Accurate and efficient immersed-boundary interpolations for viscous flows

By S. Kang, G. Iaccarino AND P. Moin

1. Motivation and background

The Immersed Boundary (IB) method is a technique for solving flow problems with irregular boundaries using a simple Cartesian grid solver. Specifically, the computational grid does not conform to all the boundaries of the domain and the numerical algorithm in the vicinity of these *immersed* surfaces is modified to enforce the desired boundary conditions. This greatly reduces the difficulties of generating meshes for complex boundaries. In addition, the IB method has the potential of high efficiency as the computational cost per grid cell is generally lower than that of general-purpose unstructured (body fitted) grid solvers.

There are numerous variants of the IB method, mainly in relation to the specific treatment of the computational cells crossing the immersed surfaces. Two major family of approaches can be distinguished: *(i)* based on strict finite volume discretization based on the cells cut by the IB (cut-cell approach) and *(ii)* based on forcing terms or solution interpolation in the vicinity of the IB.

The objective of the present study is to further develop an IB method based on forcing which has sufficient accuracy for Large Eddy Simulations (LES) at high Reynolds numbers with minimal increase in computational cost with respect to the simple, underlying Cartesian grid solver. The starting point for the present IB method is the interpolation method described in Fadlun *et al.* (2000). In addition to its simplicity, it has several advantages. The velocity boundary condition is enforced with implicit forcing, there is no severe limit on the time step, and the velocity components from the regions across the immersed boundary are decoupled. In the present study, revisions to this approach are introduced with the objective of increasing the accuracy and consistency of this method. In addition we analyze the practical importance of strict mass conservation.

In the next section, a general description of the numerical method used is given. The analysis of a various interpolation methods for the velocity field near the IB are presented in section 3. In addition, the issues of mass conservation and pressure accuracy are addressed. Results from numerical tests for laminar and turbulent flows are shown in section 4.

2. Numerical Method

The Navier-Stokes equations for an incompressible fluid are solved on a Cartesian grid using a staggered arrangement of the variables. A variant of the fractional step method (Kim & Moin 1985) is employed. The Crank-Nicholson scheme and the third-order Runge-Kutta scheme for the discretization of the diffusive and convective terms are used, respectively. An approximate ADI scheme is used to solve the momentum equation; to solve the Poisson equation for the pseudo-pressure either a direct method using the fast Fourier transform (FFT) or a multigrid method is used. In general, algorithms are

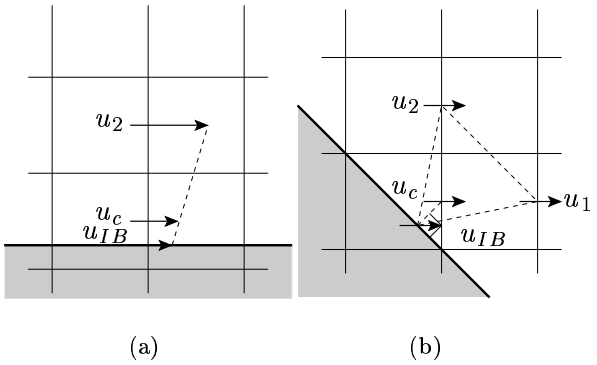


FIGURE 1. Configuration of the immersed boundary, grid and nodes in the linear interpolation method: ▬▬, immersed boundary; ———, grid; ⟶, velocity node.

parallelized using MPI, and the parallel tridiagonal matrix solver developed by Mattor *et al.* (1995) is used.

3. Interpolation methods at the immersed surfaces

In the original approach of Fadlun *et al.* (2000) the velocity components at the first grid point off the IB are determined using a linear interpolation formula rather than the discretized Navier-Stokes equation. The method is equivalent to assuming a one-dimensional linear velocity profile near the boundary. As consequence of the interpolation the velocity components are independent of each other and each component is determined by a separate interpolation. In the present study, four different interpolation methods have been considered:

- Linear interpolation method (LIM)
- Revised linear interpolation method (RLIM)
- Quadratic interpolation method (QIM)
- Quadratic+momentum interpolation method(QMIM)

3.1. *Linear interpolation method*

Fig. 1 shows two IB configurations commonly found in practical problems. In Fig. 1 (a), since the IB is approximately parallel to grid lines, a linear interpolation with u_{IB}, u_c and u_2 is easily formed along the x_2 coordinate. In Fig. 1 (b), there are two velocity components (u_1 and u_2) nearest in the horizontal and vertical directions to u_c. Then, a triangle is constructed by using two adjacent velocity nodes and a point on the IB (u_{IB}) surrounding u_c. Then, a linear interpolation stencil is built using the information available at the vertices of the triangle. The resulting interpolation formula has the following form:

$$u^k_{i,c} = w_{i,1}u^k_{i,1} + w_{i,2}u^k_{i,2} + w_{i,IB}u^k_{i,IB}, \tag{3.1}$$

where subscripts 1 and 2 denote adjacent velocity nodes in the x_1 and x_2 directions, and subscript IB denotes the point on the IB which is the boundary-normal projection of the velocity node c. Superscript k denotes the next time step, and w_i is an interpolation coefficient determined by the geometric configuration. Extending this method to a three-

dimensional geometry is straightforward. In the framework of the fractional step method the linear interpolation can be applied to either $\hat{u}_{i,c}$ or $\Delta u_{i,c} = \hat{u}^k_{i,c} - u^{k-1}_{i,c}$.

