Pressure boundary conditions for computing incompressible flows with SPH

S. Majid Hosseini[1] and James J. Feng[1,2*]

[1]*Department of Chemical and Biological Engineering, University of British Columbia Vancouver, BC V6T 1Z3, Canada*
[2]*Department of Mathematics, University of British Columbia, Vancouver, BC V6T 1Z2, Canada*

Abstract - In Smoothed Particle Hydrodynamics (SPH) methods for fluid flow, incompressibility may be imposed by a projection method with an artificial homogeneous Neumann boundary condition for the pressure Poisson equation. This is often inconsistent with physical conditions at solid walls and inflow and outflow boundaries. For this reason open-boundary flows have rarely been computed using SPH. In this work, we demonstrate that the artificial pressure boundary condition produces a numerical boundary layer that compromises the solution near boundaries. We resolve this problem by utilizing a "rotational pressure-correction scheme" with a consistent pressure boundary condition that relates the normal pressure gradient to the local vorticity. We show that this scheme computes the pressure and velocity accurately near open boundaries and solid objects, and extends the scope of SPH simulation beyond the usual periodic boundary conditions.

*Corresponding author. E-mail jfeng@chbe.ubc.ca

1   Introduction

Smoothed Particle Hydrodynamics (SPH) is a Lagrangian particle method that originated from astrophysical problems some three decades ago [1,2]. In recent years, it has evolved into a unique tool for computational fluid dynamics thanks to its meshless formalism and unique capability to handle large strains and morphological changes [3–5]. Nevertheless, SPH is still a relatively new tool in computational fluid dynamics, and is not as well developed as finite difference or finite element [6, 7]. Some fundamental issues, such as the treatment of incompressibility and boundary conditions, have not been fully resolved.

Traditionally, SPH methods employ a weakly compressible formalism with an artificial equation of state that specifies pressure as an algebraic function of density. A large speed of sound is used to maintain an acceptable density variation [8, 9, e.g.]. This weakly compressible SPH (WCSPH) algorithm is easy to program but has several disadvantages. The high speed of sound imposes a severely restrictive CFL criterion on the time step [10]. The pressure is subject to large and non-physical fluctuations, especially for relatively coarse spatial resolution. Such fluctuations may cause numerical instability. More important, they corrupt the result in problems where pressure is of physical interest, as in computing the drag on a solid object [11]. To bypass these difficulties, Cummins and Rudman [10] introduced truly incompressible SPH (ISPH) algorithms based on the *projection scheme.* Numerical results have demonstrated that ISPH largely eliminates the density and pressure fluctuations, produces more accurate predictions of velocity and forces on solids, and is in general more efficient than WCSPH [10–12].

However, these ISPH algorithms suffer from an inconsistency in the pressure boundary condition. In the pressure-correction projection scheme, pressure is computed from a Poisson equation, and is then used to correct the velocity field to make it divergence-free [13,14]. Solving the Poisson equation requires boundary conditions for the pressure, which are not required by the original Navier-Stokes equation. Following the pioneers of the original projection scheme [13, 14], Cummins and Rudman [10] and Lee *et al.* [11] both employed a homogeneous Neumann condition:

$$\boldsymbol{n} \cdot \nabla p|_\Gamma = 0, \tag{1}$$

where $\boldsymbol{n}$ is the normal vector to the boundary Γ. In many flow situations, of course, the normal pressure gradient does not vanish on the boundary. Important examples are open-boundary problems where the fluid enters and exits the computational domain driven by a pressure gradient, and flow around bluff bodies or through channels with variable cross sections. In such cases, the simple homogeneous condition of Eq. (1) produces a numerical boundary layer at the boundary and corrupts the accuracy of the solution. Partly for this reason, open-boundary flows are rarely computed using SPH. The standard practice is to impose periodic conditions on the inlet and the exit, and replace the pressure gradient driving the flow by a body force [11]. In fact, Cummins and Rudman [10] and Lee *et al.* [11] both concluded their paper by highlighting the need to develop a more suitable pressure boundary condition.

This is a well-known problem in the projection scheme, and is by no means specific to SPH. A physically consistent boundary condition on pressure has long been sought after [15]. Recently Guermond *et al.* [16] published a careful examination of the various projection methods, and recommended the "rotational pressure-correction scheme" of Timmermans *et al.* [17], with a nonhomogeneous pressure condition for the Poisson equation. Their finite-element numerical tests show that this largely removes the artificial boundary layer caused by Eq. (1) and preserves the accuracy of the solution throughout the domain.

Our work is inspired on the one hand by the need for a suitable pressure boundary condition in ISPH [10, 11], and on the other by the successful finite-element computations using the rotational scheme [16]. We implement in SPH the rotational pressure-correction scheme, with its nonhomogeneous pressure boundary condition. The new ISPH algorithm is validated by simulating pressure-driven Poiseuille flow with open boundaries and flow past a square cylinder. In each case, the SPH result is in very good agreement with analytical or finite-element benchmarks. Thus, the new pressure boundary condition enhances the capability of SPH in computing incompressible flows; previously inaccessible problems with open and solid boundaries can now be computed routinely.

2 Projection schemes

2.1 Homogeneous projection scheme

The classical projection scheme of Chorin [13] and Temam [14] features the homogeneous
pressure boundary condition (Eq. 1), and will be called hereafter the *homogeneous projec-
tion scheme*. Consider the flow of an incompressible fluid governed by the Navier-Stokes
equations:

$$\nabla \cdot \boldsymbol{u} \;=\; 0, \tag{2}$$

$$\frac{d\boldsymbol{u}}{dt} + \frac{1}{\rho}\nabla p - \nu\nabla^2\boldsymbol{u} \;=\; \boldsymbol{g}, \tag{3}$$

where $\boldsymbol{g}$ is a body force. For the time being, let us assume a Dirichlet boundary condition
on the velocity:

$$\boldsymbol{u}|_\Gamma = \boldsymbol{u}_\Gamma. \tag{4}$$

Neumann conditions on $\boldsymbol{n} \cdot \nabla\boldsymbol{u}$, as appropriate for the exit in open-boundary flows, will be
considered in subsection 2.3.

The projection scheme is a prediction-correction technique. In the prediction step, we
compute an intermediate velocity field $\tilde{\boldsymbol{u}}$ from the momentum equation (Eq. 3) without the
pressure term. This velocity satisfies the boundary condition (Eq. 4), but in general not
the continuity equation (Eq. 2). In the correction step, we compute a pressure field from
the intermediate velocity, and then use the pressure gradient to correct $\tilde{\boldsymbol{u}}$ into a solenoidal
velocity $\boldsymbol{u}$. Using second-order time stepping, the scheme can be written as [16]:

$$\frac{1}{2\Delta t}(3\tilde{\boldsymbol{u}}^{k+1} - 4\boldsymbol{u}^k + \boldsymbol{u}^{k-1}) - \nu\nabla^2\boldsymbol{u}^k = \boldsymbol{g}, \quad \tilde{\boldsymbol{u}}^{k+1}|_\Gamma = \boldsymbol{u}_\Gamma, \tag{5}$$

$$\begin{cases} \dfrac{1}{2\Delta t}(3\boldsymbol{u}^{k+1} - 3\tilde{\boldsymbol{u}}^{k+1}) + \dfrac{1}{\rho}\nabla p^{k+1} = 0, \\[2mm] \nabla \cdot \boldsymbol{u}^{k+1} = 0, \quad \boldsymbol{n} \cdot \boldsymbol{u}^{k+1}|_\Gamma = \boldsymbol{n} \cdot \boldsymbol{u}_\Gamma, \end{cases} \tag{6}$$

where k and $k+1$ indicate the old and new time steps, respectively. In the SPH implemen-
tation, the temporal differencing gives the material derivative and thus no advection term
appears. For our purpose here, the most interesting aspect is that p^{k+1} is computed from
a Poisson equation by taking the divergence of the first equation of Eq. (6):

$$\nabla^2 p^{k+1} = \frac{3\rho}{2\Delta t}\nabla \cdot \tilde{\boldsymbol{u}}^{k+1}. \tag{7}$$

Equation (6) implies an homogeneous Neumann boundary condition for p^{k+1}:

$$\boldsymbol{n} \cdot \nabla p^{k+1}|_\Gamma = 0, \tag{8}$$

which is evidently inconsistent with the momentum equation (Eq. 3). Unless the physical problem is such that the normal pressure gradient indeed vanishes at the boundary, this homogeneous Neumann condition produces a numerical boundary layer in the solution and corrupts its accuracy [16].

