Computers & Fluids 108 (2015) 92–106

Contents lists available at ScienceDirect

Computers & Fluids

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

Direct Numerical Simulation of turbulent flow in pipes with an arbitrary roughness topography using a combined momentum–mass source immersed boundary method

A.T. van Nimwegen, K.C.J. Schutte, L.M. Portela *

Department of Chemical Engineering, Delft University of Technology, Julianalaan 136, Delft, The Netherlands

ARTICLE INFO

Article history:
Received 17 April 2014
Received in revised form 29 August 2014
Accepted 3 November 2014
Available online 13 November 2014

Keywords:
Immersed boundary method
Roughness
Pipe flow
Turbulence
DNS

ABSTRACT

In this work, we consider a combined momentum–mass source immersed boundary method that can be used to easily simulate flows in domains with arbitrary wall topographies, taking advantage of existing standard fast solvers for straight walls. This method is incorporated into a standard second-order finite-volume code with a Fast Fourier Transform solver, previously used for direct numerical simulations of flow in pipes with a straight wall, making it possible to study the influence of a large range of roughness topographies on the flow. The method is validated for both laminar and turbulent flow over walls with sinusoidal undulations. The implementation of the immersed boundary method preserves the second-order accuracy of the original code, and first-order methods are presented to calculate the shear-stress and the pressure-drag at the wall. Results for turbulent flow over walls represented by a single wave-number, as well as by a superposition of many wave-numbers, show that the flow in the outer layer is not affected by the nature of the wall topography, provided that the variation in the diameter is not too large. Furthermore, it is found that the relative contributions of the pressure and the shear to the total drag on the wall behave similarly to undulated channel flow, when considered as a function of an effective slope parameter.

1. Introduction

In the past decade, immersed boundary methods (IBM) have frequently been used to simulate flow over arbitrary geometries. The method was originally developed by Peskin [1] to simulate the flow around heart valves; a summary of more recent developments is given in e.g. the review paper by Mittal and Iaccarino [2]. Traditionally, body-conformal grids are used in flow simulations; for simple geometries, e.g. pipes or channels, generating such grids is straightforward; in more complex geometries, however, generating a suitable grid can be time-consuming. Furthermore, non-structured grids, in general, do not allow the use of Fast Fourier Transform (FFT) solvers for the Poisson equation, thereby significantly increasing the computational resources required. Using an IBM, there is no need for the generation of a body-conformal grid; instead, a regular grid is used in a regular computational domain, regardless of the actual geometry. The effects of the solid boundary are incorporated by exerting a body force on the fluid inside (or near) this boundary. This saves time, since it is not necessary to generate a complex grid, and enables the use of fast solvers.

Two main types of IBM can be distinguished: the continuous-forcing and the direct-forcing methods. In the continuous-forcing method, the body force is modelled as a (damped) oscillator [3] or a porous medium [4,5]. The coefficients (spring constant and effective viscosity) are continuously adjusted to fit the boundary conditions. This method is often used in biological systems, where the walls are elastic, as for example by Peskin [1]. When simulating rigid boundaries, however, the large stiffness of the oscillator, required to accurately model the solid boundary, can lead to numerical instabilities [6]. To avoid this difficulty, Mohd-Yusof [7] and Verzicco et al. [8] developed the direct-forcing method, where the boundary conditions at the immersed boundary are directly imposed. Examples of implementation of the direct-forcing method are given by Tseng and Ferziger [9] and Bhaganagar and Hsu [10]. This method can be enhanced by including mass sources in the continuity equation, to improve mass conservation at the immersed boundary, as proposed by Parnaudeau et al. [11,12] for finite-difference codes and by Kim et al. [13] for finite-volume codes.

In this work, the IBM with momentum and mass sources by Kim et al. [13] was implemented into an existing second-order

* Corresponding author. Tel.: +31 15 2784387.
E-mail addresses: a.t.vannimwegen@tudelft.nl (A.T. van Nimwegen), k.c.j.schutte@tudelft.nl (K.C.J. Schutte), l.portela@tudelft.nl (L.M. Portela).

http://dx.doi.org/10.1016/j.compfluid.2014.11.008
0045-7930/

finite-volume code for direct numerical simulations (DNS) of flow in pipes with straight walls, written by Eggels [14], which uses an FFT solver for the Poisson equation. This IBM is a direct-forcing method, and was developed for simulations with rigid stationary boundaries. To illustrate the potential of the method, simulations of flow in pipes with arbitrary roughness topographies are performed in this paper. The method has also been used to simulate moving boundaries [15–17]. This does require a change in the calculation of the mass sources to account for the movement of the boundary [16,17]. Furthermore, freshly-cleared-cells, which are located inside the fluid, but were inside the boundary in the previous time-step have to be considered separately [2]. To our knowledge, the method has never been applied to flexible boundaries; although in principle it would be possible, the current method would have to be adjusted further to model the elasticity.

In pipe flow, the pressure-gradient is balanced by the drag exerted on the flow by the wall, and any arbitrary roughness topography will change the pressure-gradient in the flow. The influence of stochastic roughness on the pressure-gradient in turbulent pipe flow has been extensively studied by Nikuradse [18], who created roughness by gluing sand grains of varying size to the wall of a pipe. The results of this research are summarized in the Moody diagram, from which the friction factor of a pipe can be determined from the Reynolds number of the flow and the relative roughness height. More regular, yet non-flat, topographies also have an effect on the drag that the fluid exerts on the wall and, in this sense, they can also be considered roughness. Research has shown, however, that the effect of ‘roughness’ with such a regular topography on the pressure-gradient is not necessarily well represented by the Moody diagram, e.g. because different roughness elements can shelter each other [19].