Assuming a local coordinate system whose center is located at the velocity node c, we can restate the linear interpolation method as:

$$\begin{aligned} u^k_i(x_1, x_2) = &\quad a^k_{i,1} x_1 + a^k_{i,2} x_2 + u^k_{i,c} \\ u^k_i = &\quad u^k_{i,IB} \text{ at immersed boundary,} \end{aligned} \tag{3.2}$$

where $a_{i,1}$ and $a_{i,2}$ are coefficients determined by the geometry of the local IB and velocity components.

When comparing Eq. (3.1) with the discretized momentum equation it can be shown that the linear interpolation method does not explicitly account for effects of the some of the terms, i.e. the pressure forcing, the temporal variation, etc. It can be argued that through the use of the two velocity components, $u_{i,1}$ and $u_{i,2}$ (determined by the discretized momentum equation), the effects of these terms are *indirectly* included.

A notable feature of the linear interpolation method (Eq. (3.1)) is that there is no explicit contribution from the velocity field at the previous time step. Numerical errors, once generated, are accumulated and/or transfered to other regions by advection and diffusion. A problem we encountered in numerical experiments is the occurrence of abnormal pressure fluctuations around the volume cells crossed by the IB. This effect is equivalent to the so-called "checker-board effect" observed in a non-staggered grid system where the solution of the momentum equation does not reflect the local pressure gradient.

3.2. *Revised linear interpolation method*

It is possible to derive an interpolation formula based on both $\hat{u}^k$ and Δu. After some manipulations and introducing a blending factor η we obtain:

$$\begin{aligned} \hat{u}^k_{i,c} = &\; w_{i,1}\hat{u}^k_{i,1} + w_{i,2}\hat{u}^k_{i,2} + w_{i,IB}\hat{u}^k_{i,IB} \\ &+ \eta \left(u^{k-1}_i - w_{i,1}u^{k-1}_{i,1} - w_{i,2}u^{k-1}_{i,2} - w_{i,IB}u^{k-1}_{i,IB} \right), \end{aligned} \tag{3.3}$$

The effect of the local pressure gradient can be accounted for by using:

$$\begin{aligned} \hat{u}^k_{i,c} = &\; w_{i,1}\hat{u}^k_{i,1} + w_{i,2}\hat{u}^k_{i,2} + w_{i,IB}\hat{u}^k_{i,IB} \\ &+ \eta \left(u^{k-1}_i - w_{i,1}u^{k-1}_{i,1} - w_{i,2}u^{k-1}_{i,2} - w_{i,IB}u^{k-1}_{i,IB} \right) \\ &- (\gamma_k + \rho_k)\Delta t \left(\left.\frac{\partial p^{k-1}}{\partial x_i}\right|_c - w_{i,1} \left.\frac{\partial p^{k-1}}{\partial x_i}\right|_1 - w_{i,2} \left.\frac{\partial p^{k-1}}{\partial x_i}\right|_2 - w_{i,IB} \left.\frac{\partial p^{k-1}}{\partial x_i}\right|_{IB} \right). \end{aligned} \tag{3.4}$$

where the coefficient γ_k and ρ_k are related to the time integration scheme. It is easy to prove that the numerical accuracy of Eq. (3.4) is the same as the original linear interpolation formula - second order in space. Also, note that what is interpolated is $\hat{u}^k_i - u^{k-1}_i + (\gamma_k + \rho_k)\Delta t \partial p^{k-1}/\partial x_i$ when $\eta = 1$. Coupling between neighboring pressure points is then considered with this "simulated" momentum equation. This approach is referred to in the following as Revised Linear Interpolation Method (RLIM).

In practice, computing $\partial p^{k-1}/\partial x_i$ at 1, 2 and c locations is straightforward since these points are existing velocity nodes at the cell faces, but interpolation is necessary to

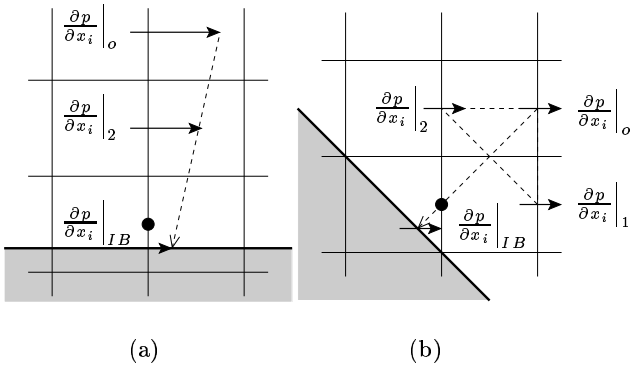


FIGURE 2. Points participating in interpolation of the pressure gradient on the IB.