2.2 Rotational projection scheme

In the prediction step, one may retain a pressure gradient based on the prior time step when computing the intermediate velocity $\tilde{\boldsymbol{u}}^{k+1}$ from the momentum equation. Then what appears in the correction step will be a pressure difference between the current and prior steps. The essence of the rotational scheme is to modify this pressure difference so as to make the pressure Neumann condition consistent with the momentum equation [17]. Following Guermond *et al.* [16], we write the scheme in this form:

$$\frac{1}{2\Delta t}(3\tilde{\boldsymbol{u}}^{k+1} - 4\boldsymbol{u}^k + \boldsymbol{u}^{k-1}) - \nu\nabla^2\tilde{\boldsymbol{u}}^{k+1} + \frac{1}{\rho}\nabla p^k = \boldsymbol{g}, \quad \tilde{\boldsymbol{u}}^{k+1}|_\Gamma = \boldsymbol{u}_\Gamma, \tag{9}$$

$$\begin{cases} \dfrac{1}{2\Delta t}(3\boldsymbol{u}^{k+1} - 3\tilde{\boldsymbol{u}}^{k+1}) + \dfrac{1}{\rho}\nabla\phi^{k+1} = 0, \\ \nabla \cdot \boldsymbol{u}^{k+1} = 0, \quad \boldsymbol{n} \cdot \boldsymbol{u}^{k+1}|_\Gamma = \boldsymbol{n} \cdot \boldsymbol{u}_\Gamma, \end{cases} \tag{10}$$

where ϕ^{k+1} is the modified pressure defined as:

$$\phi^{k+1} = p^{k+1} - p^k + \mu\nabla \cdot \tilde{\boldsymbol{u}}^{k+1}, \tag{11}$$

$\mu = \rho\nu$ being the dynamic viscosity. The following observations can be made of this projection scheme.

1. Retaining the old pressure gradient ∇p^k in the prediction step increases the accuracy of the scheme [18].

2. Substituting Eq. (11) into Eq. (10) and then adding it to Eq. (9), we arrive at the proper discretization of the momentum equation:

$$\frac{1}{2\Delta t}(3\boldsymbol{u}^{k+1} - 4\boldsymbol{u}^k + \boldsymbol{u}^{k-1}) - \nu\nabla^2\boldsymbol{u}^{k+1} + \frac{1}{\rho}\nabla p^{k+1} = \boldsymbol{g}, \tag{12}$$

where we have used the identity $\nabla(\nabla \cdot \tilde{\boldsymbol{u}}^{k+1}) - \nabla^2\tilde{\boldsymbol{u}}^{k+1} = \nabla \times \nabla \times \tilde{\boldsymbol{u}}^{k+1}$ and the fact that $\nabla \times \nabla \times \tilde{\boldsymbol{u}}^{k+1} = \nabla \times \nabla \times \boldsymbol{u}^{k+1} = -\nabla^2\boldsymbol{u}^{k+1}$ since $\nabla \times \tilde{\boldsymbol{u}}^{k+1} = \nabla \times \boldsymbol{u}^{k+1}$ and $\nabla \cdot \boldsymbol{u}^{k+1} = 0$ from Eq. (10).

3. The use of ϕ^{k+1} for correcting the velocity, instead of p^{k+1}, is the key difference from the homogeneous scheme. ϕ^{k+1} can be solved from a Poisson equation with its own homogeneous Neumann condition implied by Eq. (10):

$$\begin{cases} \nabla^2 \phi^{k+1} = \dfrac{3\rho}{2\Delta t} \nabla \cdot \tilde{\boldsymbol{u}}^{k+1}, \\[2mm] \boldsymbol{n} \cdot \nabla \phi^{k+1}|_\Gamma = 0. \end{cases} \tag{13}$$

Translated into pressure, these become

$$\begin{cases} \nabla^2 p^{k+1} = \nabla^2 p^k - \mu \nabla^2 (\nabla \cdot \tilde{\boldsymbol{u}}^{k+1}) + \dfrac{3\rho}{2\Delta t} \nabla \cdot \tilde{\boldsymbol{u}}^{k+1}, \\[2mm] \boldsymbol{n} \cdot \nabla p^{k+1}|_\Gamma = \boldsymbol{n} \cdot \nabla (p^k - \mu \nabla \cdot \tilde{\boldsymbol{u}}^{k+1})|_\Gamma. \end{cases} \tag{14}$$

Using Eq. (9) and the identities following Eq. (12), the nonhomogeneous Neumann boundary condition for p^{k+1} can be rewritten as

$$\boldsymbol{n} \cdot \nabla p^{k+1}|_\Gamma = \boldsymbol{n} \cdot (\rho \boldsymbol{g} + \mu \nabla^2 \boldsymbol{u}^{k+1})|_\Gamma, \tag{15}$$

which is *consistent* with the momentum equation (Eq. 12). It was called the "rotational" pressure boundary condition because the viscous stress term $\mu \nabla^2 \boldsymbol{u}^{k+1} = -\mu \nabla \times \nabla \times \boldsymbol{u}^{k+1}$ was viewed as the curl of the vorticity vector [16].

2.3   Natural boundary condition

So far, our discussion is limited to Dirichlet boundary conditions for the velocity. With open-boundary flows, it is more appropriate to impose a *natural boundary condition* at the exit of the flow domain. Denoting by Γ_1 the portion of the boundary with Dirichlet velocity condition and Γ_2 the exit, the boundary conditions are

$$\boldsymbol{u}|_{\Gamma_1} = \boldsymbol{u}_\Gamma, \tag{16}$$

$$\left[-p\boldsymbol{n} + \mu \boldsymbol{n} \cdot (\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^{\mathrm{T}}) \right]|_{\Gamma_2} = -p_a \boldsymbol{n}, \tag{17}$$

where p_a is the ambient pressure. Following Guermond *et al.* [16], we modify the rotational scheme as follows:

$$\begin{cases} \dfrac{1}{2\Delta t}(3\tilde{\boldsymbol{u}}^{k+1} - 4\boldsymbol{u}^k + \boldsymbol{u}^{k-1}) - \nu \nabla^2 \tilde{\boldsymbol{u}}^{k+1} + \dfrac{1}{\rho}\nabla p^k = \boldsymbol{g} \\[2mm] \tilde{\boldsymbol{u}}^{k+1}|_{\Gamma_1} = \boldsymbol{u}_\Gamma, \quad [-p^k \boldsymbol{n} + \mu \boldsymbol{n} \cdot (\nabla \tilde{\boldsymbol{u}}^{k+1} + (\nabla \tilde{\boldsymbol{u}}^{k+1})^{\mathrm{T}})]|_{\Gamma_2} = -p_a \boldsymbol{n}, \end{cases} \tag{18}$$

$$\begin{cases} \dfrac{1}{2\Delta t}(3\boldsymbol{u}^{k+1} - 3\tilde{\boldsymbol{u}}^{k+1}) + \dfrac{1}{\rho}\nabla \phi^{k+1} = 0, \\[2mm] \nabla \cdot \boldsymbol{u}^{k+1} = 0, \quad \boldsymbol{n} \cdot \boldsymbol{u}^{k+1}|_{\Gamma_1} = \boldsymbol{n} \cdot \boldsymbol{u}_\Gamma, \quad \phi^{k+1}|_{\Gamma_2} = 0 \end{cases} \tag{19}$$

$$\phi^{k+1} = p^{k+1} - p^k + \mu \nabla \cdot \tilde{\boldsymbol{u}}^{k+1}. \tag{20}$$

Note that Eq. (19) implies that on Γ_1 the homogeneous Neumann condition is imposed on ϕ^{k+1} as before, but on Γ_2 a Dirichlet condition is used. Equivalently, p^{k+1} satisfies the non-homogeneous Neumann condition Eq. (15) on Γ_1 and a non-homogeneous Dirichlet condition on Γ_2. The latter removes the indeterminacy in the pressure-Poisson equation. The homogeneous scheme (Eqs. 5, 6) can be similarly modified, with a Dirichlet pressure condition at the exit: $p^{k+1}|_{\Gamma_2} = 0$.

3 SPH implementation

To some degree, SPH particles may be viewed as representing blobs of material. But they are for the most part a numerical device for discretizing the Navier-Stokes equations in Lagrangian coordinates. The particles are interpolation points from which the fluid properties may be calculated by smoothing over neighboring particles. Although the particles have mass and move according to Newton's law of motion, the forcing terms stem directly from discretizing the continuum governing equations. Solid boundaries are discretized by solid particles, and the fluid-solid interaction is defined so as to satisfy the no-slip boundary condition.

3.1 SPH interpolation

The SPH method allows any property to be interpolated from its values at a set of discrete points—the SPH particles—using a kernel or weighting function $W(\boldsymbol{r}-\boldsymbol{r}', h)$, which specifies the contribution to any field variable at position $\boldsymbol{r}$ by a particle at $\boldsymbol{r}'$ that lies within the compact support of the kernel function. The range of the compact support, indicated by the length scale h, determines the maximum interaction length between particles. The weighting function is normalized such that

$$\int_V W(\boldsymbol{r} - \boldsymbol{r}', h)d\boldsymbol{r}' = 1, \qquad \lim_{h \to 0} W(\boldsymbol{r} - \boldsymbol{r}', h) = \delta(\boldsymbol{r} - \boldsymbol{r}'), \tag{21}$$

V being the entire space. If a field variable $A(\boldsymbol{r})$ is known only at a discrete set of particles $\boldsymbol{r}'_j$, then its value at any spatial location $\boldsymbol{r}$ can be approximated by:

$$\begin{aligned} \langle A_h\left(\boldsymbol{r}\right)\rangle &= \int_V A(\boldsymbol{r}')\, W(\boldsymbol{r} - \boldsymbol{r}', h)\, d\boldsymbol{r}' \\ &= \sum_j A(\boldsymbol{r}'_j)W(\boldsymbol{r} - \boldsymbol{r}'_j, h)V_j = \sum_j \frac{m_j}{\rho_j}A(\boldsymbol{r}'_j)W(\boldsymbol{r} - \boldsymbol{r}'_j, h), \end{aligned} \tag{22}$$

where V_j is the volume element at $\boldsymbol{r}'_j$, and has been replaced by the ratio between the mass and density of the jth particle: $V_j = m_j/\rho_j$. The summation is over all particles that lie within a circle or sphere with the radius of compact support of the kernel function. In our ISPH formalism, m_j and ρ_j are constant for all particles. The subscript j is retained to be general and consistent with convention. The same is true for the representation of viscosity μ in formulae below.