Turbulent flows over regular roughness have been studied both experimentally [19–23] and numerically; most of the numerical studies use body-conformal grids in channel geometries [24–29]. Blackburn et al. [30] performed simulations with a wall with a single sinusoidal undulation, for a turbulent pipe flow; their results will be used in the validation of our implementation.

There are also some examples of earlier research where immersed boundary methods were used for simulating flow over regular roughness elements. All these studies consider a channel geometry, and do not use mass sources in the continuity equation to improve mass conservation near the wall. The sinusoidal topographies considered consist of a single sinusoidal undulation [9] or a multiplication of two undulations [10,31]. Finally, Anderson [32] considered the flow over many different surfaces (prisms, mounds, sinusoids, “fractal-like” surfaces, as well as a synthetic city) using an IBM in which the body force is based on effective drag coefficients of the roughness elements. Yuan and Piomelli [33] used a method based on the volume-of-fluid approach [34] for Large-Eddy Simulations (LES) of the flow over surfaces with sand grain roughness and over homogeneous roughness with a parabolic autocorrelation function.

In this work, we perform DNS of turbulent flows in (i) pipes with simple wall-topographies, consisting of a single sinusoidal undulation, and (ii) pipes with complex wall-topographies, consisting of a superposition of sinusoidal undulations, to mimic real roughness. The wall-topographies are implemented using an IBM that includes mass sources. The complex wall-topographies are similar to those considered by Napoli et al. [27–29] for channel flow, using body-conformal grids. Furthermore, we present a method of calculating the shear and pressure-drag at the wall, and compare the results of the calculations with those of Napoli et al. [27].

The structure of the paper is as follows. In Section 2, the original code for pipe flow with straight walls and the immersed boundary method are discussed. Section 3 covers the methodology to compute the pressure and shear-drag at the wall. In Section 4, the method is validated for both laminar and turbulent flows, and, subsequently, in Section 5, the order of the method is discussed. Results of simulations of simple and complex wall-topographies are presented and discussed in Section 6. Finally, in Section 7, we make some concluding remarks.

2. Implementation of the immersed boundary method

The finite-volume DNS pipe-flow solver developed by Eggels [14] uses a staggered grid for the three velocity components (the radial velocity u, the circumferential velocity v and the axial velocity w) and the pressure. The *fractional-step method* is used to solve the discrete Navier–Stokes equations. In this method, when advancing the time-step, the new velocity field is approximated in the predictor step by assuming a constant pressure-gradient $\overline{\nabla P}$:

$$\frac{\vec{U}^{*}-\vec{U}^{n}}{2\Delta t}+\mathbf{A}_h\left(\vec{U}^{n}\right)=-\frac{1}{\rho}\overline{\nabla_h P}+\nu\mathbf{D}_h\left(\vec{U}^{n}\right)+\mathbf{f}^{n+1} \tag{1}$$

where $\vec{U}^{*}$ represents the intermediate velocity field, which does not necessarily satisfy mass conservation, $\mathbf{A}_h$ represents the numerical discretization of the convection term, $\mathbf{D}_h$ represents the numerical discretization of the diffusion term, $\overline{\nabla_h P}$ represents the numerical discretization of the average pressure-gradient and $\mathbf{f}^{n+1}$ denotes the external body forces acting on the fluid. The intermediate velocity field is corrected, to satisfy mass conservation, by adjusting the pressure:

$$\frac{\vec{U}^{n+1}-\vec{U}^{*}}{\Delta t}=-\frac{1}{\rho}\nabla_h P^{*} \tag{2}$$

where the pressure correction P^{*} is obtained by taking the divergence of the mass conservation equation:

$$\frac{1}{\rho}\nabla_h^2 P^{*}=\frac{1}{\Delta t}\nabla_h\cdot\vec{U}^{*} \tag{3}$$

This is called the corrector step. In Eq. (3), $\nabla_h\cdot\vec{U}^{*}$ represents the numerical discretization of the divergence of $\vec{U}^{*}$.

In the immersed boundary method, the no-slip boundary condition on the immersed boundary is implemented by introducing momentum sources in the predictor step and mass sources in the corrector step. The momentum sources are basically body forces that the wall exerts on the flow, and are located inside the immersed boundary. They are calculated using interpolations with velocities outside this boundary, inside the actual fluid. After applying the body forces ($\mathbf{f}^{n+1}$ in Eq. (1)), the intermediate velocity field $\vec{U}^{*}$ will satisfy the no-slip condition at the immersed boundary. Section 2.1 summarizes the calculation of the body forces. To better maintain the no-slip condition after the adjustment of the velocity field in the corrector step, mass sources are introduced in and near the immersed boundary. These mass sources are briefly explained in Section 2.2. For the full explanation of both the momentum and mass sources, see the work by Kim et al. [13].

2.1. Immersed boundary method in the predictor step

This section describes the calculation of the body forces in the predictor step. The body forces change the velocity inside the wall such that the velocity at the wall (calculated using an interpolation) is equal to zero. Calculating the body force is, therefore, equivalent to calculating the velocity field inside the wall. The body forces are computed separately for the three velocity components u, v and w on the staggered grid; for simplicity, only the interpolation of a single velocity component, u, is considered here. The interpolations described in this section, used to calculate these body forces, are explained here for a 2D Cartesian geometry, but