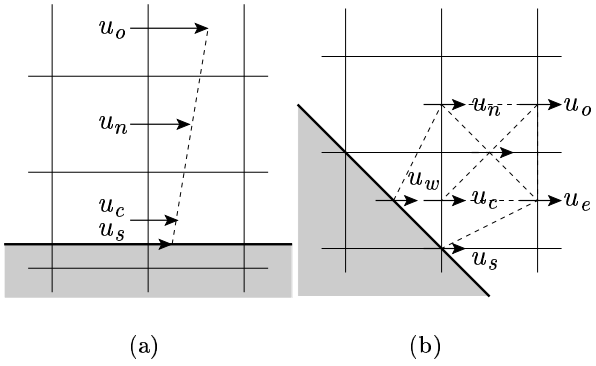


FIGURE 3. Configuration of the immersed boundary, grid and nodes in the quadratic interpolation method: ——, immersed boundary; ——, grid; ⟶, velocity node.

compute $\partial p^{k-1}/\partial x_i$ at the IB location. Applying linear interpolation using $\partial p^{k-1}/\partial x_i$ at 1, 2 and c is not appropriate in this case. So, an additional point near c is used in interpolation.

Fig. 2 shows a schematic representation of the stencil used for the pressure interpolation. This form of extrapolation has second order accuracy but becomes unstable when the c and IB points are close to each other. It is used only when η is larger than user-defined η_{crit}. Other alternatives to $\partial p^{k-1}/\partial x_i$ at IB are $\partial p^{k-1}/\partial x_i$ at c location and $\partial p^{k-1}/\partial x_i = 0$. The former produces a smaller error in the present numerical experiments and it is used when η is smaller than η_{crit}. This approximation produces a first order error term which is found not to affect the accuracy as shown in § 4.1. For the problems tested in the present study, η_{crit}=0.4 is used.

3.3. *Quadratic interpolation method*

The quadratic interpolation formula considered in the present study is:

$$\hat{u}_i^k(x_1, x_2) = a_{i,1}^k x_1^2 + b_{i,1}^k x_1 + a_{i,2}^k x_2^2 + b_{i,2}^k x_2 + \hat{u}_{i,c}^k, \tag{3.5}$$

where the origin of the local coordinate system is located at the position of $\hat{u}_{i,c}$ which is determined by the interpolation formula.

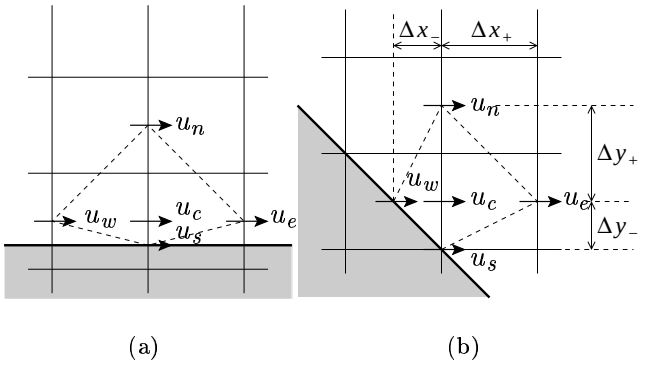


FIGURE 4. Configuration of the immersed boundary, grid and nodes in the quadratic+momentum interpolation method: ▬▬, immersed boundary; ———, grid; ⟶, velocity node.

Fig. 3 shows two IB configurations of interest. The number of velocity points needed to construct the quadratic formula is six (including $\hat{u}_{i,c}$). We can select $\hat{u}_{i,c}$ and the four surrounding velocity points. In addition, another point (subscript o) is necessary to define the interpolation stencil. In Fig. 3 (a), we choose the additional point to be the third velocity point away from the IB. Then, the points to the the left and right of $\hat{u}_{i,c}$ are ignored and a quadratic formula is generated using the four points along the coordinate normal to the IB line. In Fig. 3 (b), we choose the additional point as the center of four additional velocity nodes near the IB line.

With this quadratic interpolation formula, it is possible for the local velocity to have a non-linear profile. Also, only the spatial accuracy is increased and temporal behavior remains the same of the LIM.

3.4. *Quadratic+momentum interpolation method*

In the quadratic interpolation formula described above, there are five unknowns (including $\hat{u}^k_{i,c}$) all determined using velocities at fluid points computed through the solution of the governing equations. It is possible to follow another approach which reduces the stencil required to *close* the interpolation formula. In Fig. 4 the four adjacent velocity points are used to define four equations to determine the coefficients. The fifth relation is obtained from enforcing the momentum equations with a second order finite difference. The pressure gradient at a velocity node can be easily computed in the present staggered grid system although if some of the pressure points used may be in the boundary region (shaded region in Fig. 4). Interpolation coefficients are determined dynamically depending on the spatial and temporal flow field rather than having fixed values depending on geometry. The quadratic+momentum interpolation method (QMIM) has similarity to the cut-cell approach as a formal discretization of the momentum equation that accounts for the physical location of the IB is used.

The QMIM is computationally more complex than the LIM due to explicit evaluation of the convective and viscous terms. The overhead is about 10∼15% of the original computation time for the two-dimensional test cases presented afterward.