We have tested the cubic and quintic spline kernels [9] as well as the truncated Gaussian kernel [19]. The Gaussian kernel provides a good compromise between computational cost and accuracy, and has been used in all the results to be presented. In terms of the distance $s = |\boldsymbol{r} - \boldsymbol{r}'|/h$, it can be written as

$$W(s) = \begin{cases} \frac{e^{-s^2} - e^{-9}}{\pi h^2 (1 - 10 e^{-9})} & 0 \le s \le 3 \\ 0 & s > 3. \end{cases} \tag{23}$$

We have used $h = 1.2 L_0$, L_0 being the initial spacing between neighboring particles in a regular square lattice.

In classical SPH methods, the gradient approximations is based on convolving the variables with the kernel function [7]:

$$\langle \nabla A_h(\boldsymbol{r}) \rangle = \int_V A(\boldsymbol{r}') \nabla W(\boldsymbol{r} - \boldsymbol{r}', h) \, dr' = \sum_j \frac{m_j}{\rho_j} A(\boldsymbol{r}'_j) \nabla W(\boldsymbol{r} - \boldsymbol{r}'_j, h). \tag{24}$$

The interpolation among neighboring particles is equivalent to a Taylor expansion that theoretically has second-order accuracy [20]. However, as the particles move and their spatial distribution becomes irregular, the accuracy in approximating the gradient vector can be much compromised. To improve the accuracy of the kernel-based approximation, a number of corrective procedures have been proposed [20, 21, e.g.]. In our simulations, the normalization technique proposed by Oger *et al.* [20] works well. They constructed a correction matrix that accounts for irregular distribution of neighbors around each particle, which in two dimensions is written as

$$\boldsymbol{L}(\boldsymbol{r}) = \begin{bmatrix} \sum_j V_j (x_j - x) \frac{\partial W_{ij}}{\partial x} & \sum_j V_j (x_j - x) \frac{\partial W_{ij}}{\partial y} \\ \sum_j V_j (y_j - y) \frac{\partial W_{ij}}{\partial x} & \sum_j V_j (y_j - y) \frac{\partial W_{ij}}{\partial y} \end{bmatrix}^{-1}, \tag{25}$$

with $V_j = m_j/\rho_j$ and W_{ij} is a shorthand for $W(\boldsymbol{r}_i - \boldsymbol{r}_j, h)$. Multiplying $\boldsymbol{L}(\boldsymbol{r})$ into the gradient of the kernel produces high accuracy even with highly irregular particle distributions [20]. For example, the pressure gradient and velocity divergence are computed as

$$\nabla p_i = \sum_j \frac{m_j}{\rho_j}(p_j - p_i)\boldsymbol{L}(\boldsymbol{r}_i) \cdot \nabla_i W_{ij}, \tag{26}$$

$$\nabla \cdot \boldsymbol{u}_i = \frac{1}{\rho_i}\sum_j m_j(\boldsymbol{u}_j - \boldsymbol{u}_i) \cdot \boldsymbol{L}(\boldsymbol{r}_i) \cdot \nabla_i W_{ij}, \tag{27}$$

where $p_i = p(\boldsymbol{r}_i)$, $\boldsymbol{u}_i = \boldsymbol{u}(\boldsymbol{r}_i)$, and ∇_i indicates differentiation with respect to $\boldsymbol{r}_i$. Note that the right-hand-sides of Eqs. (26) and (27) use the "symmetric" form of the interpolation formula for the gradient [7], an oft-used alternative to Eq. (24). In simulating the flow around a square cylinder, in particular, the particle distribution becomes highly disordered at the front and back of the obstacle. The classic formulae would produce large pressure fluctuations, which are mostly suppressed by the corrected forms of the gradient and divergence formulae.

3.2 Solution algorithm

For both the homogeneous and rotational schemes, the algorithm reflects the same two-step structure. We will outline the solution procedure by referring to the equations of the rotational scheme, and a similar one is followed for the homogeneous scheme. The prediction step consists in solving a Helmholtz equation for the intermediate velocity $\tilde{\boldsymbol{u}}^{k+1}$ (Eq. 9 or 18), while the correction step a Poisson equation for the modified pressure ϕ^{k+1} (Eq. 13 or its counterpart for natural boundary conditions at the exit). In a time-dependent setting, we have tested implicit and explicit schemes in both steps. Balancing stability and efficiency, we have settled on a scheme similar to that of Lee *et al.* [11] of solving for the intermediate velocity explicitly in the prediction step and for the modified pressure implicitly in the correction step. Of course, in the end we correct the velocity to make it divergence-free via Eq. (10) or Eq. (19).

The Laplacian of velocity and modified pressure are discretized in SPH using the following formulae [9, 22]:

$$\nu\nabla^2\boldsymbol{u}_i = \sum_j \frac{m_j(\mu_i + \mu_j)\boldsymbol{r}_{ij} \cdot \boldsymbol{L}(\boldsymbol{r}_i) \cdot \nabla_i W_{ij}}{\rho_i \rho_j r_{ij}^2}(\boldsymbol{u}_i - \boldsymbol{u}_j), \tag{28}$$

$$\nabla^2\phi_i = \sum_j \frac{2m_j}{\rho_j}\frac{\boldsymbol{r}_{ij} \cdot \boldsymbol{L}(\boldsymbol{r}_i) \cdot \nabla_i W_{ij}}{r_{ij}^2}(\phi_i - \phi_j), \tag{29}$$
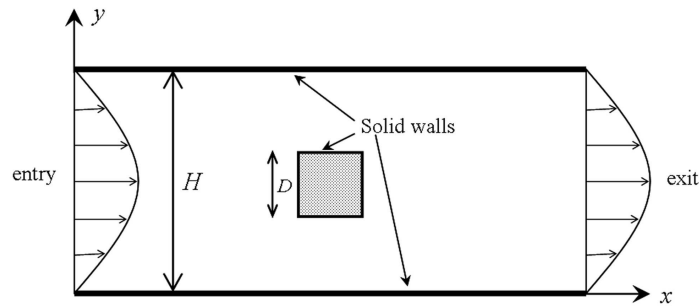
Figure 1: Computational domain for flow past a square cylinder of side D in a channel of width H, showing the three types of boundaries to be encountered in the simulation.

where $\boldsymbol{r}_{ij} = \boldsymbol{r}_i - \boldsymbol{r}_j$ and $r_{ij} = |\boldsymbol{r}_{ij}|$. In solving the pressure-Poisson equation, we combine Eq. (29) for all the particles to construct a linear system of equations for ϕ^{k+1}. If all boundary segments of the domain have Dirichlet boundary condition on $\boldsymbol{u}$ and Neumann boundary condition on ϕ, there will be an indeterminacy in ϕ and the resulting linear system will be degenerate. This is to be expected since in the incompressible Navier-Stokes system, the absolute value of pressure is immaterial. The indeterminacy is removed by replacing one of the diagonal elements of the coefficient matrix by a very large number, thereby setting that ϕ value to nil [23]. The linear system is solved by using the stabilized version of the bi-conjugate gradient method (Bi-CGSTAB).

3.3 Boundary conditions

Because of the Lagrangian nature of SPH particles, implementing boundary conditions at fixed spatial boundaries is less straightforward than in mesh-based methods. To be specific, we discuss three types of boundaries that appear in our simulations: solid walls, inflow boundary (entry) and outflow boundary (exit). These come from the geometry of flow around an obstacle in a channel (Fig. 1), and are common to a broad class of flow simulations.

3.3.1 Solid walls

For velocity, the *no-slip boundary condition* on solid walls has been implemented in earlier work with image particles [10, 24], dummy particles with zero velocity [11, 25], dummy

particles with extrapolated velocity [9], and a normalization procedure to compensate for the boundary deficiency of neighboring particles [26]. Here we adopt the method of Morris *et al.* [9] of extrapolating the fluid velocity onto layers of dummy particles inside the solid wall. Initially we deploy a square lattice of SPH particles with spacing L_0 to cover the fluid domain as well as the solid surfaces. Furthermore, three layers of dummy particles are drawn inside the solid, also at spacing L_0. Our Gaussian kernel has a cut-off range of $3h$ (Eq. 23) and we set $h = 1.2L_0$. Thus three layers of dummy particles, in addition to the wall particles, cover the compact support of all fluid particles. At the beginning of the prediction step, we assign a velocity to each dummy particle that is the negative of the interpolated velocity of its image inside the fluid domain. These dummy particles will serve as neighbors in solving the momentum equation for the fluid particles, but they will not move from one time step to the next. The velocity of the wall particles remains zero, of course. This setup is used for the channel walls as well as the solid obstacle in the middle of the domain. For dummy particles along the diagonal inside the corners of the square cylinder (cf. Fig. 1), we follow the scheme of Lee *et al.* [11] by averaging between their neighbors. The extrapolation of velocity into the solid wall is essential for accurate evaluation of wall shear stresses and the overall drag [27].

