-1) = 0$. Integrating the above equation along ξ results in

$$\Psi_1(\xi,-1) = c \sum_{n=-N_M, n\neq 0}^{n=+N_M} H_L^{(n)} e^{in\alpha\xi} - \Psi_0(\xi,-1) + (const)_L. \tag{C18a}$$

Similar expression for the upper wall takes the form

$$\Psi_1(\xi,+1) = c \sum_{n=-N_M, n\neq 0}^{n=+N_M} H_U^{(n)} e^{in\alpha\xi} - \Psi_0(\xi,+1) + (const)_U. \tag{C18b}$$

As the definition of stream function is accurate up to a constant, we can arbitrarily assign the value of the stream function to be zero at $\xi = 0$, which results in

$$(const)_L = -c \sum_{n=-N_M, n\neq 0}^{n=+N_M} H_L^{(n)}. \tag{C19}$$

Volume flux Q along the conduit can be evaluated by integrating the x-velocity component across the conduit, i.e.,

$$Q(X) = \int_{y_L}^{y_U} u(X,y)dy. \tag{C20}$$

In the (ξ,η) coordinates the expression for the volume flux Q becomes

$$Q(\xi) = \int_{-1}^{+1} (\partial_\eta \Psi) d\eta = \Psi_0(\xi,+1) + \Psi_1(\xi,+1) - \Psi_0(\xi,-1) - \Psi_1(\xi,-1), \tag{C21}$$

where the $\Psi_0(\xi,-1)$ and $\Psi_0(\xi,+1)$ are known from the solution of the reference Poiseuille flow and can be expressed in terms of Fourier series as

$$\Psi_0(\xi,-1) = \sum_{n=-N_M}^{n=+N_M} \mathcal{P}_L^{(n)} e^{in\alpha\xi}, \quad \Psi_0(\xi,+1) = \sum_{n=-N_M}^{n=+N_M} \mathcal{P}_U^{(n)} e^{in\alpha\xi}. \tag{C22}$$

The volume flux can be written in the form of Fourier expansion

$$Q(\xi) = \sum_{n=-N_M}^{n=+N_M} \widehat{Q}^{(n)} e^{in\alpha\xi}, \tag{C23}$$

where the zero term, i.e., $\widehat{Q}^{(0)}$, represents the net mass flux along the conduit. Substitution of Eqs. (C22), (C23) and (C19) into Eq. (C21) and extraction of mode zero results in

$$(const)_U = \widehat{Q}^{(0)} - c \sum_{n=-N_M, n\neq 0}^{n=+N_M} H_L^{(n)}. \tag{C24}$$

Substitution of Eqs. (C9), (C19), (C22) and (C24) into Eq. (C18) and separation of mode zero results in the two closing boundary conditions in the form

$$\begin{aligned} \phi^{(0)}(-1) &= P_L^{(0)} - c \sum_{n=-N_M, n\neq 0}^{n=+N_M} H_L^{(n)}, \\ \phi^{(0)}(+1) &= P_U^{(0)} - c \sum_{n=-N_M, n\neq 0}^{n=+N_M} H_L^{(n)} + \widehat{Q}^{(0)}. \end{aligned} \tag{C25a,b}$$

The unknown $\phi^{(n)}(\eta)$ can be expressed with spectral accuracy using Chebyshev expansion in the form

$$\phi^{(n)}(\eta) = \sum_{k=0}^{k=\infty} S_k^{(n)} T_k(\eta) \approx \sum_{k=0}^{k=N_T} S_k^{(n)} T_k(\eta). \tag{C26}$$

Application of Galerkin procedure to Eq. (C12), as described in Section 3, leads to $N_T - 3$ algebraic equations in terms of the unknown coefficients $S_k^{(n)}$ for each Fourier mode. The remaining closing conditions come from discretization of the boundary conditions given by Eqs. (C16) and (C25). Solution of the complete problem involves an iterative process.

References

[1] J.M. Floryan, H. Rasmussen, Numerical analysis of viscous flows with free surfaces, Appl. Mech. Rev. 42 (1989) 323–341.
[2] R. Scardovelli, S. Zaleski, Direct numerical simulation of free surface and interfacial flow, Annu Rev. Fluid Mech. 31 (1999) 567–603.
[3] J.M. Hyman, Numerical methods for tracking of interfaces, Physica 12D (1984) 396–407.