3.5. *Enforcing mass conservation*

The interpolation methods considered focus on the velocity and the momentum equation. There is another issue that plays a crucial role in practical calculations: mass conservation.

In IB approaches, the computational domain is divided into fluid regions where solutions to the Navier-Stokes equations are desired, and solid regions (e.g. shaded area in Fig. 1) where no solution is needed. The immersed surface separates these two regions. In Fadlun *et al.* (2000), the interpolation method determines the velocity components near the IB in the flow region. These components are not directly related to the velocity field in the boundary region. But, the flow field in the fluid region is affected by the velocity field in the solid region, since conservation of mass is satisfied for all the computational cells regardless of the presence of the IB. Mass conservation is not satisfied for the virtual cells obtained from cutting the fluid cells with the IB, and the subsequent errors may not be negligible.

To address mass conservation, the basic idea is to build new computational cells formed by the IB and existing cell faces, and to completely decouple the solid region from the fluid region. that the velocity. This is closely related to the finite volume method for an unstructured polyhedral mesh, and it is similar to the mass forcing concept by Kim *et al.* (2001) and cut-cell approaches (Ye *et al.* (1999); Kirkpatrick *et al.* (2003), among others. The present IB method, therefore, uses an efficient interpolation method for the velocity components and a modified finite volume operator for mass conservation at the immersed surfaces.

The finite volume discretization for the cut-cells introduces substantial complexity and, moreover, the computational stencil requires the modification of the linear system solver with potential negative impact on the efficiency of the code. An alternative approach is to consider strict flux conservation only for the explicit terms in the discretization stencil, leaving the implicit matrix unchanged (*lagged* correction). We refer to the former approach as a *strict* conservation and the latter as *approximate* conservation.

3.6. *Recovering the pressure field*

The final aspect of the IB algorithm that we investigated is the compatibility between the pressure field and the velocity field after any of the interpolation operators is used.

The basic idea is to perform an additional correction step (solving an extra Poisson equation) after the standard pressure correction equation. This equation is obtained by inserting the interpolated field at IB (obtained using any of the previously described interpolation formula) in the momentum equation and taking its divergence.

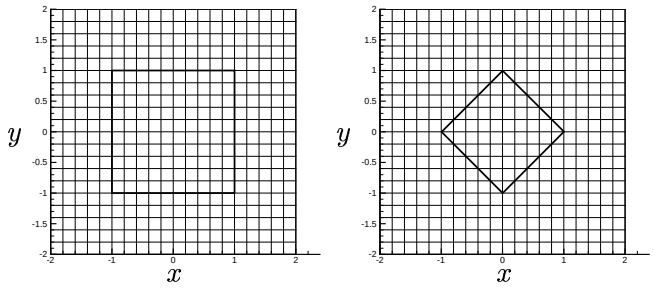
The amount of correction is clearly different depending on the specific interpolation method used. Our numerical experiments show that the QMIM interpolation requires the least correction.

4. Numerical experiments: decaying vortex problem

The problem of decaying vortices in a periodic has been widely used as a benchmark case to test the accuracy of numerical schemes, since it is an unsteady problem with an analytic solution. The velocity and pressure fields are given as:

$$u(x,y,t) = -\cos\pi x \sin\pi y \, e^{-2\pi^2 t/Re}, \tag{4.1}$$

$$v(x,y,t) = \sin\pi x \cos\pi y \, e^{-2\pi^2 t/Re}, \tag{4.2}$$



(a) Square (b) Rotated

FIGURE 5. Grid and IB configuration for a decaying vortex problem: (a) IB lines aligned on the grid lines; (b) IB lines inclined by 45° with respect to the grid lines.

$$p(x,y,t) = -0.25\,(\cos 2\pi x + \cos 2\pi y)\,e^{-4\pi^2 t/Re}. \tag{4.3}$$

In the present study, the exact flow field at $t = 0$ is integrated in time. The Reynolds number is set to 10. The computed flow field at $t = 0.2$ is compared with the analytic solution. The order of accuracy is investigated by computing the maximum (L_∞) error of the u_1 velocity and pressure fields with different grid sizes and time steps. A set of grid sizes (Δx=0.2, 0.1, 0.05 and 0.025) is chosen with a set of time steps (Δt=0.02, 0.01, 0.005 and 0.0025) as parameters. Both grid size and time step are halved to test the order accuracy of the numerical scheme in space and time at once.