On solid walls, the *homogeneous Neumann condition* for p (Eq. 8) or ϕ (Eq. 13) applies. Following Lee *et al.* [11], we simply propagate the p or ϕ value of the wall particle to the dummy particles along the normal direction. This is implemented numerically by manipulating the relevant entries in the linear system.

3.3.2 Entry

Most SPH simulations up-to-date have replaced entry and exit conditions by periodic conditions [9,11]. These are rather easy to implement; particles exiting the computational domain are simply reinserted at the entry, carrying the same properties. Lastiwka *et al.* [28] have implemented a genuine entry condition, for compressible flows, by attaching an inflow zone upstream of the computational domain. Properties of particles inside the inflow zone are assigned to reflect the analytical boundary conditions to be realized at the entry.

For our incompressible flow, the proper entry condition is a prescribed unidirectional velocity profile $(u_{in}(y), 0)$, say a parabolic profile representing fully developed Poiseuille flow.

Our approach is similar in spirit to Lastiwka *et al.* [28]. We replicate the first line of fluid particles at the entry, called the *entry layer*, three times upstream at spacing L_0. The entry layer and the three layers of image particles are all assigned the prescribed velocity profile u_{in}. Thus, the fluid particles immediately downstream of the entry layer are preceded by four particles upstream, all bearing the prescribed velocity u_{in}, and the boundary deficiency problem is solved. The Neumann condition $\boldsymbol{n} \cdot \nabla \phi = 0$ (or $\boldsymbol{n} \cdot \nabla p = 0$ for the homogeneous scheme) is easily implemented by assigning the ϕ (or p) value of an entry-layer particle onto its images upstream.

When a particle of the entry layer moves beyond L_0 downstream of the inlet, it becomes an interior particle and its upstream image particle becomes part of the entry layer. Meanwhile, another image particle is injected far upstream to be the third image particle.

3.3.3 Exit

Under certain circumstances, it is possible to treat the exit the same way as the entry, with a prescribed velocity profile. For instance, this is reasonable for Poiseuille flow in a straight channel and for more complex channel flows with an exit far downstream of flow disturbances. We have tested these cases in numerical experiments. However, the natural boundary condition (Eq. 17) is more appropriate in most cases.

We will deal with the simple case of equilibrated pressures at the exit (i.e., no suction or blowing), so that the normal stress balance of Eq. (17) reduces to a simple Neumann condition for the velocity $\boldsymbol{n} \cdot \nabla \boldsymbol{u} = 0$. To compensate for the boundary deficiency at the exit, we again copy the last layer of fluid particles (the *exit layer*) onto three layers of image particles downstream of the exit with spacing L_0, each bearing the same velocity as their progenitor in the exit layer. Now the velocity of the near-exit fluid particles can be updated as for the inner particles, and the viscous stress and velocity divergence can be evaluated in the standard way. For pressure, we impose the Dirichlet conditions $p = 0$ and $\phi = 0$ at the exit for the homogeneous and rotational schemes, respectively. To implement this, the image particles downstream of the exit take on p or ϕ values extrapolated from the upstream particles, which are incorporated into the discretization of the Poisson equation by properly modifying the matrix.

When a particle of the exit layer crosses the outlet boundary, it ceases to be a fluid particle and becomes an image particle, bearing the same velocity. Meanwhile the upstream fluid particle becomes part of the exit layer. The p or ϕ value of the new image particle is again extrapolated from the upstream fluid particles with the condition that $p = 0$ or $\phi = 0$ at the exit.

Finally, we mention a boundary-deficiency issue in computing $\nabla \cdot \tilde{\boldsymbol{u}}$ for the rotational scheme, which appears in the Poisson equation for ϕ and at the end of the correction step when extracting the physical pressure p from ϕ (Eq. 11). Computing $\nabla \cdot \tilde{\boldsymbol{u}}$ near the boundaries requires $\tilde{\boldsymbol{u}}$ for the dummy or image particles outside the domain. At the entry and exit, we assign the $\tilde{\boldsymbol{u}}$ value of the entry or exit layer to the image particles as described above, and thus $\nabla \cdot \tilde{\boldsymbol{u}}$ can be computed. Near solid boundaries, however, using the linearly extrapolated velocity for dummy particles, as is done in implementing the no-slip boundary condition, would produce large spurious pressure on the dummy particles and corrupt the solution near the solid. This is because the pressure correction scheme in Eq. (11) is designed for fluid particles and does not account for the peculiar property of the dummy particles in ostensibly carrying a velocity but not really moving. Instead, it is more reasonable to assign $\tilde{\boldsymbol{u}} = 0$ on the dummy particles inside solid walls in computing $\nabla \cdot \tilde{\boldsymbol{u}}$. Numerical experiments show that this more accurately predicts the pressure field near solid walls. Interestingly, the homogeneous scheme is very insensitive to $\tilde{\boldsymbol{u}}$ values on the dummy particles. The pressure field differs only by some 2% between the two choices. This may be because in the homogeneous scheme, pressure p^{k+1} is computed directly from the Poisson equation with no further correction by $\nabla \cdot \tilde{\boldsymbol{u}}$.

3.4   Tensile instability

The SPH representation of a fluid of constant and uniform density relies on a more or less uniform distribution of the particles. Because of how the particles interact via the kernel function, however, clustering of particles is liable to arise, especially when the material is subject to tensile deformation. This gives rise to the well-documented tensile instability [6]. An example from our own computations is illustrated in Fig. 2(a) for flow around a square cylinder to be discussed in detail in section 4.3.
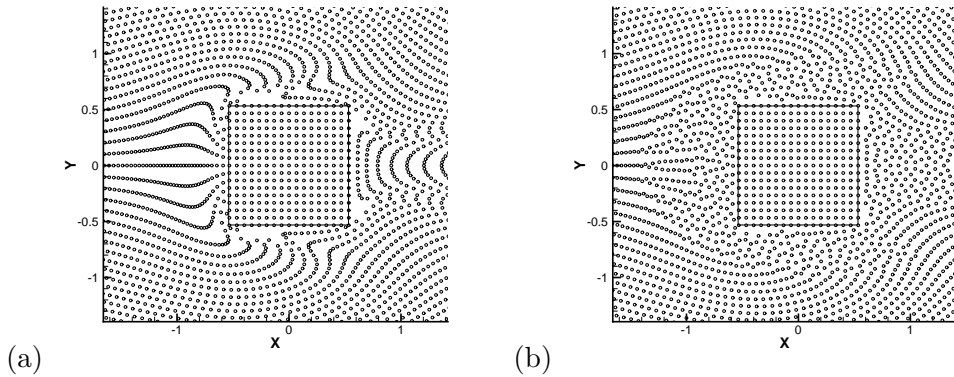
13

Figure 2: (a) Tensile instability caused by the particles forming anisotropic strings in flow around a square cylinder. (b) The particle shifting approach [22] largely removes the undesirable clustering and maintains a more or less uniform particle distribution.

Among the various remedies proposed in the literature [6,21,22,29,30], we have found the *particle shifting strategy* of Xu *et al.* [22] most effective and efficient for our ISPH projection scheme. Originally proposed by Nestor *et al.* [30] for a Finite Volume Particle Method, the idea consists in shifting particles slightly away from their streamlines and correcting their velocity and pressure according to a first-order interpolation. The direction and amount of shifting are determined from the arrangement of nearby particles, and we have adopted the formula of Xu *et al.* [22]. Figure 2(b) shows that this scheme effectively suppresses the formation of particle strings and averts the onset of tensile instability.

4 Numerical results

4.1 Poiseuille flow driven by body force in periodic domain

As a baseline, we compute the transient development of a Poiseuille flow in a straight channel driven by a body force. The SPH simulation is set up in the usual way as in the literature [9], with the pressure gradient set to zero, and periodic boundary conditions imposed on both ends of the domain. Thus, $\boldsymbol{n} \cdot \nabla p$ does vanish on all boundary segments, and the homogeneous and rotational projection schemes become identical for this simple problem. We have two objectives. The first is to validate our code for both projection methods against this benchmark problem, which has an analytical solution [9]. The second is
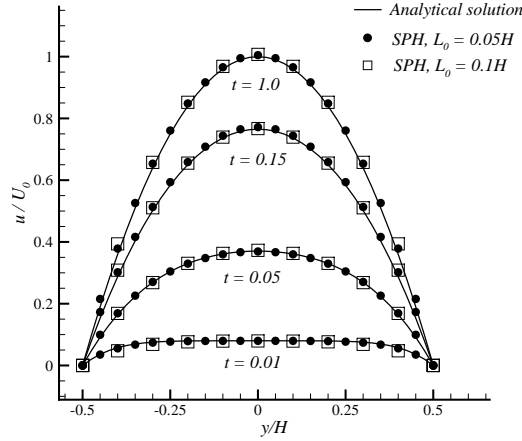
Figure 3: Development of the Poiseuille velocity profile in time: comparison between SPH simulation and the analytical solution. Two numerical solutions are shown, corresponding to initial particle spacing $L_0 = 0.05H$ and $0.1H$. Time is made dimensionless by H/U_0.

to examine the accuracy of the solution with increasing numbers of particles, and determine the necessary level of spatial resolution.