[4] J. Glimm, J.W. Grove, X.L. Li, K.M. Shyue, Y. Zeng, Q. Zhang, Three-dimensional front tracking, SIAM J. Sci. Comput. 19 (1998) 703–727.
[5] C.W. Hirt, B.D. Nichols, Volume of fluid (VOF) method for the dynamics of free boundaries, J. Comput. Phys. 39 (1981) 201–225.
[6] F.H. Harlow, J.E. Welch, Numerical study of large amplitude free surface motions, Phys. Fluids 9 (1966) 842–851.
[7] S.J. Osher, J.A. Sethian, Fronts propagating with curvature dependent speed: algorithms based on Hamilton–Jacobi formulations, J. Comput. Phys. 79 (1988) 12–49.
[8] J.A. Sethian, P. Smereka, Level set methods for fluid interfaces, Annu. Rev. Fluid Mech. 35 (2003) 341–372.
[9] I. Inculet, J.M. Floryan, R. Haywood, Dynamics of water droplets break-up in electric fields, IEEE Trans. Indus. Appl. 28 (1992) 1203–1209.
[10] M. Hamed, J.M. Floryan, Numerical simulation of unsteady nonisothermal capillary interfaces, J. Comput. Phys. 145 (1998) 110–140.
[11] C.S. Peskin, The fluid dynamics of heart valves: experimental, theoretical and computational methods, Annu. Rev. Fluid Mech. 14 (1982) 235–259.
[12] C.S. Peskin, The immersed boundary method, Acta Numer. (2002) 479–517.
[13] R. Mittal, G. Iaccarino, Immersed boundary methods, Annu. Rev. Fluid Mech. 37 (2005) 239–261.
[14] J. Deng, X.M. Shao, A.L. Ren, A new modification of the immersed-boundary method for simulating flows with complex moving boundaries, Int. J. Numer. Methods Fluids 52 (2006) 1195–1213.
[15] K. Taira, T. Colonius, The immersed boundary method: a projection approach, J. Comput. Phys. 225 (2007) 2118–2137.
[16] J. Kim, D. Kim, H. Choi, An immersed-boundary finite-volume method for simulation of flows in complex geometries, J. Comput. Phys. 171 (2001) 132–150.
[17] S. Marella, S. Krishnan, H. Liu, H.S. Udaykumar, Sharp interface cartesian grid method I: an easily implemented technique for 3D moving boundary computations, J. Comput. Phys. 210 (2005) 1–31.
[18] J. Szumbarski, J.M. Floryan, A direct spectral method for determination of flows over corrugated boundaries, J. Comput. Phys. 153 (1999) 378–402.
[19] J.M. Floryan, Centrifugal instability of couette flow over a wavy wall, Phys. Fluids 14 (2002) 312–322.
[20] J.M. Floryan, Three-dimensional instabilities of laminar flow in a rough channel and the concept of hydraulically smooth wall, Eur. J. Mech. B/Fluids 26 (2007) 305–329.
[21] S.Z. Husain, J.M. Floryan, Immersed boundary conditions method for unsteady flow problems described by the Laplace operator, Int. J. Numer. Methods Fluids 56 (2008) 1765–1786.
[22] S.Z. Husain, J.M. Floryan, Implicit spectrally-accurate method for moving boundary problems using immersed boundary conditions concept, J. Comput. Phys. 227 (9) (2008) 4459–4477.
[23] S.Z. Husain, J.M. Floryan, Gridless spectral algorithm for Stokes flow with moving boundaries, Comput. Methods Appl. Mech. Eng. (2008). doi:10.1016/j.cma.2008.07.007.
[24] C. Canuto, M.Y. Hussaini, A. Quarteroni, T.A. Zang, Spectral Methods in Fluid Dynamics, Springer, 1987.
[25] S.Z. Husain, J. Szumbarski, J.M. Floryan, Over-constrained formulation of the immersed boundary condition method, Computer Methods in Applied Mechanics and Engineering 199 (2009) 94–112.
[26] S.Z. Husain, J.M. Floryan, An efficient linear solver for problems arising from the spectral implementation of the immersed boundary conditions method, Expert Systems in Fluid Dynamics Research Laboratory Report ESFD-2/2008, Department of Mechanical and Materials Engineering, The University of Western Ontario, London, Ontario, N6A 5B9 Canada, 2008.