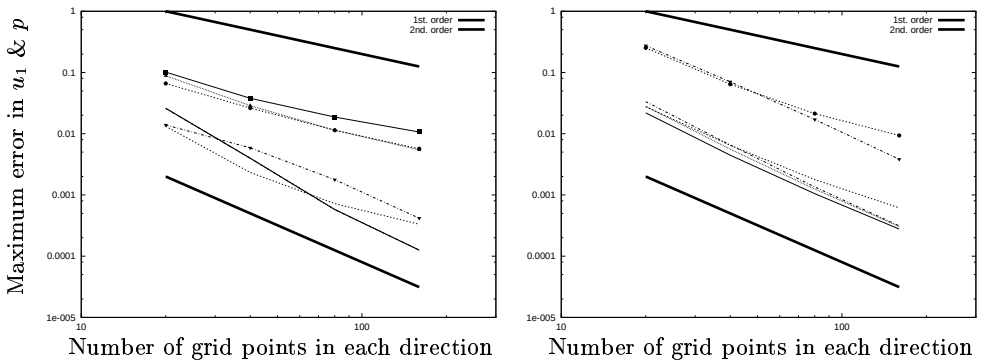
4.1. *Linear interpolation methods*

In this section, results with the linear interpolation methods introduced in § 3.1-3.2 are shown. The list of the tested cases are:

A. LIM for $\hat{u}^k$, Eq. (3.1)
B. LIM for Δu, the delta form of Eq. (3.1)
C. Mixed LIM of A and B, Eq. (3.3)
D. RLIM, Eq. (3.4)

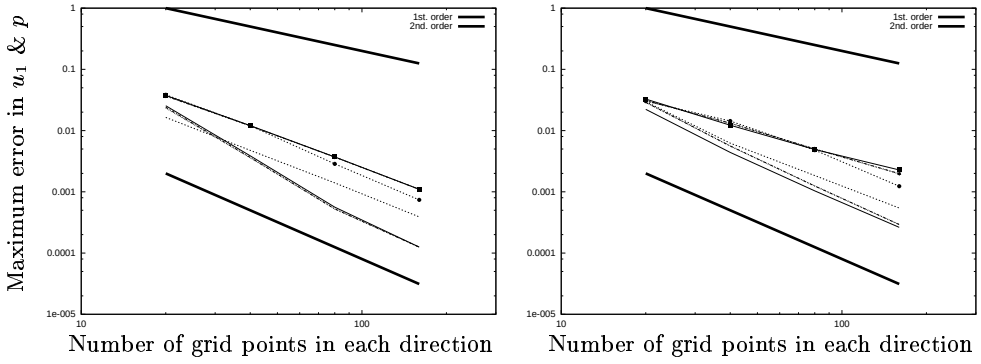
Tested IB geometries are the same as those used by Kim *et al.* (2001). In these geometries, grid lines in both directions and the IB lines meet at the edge points of the grid cells, which maintains a constant configuration between the grid and the IB lines independent of the number of grid points used. This problem setup allows for an effective investigation on the accuracy of IB methods. Fig. 5 shows the test geometries with the grid. Unless specified otherwise, the velocity field inside the IB geometry is solved and the velocity field outside of the IB is set to zero.

Fig. 6 shows the maximum error in u_1 and p with different numbers of grid points. When the IB line coincides with a grid line (square IB in Fig. 5 (a)), every interpolation method produces an acceptable result. In this case, the maximum absolute value of $\nabla \cdot u$ is $10^{-3} \sim 10^{-4}$ without any special treatment for mass conservation. But, when the IB line is inclined by 45° with respect to grid lines (rotated IB in Fig. 5 (b)), a large error ($\sim 10^2$) in the pressure is found for the LIM and mixed LIM cases (not shown in the figure). This indicates the occurrence of the local pressure build-up mentioned earlier. Interestingly, the accuracy of the velocity field is still acceptable. The LIM for Δu and the RLIM (Eq. (3.4)) show relatively reasonable pressure error. Between the LIM for Δu and Eq. (3.4), the accuracy of the pressure is shown to be better for the latter,



(a) Square (b) Rotated

FIGURE 6. Maximum error in u_1 and p at t=0.2 with the linear interpolation schemes: ——— , LIM for $\hat{u}^k$; – – – –, LIM for Δu; · · · ·, mixed LIM; —·—, RLIM. The symbol denotes the pressure.

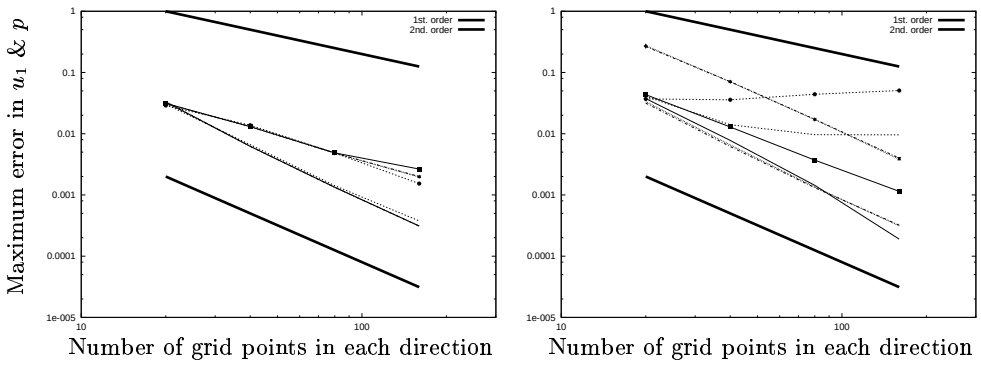


(a) Square (b) Rotated

FIGURE 7. Maximum error in u_1 and p at t=0.2 with the linear interpolation schemes and the pressure recovery scheme: ———, LIM for $\hat{u}^k$; – – – –, LIM for Δu; · · · ·, mixed LIM; —·—, RLIM. The symbol denotes the pressure.