The channel has a width H and a length $2H$. The length may seem small, but has no effect on the solution as no streamwise variation is detected in the solution. Initially the particles are arranged in a uniform square pattern of spacing L_0, with zero initial velocity. At $t = 0$, the particles start to move under a constant body force g. The temporal development of the velocity profiles is recorded and compared with the analytical solution. The final steady-state centerline velocity $U_0 = gH^2/(8\nu)$ corresponds to a Reynolds number $Re = U_0H/\nu = 1$.

Figure 3 compares the computed velocity profiles at four times with the analytical solution, the last one $t = 1$ approaching steady state. In this unidirectional flow, the particles on the same streamline move with identical velocity, and each data point in the plot corresponds to one such row of particles. Length, velocity and time are made dimensionless by H, U_0 and H/U_0, respectively. Numerical solutions at two spatial resolutions are shown; initial particle spacing $L_0 = 0.05H$ and $L_0 = 0.1H$ correspond to a total of 1120 and 360 particles. Note first the excellent agreement between the SPH solutions and the analytical solution. This serves as a validation of our methodology and code. Second, the numerical
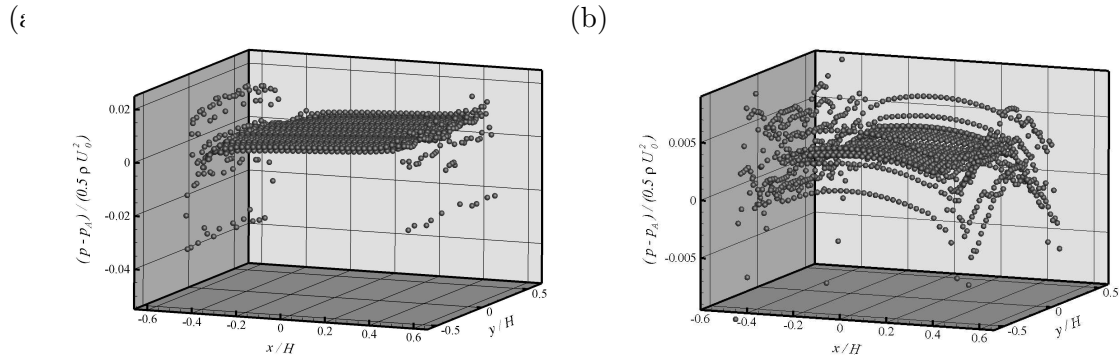
15

Figure 4: Deviation of the computed pressure field $p(\boldsymbol{r})$ from the analytical solution $p_A(\boldsymbol{r})$ (i.e., a linear distribution along the flow direction). (a) Homogeneous projection scheme. (b) Rotational projection scheme. The pressure error has been scaled by $\frac{1}{2}\rho U_0^2$ and zeroed at the center of the domain.

solutions at both spatial resolutions are practically indistinguishable; this suggests that $L_0 = 0.1H$ is sufficient for such a simple problem. Finer resolutions are required for more complex flow geometries, as will be shown shortly.

4.2 Fully developed Poiseuille flow with open boundaries

To investigate the effect of the pressure Neumann boundary condition in the homogenous and rotational schemes, we simulate the steady-state Poiseuille flow in a straight channel with *entry* and *exit boundary conditions*. The channel width and length are both H. The velocity and pressure boundary conditions are imposed as described in section 3.3. Specifically, the velocity assumes a parabolic profile at the inlet:

$$u_{in}(y) = \frac{U_0}{H^2}(H^2 - 4y^2), \tag{30}$$

and vanishes on the side walls. The Reynolds number $Re = U_0 H/\nu = 1$. On both the inlet and side walls, p or ϕ, in the two projection schemes respectively, satisfies the homogeneous Neumann condition. At the exit, the velocity has zero normal gradient, and the Dirichlet conditions $p = 0$ and $\phi = 0$ are imposed for the homogeneous and rotational schemes. For this calculation we used a total of 2000 particles with $L_0 = 0.025H$, which provides sufficient resolution.

Once the initial transient dies out, both schemes accurately reproduce the parabolic velocity profile throughout the domain, and their difference lies in the pressure field. To

16

illustrate the pressure boundary layers clearly, we subtract the analytical linear pressure distribution from the computed pressure field to produce a pressure error, with its value zeroed at the center of the domain for a clearer view. Figure 4 depicts the two-dimensional distribution of such pressure errors for the two schemes using a snapshot of the SPH particles. As expected, the homogeneous scheme produces a prominent boundary layer at the entry (Fig. 4a). However, there is also appreciable error at the exit. Unlike at the entry, the $p = 0$ condition here and the linear extrapolation of p for downstream image particles are consistent with the expected linear pressure profile. Thus the exit error arises from a different source than the entry boundary layer; it is due to errors in the velocity since a weaker Neumann condition $\boldsymbol{n} \cdot \nabla \boldsymbol{u} = 0$ is imposed at the exit. In Fig. 4(b), the rotational scheme effectively eliminates the entry boundary layer, and reduces the maximum error by an order of magnitude. Note that even in this case, the entry and exit tend to produce larger errors due to extrapolation for the image particles outside the domain. Given the analytical pressure gradient in a Poiseuille flow $dp/dx = 8\mu U_0/H^2$ and the domain length of H, the scaled pressure drop between the entry and exit would be $16/Re = 16$. Relative to this, the maximum pressure error in Fig. 4(a) for the homogeneous scheme amounts to 0.3%, which is quite acceptable. The rotational scheme has a maximum error of about 0.04%.

The results have clearly demonstrated the pressure boundary layer suffered by the homogeneous projection scheme and the capacity of the rotational scheme to eliminate it. But for this simple problem, the ill effect is small and limited to a small area of the domain, and does not seriously corrupt the quality of the solution on the whole. This is no longer the case for flow around solid objects, where the incorrect pressure boundary condition seriously compromises quantities of physical interest such as the drag force.

4.3 Channel flow past a square cylinder

The last problem of our numerical experimentation is the flow around a square cylinder in a two-dimensional channel (Fig. 1). As a test of the rotational projection scheme, this problem is attractive for several reasons. First, flow around obstacles is known to cause difficulties for WCSPH schemes [11]. The artificial pressure is liable to large spurious oscillations near the solid surface, thus hampering an accurate computation of the drag. Such

problems are where the ISPH, with the rotational pressure boundary condition, is poten-
tially most advantageous and useful. Second, this is an oft-used benchmark problem in the
literature for testing the capability of various numerical methods to accurately calculate
the pressure and velocity fields and the drag force [31, e.g.]. For the geometry and param-
eters of our SPH simulation, we have carried out a high-resolution finite-element solution
as a benchmark. Finally and most interestingly, Lee *et al.* [11] have computed a similar
problem using ISPH. In their setup, the fluid flow is driven by a body force in a channel
containing the square cylinder, on which the homogeneous Neumann condition is imposed
for the pressure. Periodic boundary conditions are used between the inlet and the outlet;
this necessitates a very long section downstream of the obstacle to allow the flow to relax to
the same condition at the inlet. Even with such precautions, the calculated drag coefficient
is nearly 40% lower than their finite-volume benchmark. Thus, solving this problem using
our schemes provides an opportunity to investigate the benefits of the non-homogeneous
Neumann boundary condition on pressure.

The geometry of the problem is schematically shown in Fig. 1. The channel width is
H and its length is $3H$, and a square cylinder of side $D = H/8$ is situated at the center
of the channel. Thus, the entry is at $x = -12D$ and the exit at $x = 12D$. The geometry
and length scales are modeled after similar computations of Breuer *et al.* [31] and Lee
et al. [11]. Initially SPH particles are deployed on a regular square lattice of side L_0. The
square cylinder is represented by wall particles and interior dummy particles at the same
spacing. The boundary conditions are imposed as explained in section 3.3, with a parabolic
velocity at the entry and zero normal-velocity-gradient at the exit. For the homogeneous
projection scheme, $p = 0$ at the exit and $\boldsymbol{n} \cdot \nabla p = 0$ on the other three sides of the channel
and on the surface of the square cylinder. For the rotational scheme, we require $\phi = 0$ at
the exit and $\boldsymbol{n} \cdot \nabla \phi = 0$ on all the other boundary segments. The sharp corners of the square
cylinder tend to cause spurious disturbances to the pressure of nearby particles. This is
alleviated by local smoothing: in computing the pressure gradient ∇p^k in the prediction
step, the corner pressure is assumed equal to that of the nearest wall particle.

In presenting results in the following, we scale length by the width of the square cylinder
D, velocity by the centerline velocity U_0, time by D/U_0 and pressure by $\frac{1}{2}\rho U_0^2$. Most
computations to be reported below correspond to a Reynolds number $Re = U_0 D/\nu = 1$.

Higher Re results will be compared with prior studies in subsection 4.3.3. As a benchmark, we have solved the same problem using our in-house finite-element software AMPHI [32]. The finest finite-element mesh size is approximately $0.06D$ and a total of 39973 elements are used. Mesh refinement shows that this spatial resolution is sufficient; doubling the total number of elements causes a 0.7% change in the drag.

4.3.1 Convergence with number of SPH particles

To examine how the SPH results depend on spatial resolution, we have carried out computations using the rotational projection scheme at five spatial resolutions, with $L_0 = H/40$, $H/58$, $H/80$, $H/120$ and $H/160$ and the total number of particles being 5490, 11664, 21300, 46260 and 79860, respectively. The results are then compared with the finite-element (FE) solution to establish convergence with respect to increasing particle numbers, and to determine the level of resolution needed for an accurate solution that does not depend on the number of particles. The SPH solutions at the five spatial resolutions show the same qualitative trend. Starting with zero velocity throughout the domain except for the parabolic velocity profile imposed at the entry, the flow develops in time and approaches a steady state at $t \approx 1$. This "steady state", of course, is in an average sense that allows fluctuations due to the motion of individual particles. In the following, we will mostly discuss steady-state results.

First, we examine how the drag on the square cylinder converges with increasing spatial resolution. The drag force f_d, along the flow direction (x), is computed by integrating the pressure and viscous stresses around the surface of the square cylinder:

$$f_d = \int_{S_x} \left(-p + 2\mu \frac{\partial u}{\partial x} \right) n_x dy + \int_{S_y} \mu \frac{\partial u}{\partial y} n_y dx = - \int_{S_x} p n_x dy + \mu \int_{S_y} \frac{\partial u}{\partial y} n_y dx, \qquad (31)$$

where S_x refers to the left and right faces of the cylinder with $n_x = \pm 1$ being the x component of the outward normal vector, and S_y the top and bottom faces of the cylinder with $n_y = \pm 1$ being the y component of the outward normal. The viscous normal stress vanishes at the solid surface S_x because $\partial u/\partial x = -\partial v/\partial y = 0$. We use the pressure of the surface particles on the square cylinder for p. The viscous stress (or velocity gradient) is evaluated using fluid and wall particles as well as dummy particles inside the cylinder, the
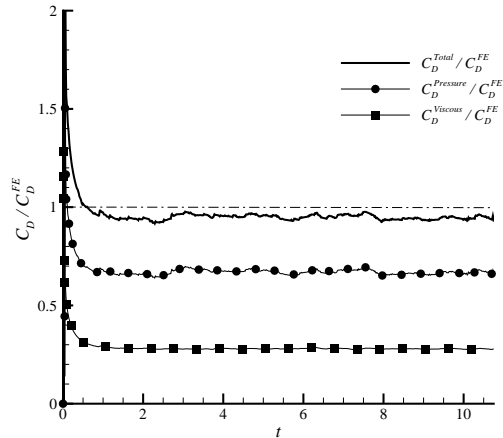
Figure 5: Temporal evolution of the drag coefficient C_D and its components due to pressure and viscous friction. The spatial resolution is at $L_0 = H/80$ with 21300 particles. The drag coefficient is scaled by the finite-element benchmark value C_D^{FE} and time is made dimensionless by D/U_0.

latter bearing velocity linearly extrapolated from that of the fluid and wall particles. This is known to produce the shear stress accurately on solid walls [27]. A drag coefficient is then defined as

$$C_D = \frac{f_d}{\frac{1}{2}\rho U_0^2 D^2}. \tag{32}$$

As an example, Fig. 5 shows the temporal variations of the drag force and its pressure and viscous components at a spatial resolution of $L_0 = H/80$. Note that the forces have been made dimensionless into drag coefficients, which are then scaled by the drag coefficient C_D^{FE} computed by finite elements under the same conditions. The steady state is attained around $t = 1$. The subsequent fluctuations are small, with a magnitude on the order of 2% of the mean. The total drag coefficient falls below the finite-element benchmark by some 5% at this spatial resolution, and the pressure contribution is about twice that due to viscous friction.

A time-averaged drag has been computed over a time period after attaining the steady state ($1 \leq t \leq 10$) to smooth out the fluctuations associated with the passage of individual particles around the cylinder. Figure 6 depicts this averaged drag coefficient for the five particle numbers in terms of its deviation from the FE benchmark C_D^{FE}. For the rotational
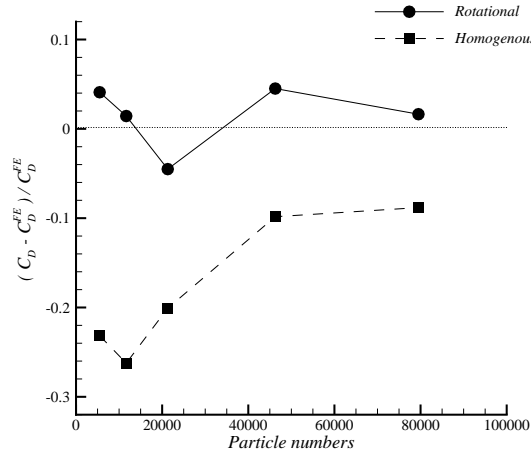
Figure 6: Drag coefficient C_D computed by the rotational and homogeneous projection schemes at five particle numbers. The data are shown as fractional deviations from the benchmark FE drag coefficient C_D^{FE}.
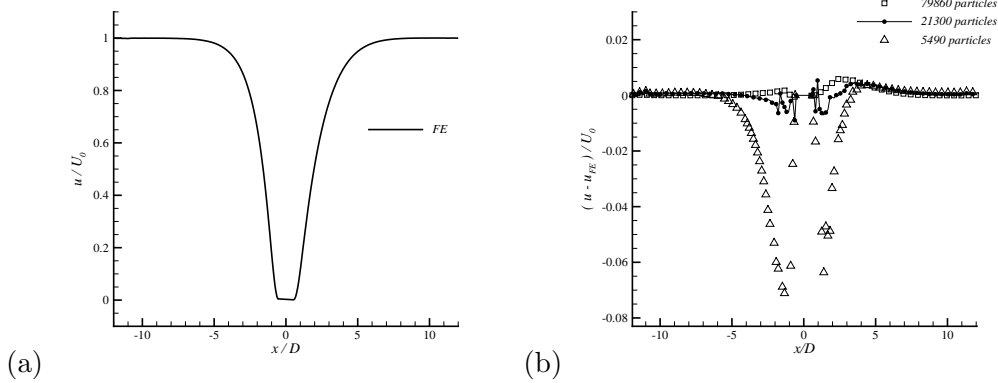

Figure 7: (a) The centerline velocity profile $u(x)$ computed by finite elements. (b) Deviation of the SPH velocity profiles from the FE profile computed using increasing numbers of particles.

scheme, C_D oscillates around C_D^{FE} and tends to converge to the latter as the number of particles increases. The maximum deviation is 5%. At the finest resolution, with 79860 particles, the drag is predicted within 2% of the benchmark. For comparison, C_D computed using the homogeneous scheme is also plotted and will be discussed in the next subsection.

Now we examine convergence of the velocity and pressure fields inside the computational domain. The time-averaged centerline profiles of the horizontal velocity $u(x)$, computed
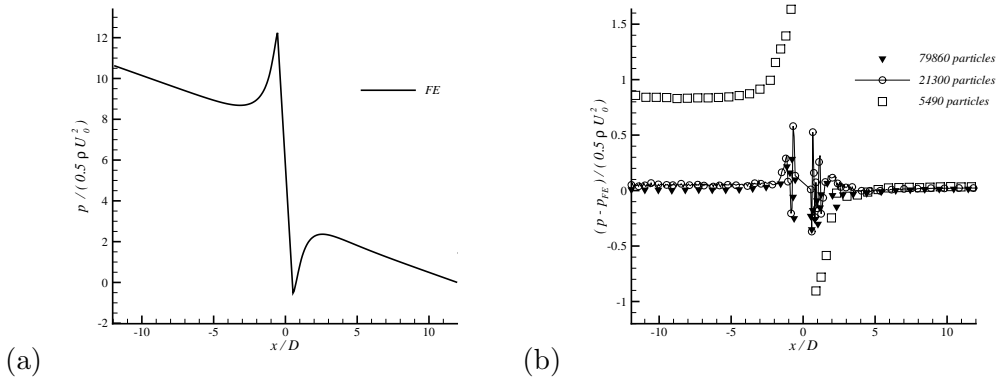
21

Figure 8: (a) The centerline pressure profile $p(x)$ computed by finite elements. (b) Deviation of the SPH pressure profiles from the FE profile computed using increasing number of particles. Pressure has been scaled by $\frac{1}{2}\rho U_0^2$ and zeroed at the exit of the channel ($x = 12D$).

using increasing numbers of particles, are compared with the FE profile in Fig. 7. In this and subsequent centerline profiles, the time-averaging is done as follows. We record the longitudinal position x, velocity u and pressure p of particles within $0.025L_0$ of the centerline for a time period. Then we divide the centerline into a number of bins of equal width and average the u and p for particles within each bin to produce the centerline profiles. For clarity we omitted the results for 11664 and 46260 particles, as they conform to the same trend of convergence. The convergence with increasing number of particles is evident, with the maximum error relative to the characteristic velocity U_0 being around 7% for 5490 particles, 1% for 21300 particles and below 0.5% for 79860 particles.

The convergence with increasing number of particles is also shown by the pressure profiles of Fig. 8. The greatest errors in pressure occur immediately upstream and downstream of the square cylinder. Relative to the FE pressure drop across the obstacle, the maximum pressure error in SPH computations drops from about 13% for 5490 particles to 5% for 21300 particles and finally to 3% for 79860 particles. Note that this is several times greater than the velocity errors. But in comparison with the weakly compressible SPH scheme, the pressure is more accurately computed by a wide margin; WCSPH underpredicts the centerline pressure behind the square cylinder by up to 30% [11].