which justifies the revision made by accounting for the local pressure gradient. Notably, with the LIM for Δu, the accuracy of the flow field is shown to decrease as the mesh becomes finer. For these cases, $\partial\phi/\partial\mathbf{n}$ is not zero at the IB since the approximation to the discretized Poisson equation is used. Thus, every velocity component is modified by the velocity projection step including the boundary conditions. If Δu at the IB is used as the boundary condition at the next time step, pre-existing error will not be reduced and will accumulate. The order of accuracy of the velocity field with the other interpolation methods is shown to be second order in Fig. 6.

Fig. 7 shows the maximum error with the pressure recovery scheme introduced in § 3.6. For the square IB, all error curves of the pressure collapse except for the LIM for Δu.



(a) (b)

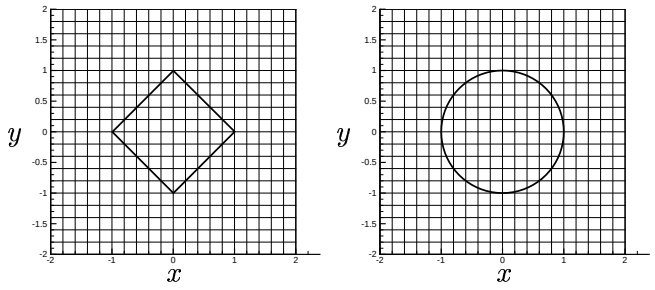
FIGURE 8. Maximum error in u_1 and p at t=0.2 for the rotated IB case: (a) with the LIMs, the pressure recovery scheme, and strict mass conservation: ——— , LIM for $\hat{u}^k$; - - - - , LIM for Δu; · · · · , mixed LIM; —·— , RLIM (b) with the different settings for mass conservation with the RLIM : ——— , with the analytic initial and boundary conditions outside of the IB; - - - - , with the analytic initial condition only; · · · · , with the approximate mass conservation; —·— , with strict mass conservation. The symbol denotes the pressure.

For all cases, the pressure accuracy is increased. There is no noticeable change in the accuracy of the velocity.

Fig. 8 (a) shows the maximum error with the pressure recovery scheme and strict mass conservation introduced in § 3.5 for the rotated IB case. The maximum value of $|\nabla \cdot u|$ is kept less than 10^{-5} at every time step. The accuracy of the velocity is not increased except when the LIM for Δu is used. For this method, however, the deterioration of the accuracy on finer meshes disappeared. With strict mass conservation, $\partial\phi/\partial\mathbf{n} = 0$ is satisfied at the IB, and there is no accumulation of error. The pressure accuracy is slightly different for the different interpolation methods.

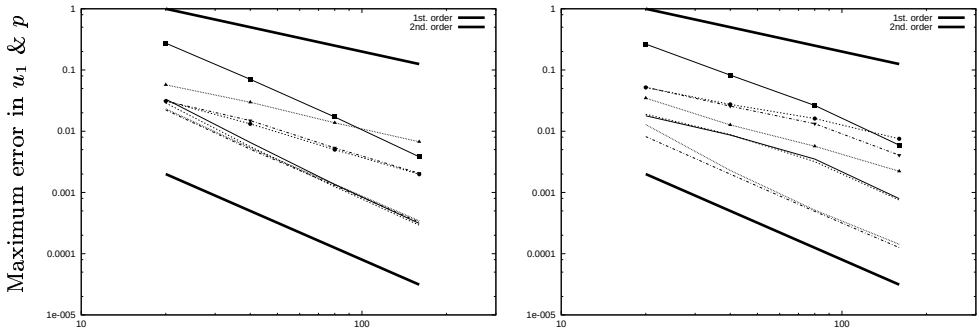
In order to further investigate the importance of mass conservation for the accuracy of the present IB method, two additional simulations without the mass conservation corrections of § 3.5 are tested. In this case the flow fields on both sides of the IB are coupled. The first set of results corresponds to the specification of the velocity and pressure outside the computational domain based on the exact solution. In the second calculation, on the other hand, the initial exact field (corresponding to $t = 0$) is used at all late times. Fig. 8 (b) shows the maximum error with the RLIM and the different option for mass conservation described above. The results indicates that the flow field in the flow region is affected to a large extent by the flow field in the boundary region if mass conservation for the IB cells is not properly satisfied. Also, the difference between the approximate and strict mass conservation cases is shown to be small.

To summarize, the IB method based on the LIM shows second order accuracy in time and space. Among the linear interpolation variants, the RLIM (Eq. (3.4)) is found to be most viable in the different IB geometries, even without strict mass conservation. In order to reduce the error in the pressure, the pressure recovery scheme is found to be more effective than strict mass conservation for the IB cells.



(a) Rotated (b) Circle

FIGURE 9. Grid and IB configuration for a decaying vortex problem: (a) IB lines inclined by 45° with respect to the grid lines; (b) IB line of a circular shape.