Based on the above discussions, it is clear that the SPH result converges with increasing number of particles, and $L_0 = H/80$, with 21300 particles, provides sufficient spatial resolution. Results presented hereafter are all for 21300 particles.
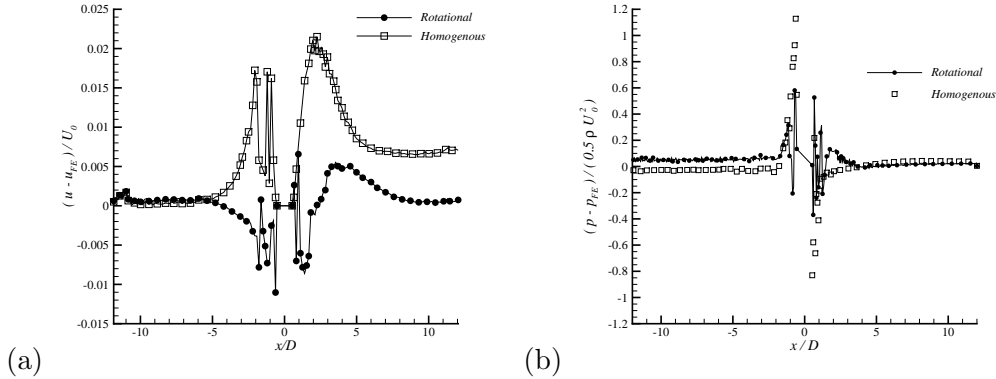
22

Figure 9: Comparison between (a) velocity profiles and (b) pressure profiles along the centerline computed by homogenous and rotational schemes. We plot the deviation of the SPH results from the FE benchmarks in Figs. 7(a) and 8(a).

4.3.2 Superiority of the nonhomogeneous pressure boundary condition

To ascertain the superiority of the rotational scheme over the homogeneous one for flows more complex than unidirectional channel flows, we have used both schemes to simulate the channel flow around the square cylinder at a spatial resolution of $L_0 = H/80$, with 21300 particles. For both schemes, the overall behavior of the solution is the same. Starting with the parabolic velocity profile at the entry and zero velocity in the domain, the flow undergoes an initial transient before reaching a steady state. Although the transient is not of interest here, we note that the rotational scheme is more robust and allows twice as large a time step than the homogenous scheme without incurring numerical instability. This is a potential benefit for simulating transient flows.

The time-averaged centerline velocity and pressure profiles are compared between the two schemes in Fig. 9, using the FE profiles of Figs. 7(a) and 8(a) as the baseline. The maximum velocity error is roughly 2% for the homogeneous scheme, and 0.8% for the rotational scheme. Similarly the pressure error relative to the FE pressure drop across the cylinder (Fig. 8a) decreases from roughly 9% for the homogeneous scheme to 5% for the rotational one. For the homogeneous scheme, the larger errors at the front and the back of the square cylinder can be ascribed to the homogeneous Neumann boundary condition $dp/dn = 0$. At the entry of the channel, there is a pressure boundary layer similar to
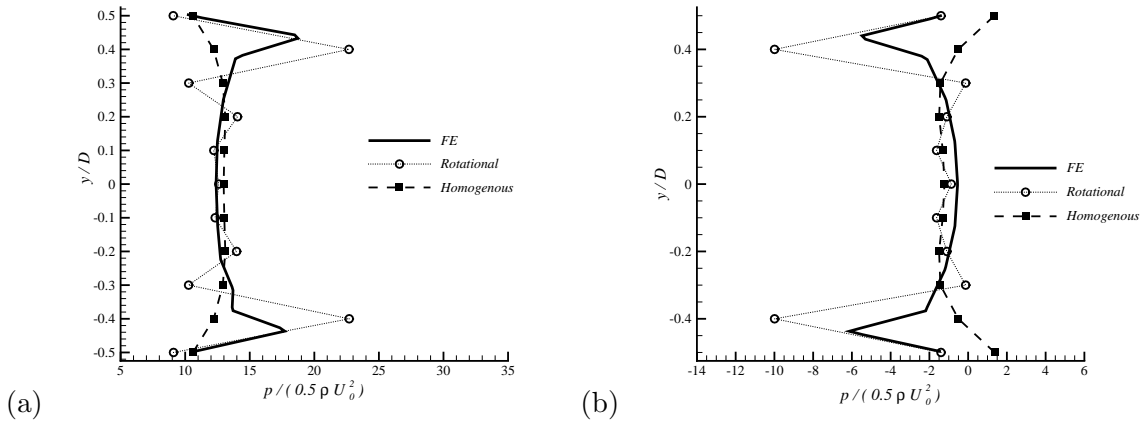
Figure 10: Pressure profile along the front (a) and the back (b) of the square cylinder computed by finite elements and the homogenous and rotational schemes. In all three computations p is zeroed at the exit.

Fig. 4(a). But its magnitude is so much smaller than that at the solid cylinder as to be invisible in the plot.

We further compare the rotational and homogeneous schemes by the pressure distribution along the front and back faces of the square cylinder (Fig. 10). For the SPH results, each data point represents the time-averaged pressure on a wall particle. Within the central part of the solid surface, roughly $-0.2 < y/D < 0.2$, both SPH profiles stay fairly close to the FE profile. The velocity is low immediately upstream and downstream of the cylinder, and there is no recirculating eddy at $Re = 1$. Thus, p is relatively uniform in these "dead zones". At the corner regions, however, the FE solution features sharp peaks in the front and valleys in the back, corresponding to the rapid turning of the streamlines. To a good degree, the rotational scheme captures these features while the homogeneous scheme misses out completely. But note the oscillations in the rotational profiles, indicative of the greater sensitivity of the scheme to errors in enforcing the solenoidality of the velocity.

This difference in the surface pressure bears directly on the computed drag force on the cylinder. As the homogeneous scheme underpredicts p at the front and overpredicts it at the back, it produces a drag that is between 10% and 27% below the correct value in Figure 6, depending on spatial resolution. In contrast, the prediction of the rotational scheme falls within 5%. Based on the results discussed in this subsection, we may conclude that the rotational scheme, with the physically consistent nonhomogeneous Neumann condition for pressure, has accomplished the goals set out at the beginning of this work.

4.3.3 Higher Reynolds numbers

So far we have discussed numerical results at $Re = 1$ for flow around the square cylinder. This problem was chosen partly because Breuer *et al.* [31] and Lee *et al.* [11] have published similar simulations using finite volume, lattice-Boltzmann and SPH methods. Their computations are for Reynolds numbers up to a few hundred. In the following, we present computations using the rotational projection scheme at higher Reynolds numbers and compare them with prior results as another test of the nonhomogeneous Neumann condition for pressure.

We have obtained accurate steady-state solutions at $Re = 5$, 10, 20, 30 and 50. The rotational projection scheme works equally well in all cases; the nonhomogeneous pressure boundary condition (Eq. 14) is apparently unaffected by the increasing inertia. However, higher Re does put a more stringent demand on the SPH scheme itself. For example, increasing the flow speed reduces the allowable time step via the CFL condition. At higher Re, fine wake structures may develop that demand higher spatial resolution, and a longer wake will require a longer domain. Both increase the magnitude of the computation and slow down convergence in our iterative solver for the pressure Poisson equation. In addition, we have also noticed that the simulation tends to become noisier at higher Re. For example, for $Re = 1$ there is no need to apply the XSPH velocity smoothing among neighboring particles [7], a remedy often used in the literature, especially in WCSPH scheme [9]. For $Re \geq 20$, however, we notice small spatial fluctuations in the wake of the cylinder that can be removed by velocity-smoothing with an XSPH coefficient of 0.02. Finally, the natural boundary condition at the exit, Eq. (17), tends to generate small lateral velocity disturbances at higher Re. The Dirichlet boundary condition is generally more rigid than the Neumann condition, and this observation is not specific to SPH. For Re above 10, therefore, we have imposed the parabolic Poiseuille velocity profile at the exit. That the domain length is sufficient for this profile to develop at the exit has been confirmed separately by finite-element computations.

The highest Reynolds number that we have tested is $Re = 100$, at which the SPH scheme fails to yield a converged solution. One expects an oscillating wake with vortex shedding in this case. Our scheme seems to incur a numerical instability with the oscillation growing in
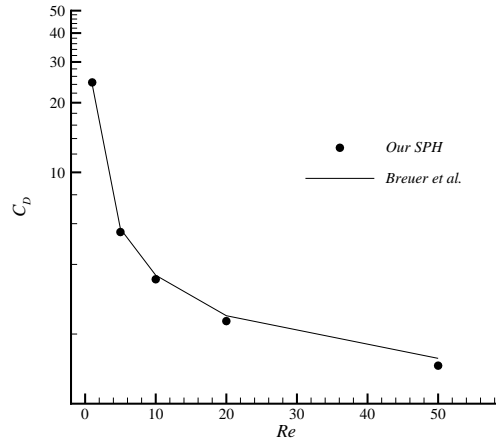
Figure 11: Drag coefficient C_D on the square cylinder as a function of the Reynolds number Re. The SPH computation with the rotational scheme uses 68000 particles. The lattice-Boltzmann results of Breuer *et al.* [31] are shown for comparison.

time, until the implicit solver for pressure fails. This may indicate that the wake now requires finer temporal and spatial resolutions and a longer domain length. Both the homogeneous and rotational schemes fail at this Re, suggesting that the failure is due to factors other than the projection scheme and pressure boundary condition. Therefore we have not investigated this further. In the following, we will compare the steady-state solutions, obtained using the rotational projection scheme, with those in the literature.