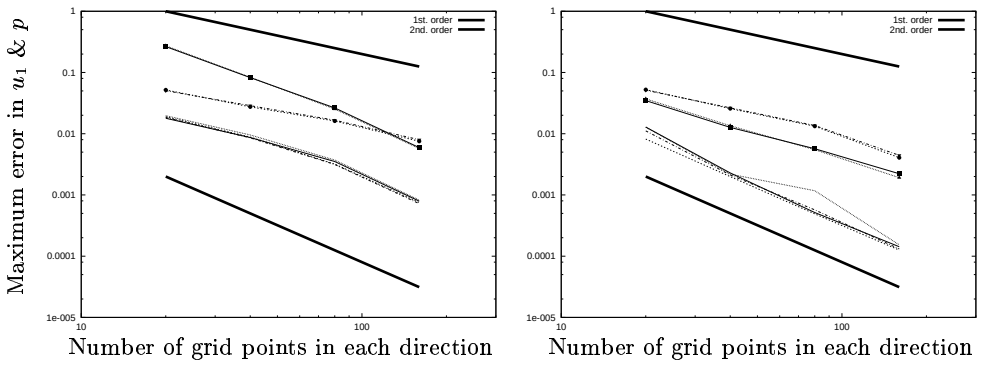


(a) Rotated (b) Circle

FIGURE 10. Maximum error in u_1 and p at t=0.2 with the different interpolation methods: ——— , RLIM; – – – –, RLIM with the pressure recovery scheme; · · · ·, QMIM; —·—, QMIM with the pressure recovery scheme. The symbol denotes the pressure.

4.2. *Quadratic interpolation method*

Quadratic interpolation method is here compared with the most accurate linear scheme presented in the previous section: the RLIM. These interpolation methods are tested with two IB geometries shown in Fig. 9. Fig. 9 (a) shows the rotated IB geometry identical to that in Fig. 5 (b). Fig. 9 (b) shows a circular IB geometry. This last configuration is more realistic as the intersection between the grid and the IB lines changes with the number of grid points. Considering that the error of an IB method is a function of distance between the fluid points and the IB, this configuration may cause some fluctuations in the observed accuracy. Fig. 10 shows the maximum error in u_1 and p with and without the pressure recovery scheme. In the rotated IB geometry, all cases show second order accuracy for the velocity. In the circular IB geometry, the QMIM (in § 3.4) shows second order accuracy for the velocity, while the RLIM (Eq. (3.4)) shows second order accuracy only when the grid spacing Δx is less than 0.05. The accuracy of the pressure is close to first order for the QMIM. The RLIM produces a higher order of accuracy for the



(a) RLIM (b) QMIM

FIGURE 11. Maximum error in u_1 and p at t=0.2 with the different interpolation methods and additional treatments in the circular IB: ——— , no additional treatment; – – – – , pressure recovery scheme; · · · · · , strict mass conservation; —·— , both treatments. The symbol denotes the pressure.

pressure, but the magnitude of the pressure error is larger than the QMIM for the grid spacings used.

With the pressure recovery scheme, velocity and pressure error is reduced in most cases. Also, the result confirms that the pressure recovery scheme is effective in reducing pressure error in a coarse mesh. It is noteworthy that the pressure recovery scheme seems to be less helpful in combination with the QMIM than with the RLIM.

Fig. 11 shows the maximum error in u_1 and p with the different additional corrections indicated before to investigate the combinations that are effective. A noticeable enhancement is observed when using strict mass conservation, but its effect is much smaller than that of using the pressure recovery scheme or different interpolation method. Considering the large amount of computational time used for strict mass conservation, it does not seem to be cost-effective for the cases tested.

In summary, the QMIM scheme is observed to be more viable than the other interpolation methods in a practical IB configuration. One weakness it showed is the lower order of accuracy in the pressure than the RLIM.

4.3. *Comparison of instantaneous flow fields*

As shown in the previous sections, all the interpolation methods considered in the present study show acceptable accuracy of the velocity and pressure, but some methods may result in spurious local pressure oscillations. Fig. 12 shows the instantaneous pressure contours obtained when using the LIM for $\hat{u}^k$ (Eq. (3.1)), revised LIM (Eq. (3.4)), and QMIM scheme (§ 3.4) for the circular IB geometry. The pressure recovery scheme is not used for these cases. A grid spacing of Δx=0.1 is used and the grid lines are shown in the figures. Large pressure oscillations and build-up at the cells crossed by the IB are clearly visible in the case of the LIM for $\hat{u}^k$. Much smaller errors are found in the pressure contours from the RLIM case; the results obtained with the QMIM are clean with no distortions.

Fig. 13 shows the instantaneous pressure contours from the different interpolation methods with the pressure recovery scheme. The pressure contours for all cases are now

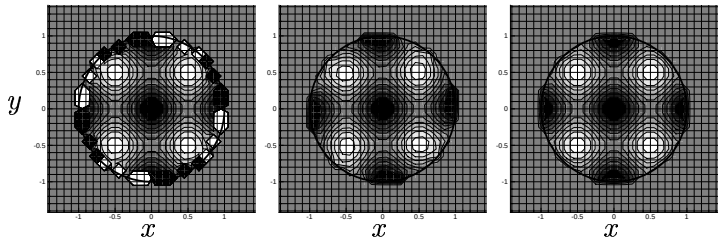


FIGURE 12. Instantaneous pressure contours computed using the different interpolation methods without the pressure recovery scheme: (a) LIM for $\hat{u}^k$; (b) RLIM; (c) QMIM.

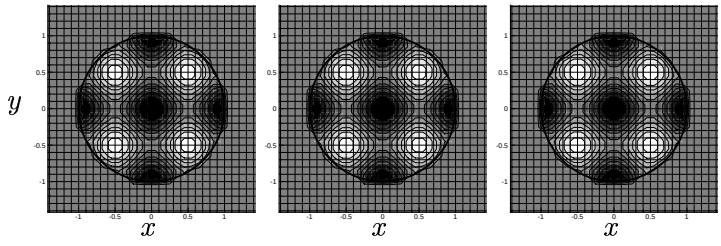


FIGURE 13. Instantaneous pressure contours computed using the pressure recovery scheme: (a) LIM for $\hat{u}^k$; (b) RLIM; (c) QMIM.

clean and look very similar. It must be pointed out that clean pressure contours are obtained for the LIM for $\hat{u}^k$ and RLIM as the grid resolution becomes very fine, even if the pressure recovery scheme is not used. This result confirms that a fine mesh resolution or the special treatment is necessary to get an acceptable pressure field when the LIM is used.

5. Conclusions

The immersed boundary method is a promising numerical method for flow problems with complex boundaries as it relaxes the difficulties associated with mesh generation. The major issues to be addressed are sufficient accuracy for LES or DNS and numerical efficiency. To this end, several revisions to the method of Fadlun *et al.* (2000) were considered in the present study.

In spite of second order accuracy shown theoretically as well as numerically, the simple linear interpolation method (LIM) was found to have incompatibilities with the time-marching scheme. This incompatibility is expressed as local error accumulated in the pressure field. In order to mitigate this problem, a revised version of the LIM was proposed. A few additional corrections were added by handling the contributions from some of the terms in the decoupled momentum equation explicitly. The decaying vortex test problem indicated that the RLIM is effective in reducing error in the pressure field while maintaining the accuracy of the original LIM. Another interpolation method was devised

by combining a quadratic interpolation formula and the momentum equation. The resulting scheme, QMIM, simulates the discretized Navier-Stokes equation more directly than the LIM, and is similar to the cut-cell approach in that a formal discretization of the momentum equation is used. In the numerical tests, the QMIM scheme produces smaller errors especially when for coarse grid resolutions. Second order accuracy for the velocity and higher than first order accuracy for the pressure were observed for both interpolation methods. Computational overhead was found to be less than 15% of the original computation time for the test case considered.

Another approach for addressing the incompatibility between the interpolation method and the time-marching scheme was introduced. A condition for the intermediate velocity field was enforced to obtain proper accuracy for the pressure. In order to satisfy this condition, a correction to the intermediate velocity field was considered. This additional correction, referred to as a pressure recovery scheme, reduced the velocity and pressure error in most cases. The pressure recovery scheme is especially effective for coarse meshes. The practical importance of mass conservation for the IB method was another issue addressed. The additional treatments for mass conservation are closely related to the finite volume method for an unstructured mesh. The different levels of mass conservation for new volume cells formed by the IB and existing cell faces were considered. This revision is very useful when the velocity boundary condition is not zero. In the numerical test, we observed that the flow field can be affected to a large extent by the boundary region if mass conservation for the IB cells is not carefully considered. Various approaches have been considered to solve this problem.

REFERENCES

FADLUN, E.A., VERZICCO, R., ORLANDI, P., AND MOHD-YUSOF, J. 2000 Combined immersed-boundary/finite-difference methods for three-dimensional complex flow simulations, *J. Comput. Phys.* **161**, 35.

KIM, J., KIM, D. AND CHOI, H. 2001 An immersed-boundary finite-volume method for simulations of flow in complex geometries, *J. Comput. Phys.* **171**, 132.

KIM, J. AND MOIN, P. 1985 Application of a fractional-step method to incompressible Navier-Stokes equations, *J. Comput. Phys.* **59**, 308.

KIRKPATRICK, M.P., ARMFIELD, S.W. AND KENT, J.H. 2003 A representation of curved boundaries for the solution of the Navier-Stokes equations on a staggered three-dimensional Cartesian grid, *J. Comput. Phys.* **184**, 1.

MATTOR, N., WILLIAMS, T.J. AND HEWETT, D.W. 1995 Algorithm for solving tridiagonal matrix problems in parallel, *Parallel Computing* **21**, 1769.

YE, T., R. MITTAL, R., UDAYKUMAR, H.S. AND SHYY, W. 1999 An accurate Cartesian grid method for viscous incompressible flows with complex immersed boundaries, *J. Comput. Phys.* **156**, 209.