Figure 11 compares the drag coefficient on the square cylinder with previous predictions of Breuer *et al.* [31] using lattice-Boltzmann method, for Re up to 50. Our computational geometry is matched to that of [31], with $H = 8D$. Our domain length is $32D$, shorter than their $50D$, but the results are insensitive to this. In general, the SPH drag tends to fall slightly below that of the lattice-Boltzmann drag; the percentage error ranges from 2% for $Re = 1$ to under 7% for $Re = 50$. This is consistent with the accuracy noted in Figs. 5 and 6. Therefore, we consider the agreement satisfactory.

In a slightly different geometry, with a channel width $H = 5D$ and a length of $32D$, Lee *et al.* [11] computed the flow around the square cylinder using WCSPH and ISPH, and compared the results with a finite-volume benchmark. At $Re = 30$, they obtained a pressure drag coefficient $C_D^p = 2.46$ by finite volume. The ISPH yielded $C_D^p = 1.55$,

a 37% underprediction, which is consistent with the error of our homogeneous scheme in Fig. 6. The WCSPH scheme predicted a much larger $C_D^p = 6.49$, symptomatic of the spurious pressure in the weakly compressible treatment. Under identical conditions, our SPH method with the rotational scheme predicts $C_D^p = 2.14$, much closer to the benchmark than their ISPH using the homogeneous projection scheme. Note that Lee *et al.* [11] defined Re and C_D^p using the average velocity, and we have converted their values according to our definitions using the centerline velocity at the entry.

5 Conclusions

In this work, we have implemented a rotational projection scheme to compute incompressible flows using SPH. The main difference from conventional projection schemes is a nonhomogeneous Neumann boundary condition for the pressure Poisson equation that is consistent with the momentum balance. This resolves a well-known difficulty in projection schemes wherein imposing an artificial boundary condition of vanishing normal pressure gradient produces pressure boundary layers at the inlet and outlet to flow domains and on solid surfaces.

Applying the rotational projection scheme to two-dimensional channel flows and flow around a square cylinder, we have demonstrated that the new pressure boundary condition removes the spurious pressure boundary layers and markedly improves the accuracy of the solution. In particular, it predicts a drag coefficient on the cylinder within 7% of the correct value for Reynolds numbers up to 50, as compared with errors on the order of 20% incurred by the artificial pressure boundary condition.

Our numerical experimentation has shown the rotational projection scheme as robust, accurate and efficient in dealing with open boundaries and flow around solid obstacles. Based on these findings, we recommend it as a superior algorithm to the homogeneous projection scheme in computing truly incompressible flows using SPH. This removes a shortcoming in the formalism that has at times limited SPH simulations to periodic boundary conditions and hampered accurate evaluation of the pressure and drag force on solid obstacles. Although the scheme needs to be tested in more complex flow situations, it has

shown promise in extending the capacity of incompressible SPH to interesting and important problems previously inaccessible to SPH.

References

[1] R. A. Gingold, J. J. Monaghan, Smoothed particle hydrodynamics: theory and application to non-spherical stars, Mon. Not. Roy. Astron. Soc. 181 (1977) 375–389.

[2] L. B. Lucy, A numerical approach to the testing of the fission hypothesis, Astron. J. 82 (1977) 1013–1024.

[3] G. Oger, M. Doring, B. Alessandrini, P. Ferrant, Two-dimensional SPH simulations of wedge water entries, J. Comput. Phys. 213 (2006) 803–822.

[4] A. M. Tartakovsky, A. L. Ward, P. Meakin, Pore-scale simulations of drainage of heterogeneous and anisotropic porous media, Phys. Fluids 19 (2007) 103301.

[5] S. M. Hosseini, N. Amanifard, Presenting a modified SPH algorithm for numerical studies of fluid-structure interaction problems, Int. J. Eng. Trans. B 20 (2007) 167–178.

[6] J. J. Monaghan, SPH without a tensile instability, J. Comput. Phys. 159 (2000) 290–311.

[7] J. J. Monaghan, Smoothed particle hydrodynamics, Rep. Prog. Phys. 68 (2005) 1703–1759.

[8] J. J. Monaghan, Smoothed particle hydrodynamics, Ann. Rev. Astron. Astrophys. 30 (1992) 543–574.

[9] J. P. Morris, P. J. Fox, Y. Zhu, Modeling low Reynolds number incompressible flows using SPH, J. Comput. Phys. 136 (1997) 214–226.

[10] S. J. Cummins, M. Rudman, An SPH projection method, J. Comput. Phys. 152 (1999) 584–607.

[11] E.-S. Lee, C. Moulinec, R. Xu, D. Violeau, D. Laurence, P. Stansby, Comparisons of weakly compressible and truly incompressible algorithms for the SPH mesh free particle method, J. Comput. Phys. 227 (2008) 8417–8436.

[12] M. Yildiz, R. A. Rook, A. Suleman, SPH with the multiple boundary tangent method, Int. J. Numer. Methods Eng. 77 (2009) 1416–1438.

[13] A.J. Chorin, Numerical solution of the Navier-Stokes equations, Math. Comput. 22 (1968) 745–762.

[14] R. Temam, Sur l'approximation de la solution des equations de Navier-Stokes par la methode des pas fractionnaires ii, Arch. Ration. Mech. Anal 33 (1969) 377–385.

[15] P. M. Gresho, R. L. Sani, On pressure boundary conditions for the incompressible Navier-Stokes equations, Int. J. Num. Methods Fluids 7 (1987) 1111–1145.

[16] J. L. Guermond, P. Minev, J. Shen, An overview of projection methods for incompressible flows, Comput. Methods Appl. Mech. Engrg 195 (2006) 6011–6045.

[17] L. J. P. Timmermans, P. D. Minev, F. N. van de Vosse, An approximate projection scheme for incompressible flow using spectral elements, Int. J. Numer. Methods Fluids 22 (1996) 673–688.

[18] J. van Kan, A second-order accurate pressure-correction scheme for viscous incompressible flow, SIAM J. Sci. Comput. 7 (1986) 870–891.

[19] A. Colagrossi, M. Landrini, Numerical simulation of interfacial flows by smoothed particle hydrodynamics, J. Comput. Phys. 191 (2003) 448–475.

[20] G. Oger, M. Doring, B. Alessandrini, P. Ferrant, An improved SPH method: Towards higher order convergence, J. Comput. Phys. 225 (2007) 1472–1492.

[21] J. K. Chen, J. E. Beraun, C. J. Jih, An improvement for tensile instability in smoothed particle hydrodynamics, Comput. Mech. 23 (1999) 279–287.

[22] R. Xu, P. Stansby, D. Laurence, Accuracy and stability in incompressible SPH (ISPH) based on the projection method and a new approach, J. Comput. Phys. 228 (2009) 6703–6725.

[23] J. M. Tang, C. Vuik, On deflation and singular symmetric positive semi-definite matrices, J. Comput. Applied Math. 206 (2007) 603–614.

[24] L. D. Libersky, A. G. Petschek, T. C. Carney, J. R. Hipp, F. A. Allahdadi, High strain Lagrangian hydrodynamics, J. Comput. Phys. 109 (1993) 67–75.

[25] S. M. Hosseini, M. T. Manzari, S. K. Hannani, A fully explicit three-step SPH algorithm for simulation of non-Newtonian fluid flow, Int. J. Num. Methods Heat Fluid Flow 17 (2007) 715–735.

[26] R. W. Randles, L. D. Libersky, Smoothed particle hydrodynamics: some recent improvements and applications, Comput. Methods Appl. Mech. Eng. 139 (1996) 375–408.

[27] H. Takeda, S. M. Miyama, M. Sekiya, Numerical simulation of viscous flow by smoothed particle hydrodynamics, Prog. Theor. Phys. 92 (1994) 939–960.

[28] M. Lastiwka, M. Basa, N. J. Quinlan, Permeable and nonreflecting boundary conditions in SPH, Int. J. Num. Methods Fluids 61 (2008) 709–724.

[29] J. P. Gray, Monaghan J. J., Swift R. P., SPH elastic dynamics, Comput. Meth. Appl. Mech. Eng. 190 (2001) 6641–6662.

[30] R. Nestor, M. Basa, N. Quinlan, Moving boundary problems in the finite volume particle method, In P. Maruzewski, editor, ERCOFTAC SIG SPHERIC Third International Workshop Proceeding, pages 109–114, Lausanne, Switzerland, 2008. EPFL.

[31] M. Breuer, J. Bernsdorf, T. Zeiser, F. Durst, Accurate computations of the laminar flow past a square cylinder based on two different methods: lattice-Boltzmann and finite-volume, Int. J. Heat Fluid Flow 21 (2000) 186–196.

[32] P. Yue, C. Zhou, J. J. Feng, C. F. Ollivier-Gooch, H. H. Hu, Phase-field simulations of interfacial dynamics in viscoelastic fluids using finite elements with adaptive meshing, J. Comput. Phys. 219 (2006) 47–67.