The Multigrid/τ-extrapolation technique applied to the IBM.

F. Pacull[1] and M. Garbey[2]

[1] Mathematics Department, University of Houston, fpacull@math.uh.edu
[2] Computer Science Department, University of Houston, garbey@cs.uh.edu

The Immersed Boundary Method (IBM), originally developed by C.S. Peskin [Pes72], is a very practical method of simulating fluid-structure interactions. It combines Eulerian and Lagrangian descriptions of flow and moving elastic boundaries using Dirac delta functions. Incompressible Navier-Stokes and Elasticity theory can be unified by the same set of equations to get a combined model of the interaction.
There are numerous applications of the IBM in Bio-Engineering or in more general Computational Fluid Dynamics applications.
We present a numerical study of the accuracy and computational cost of the method, in the frame of finite-differences, based on the implementation of several mathematical tools such as multigrid solvers, τ-extrapolation technique, multilevel discretization and more generally numerical methods for differential equations with singular source terms. These implementations are being made on test-cases that are relevant for the IBM applications, keeping in mind that we want to keep the simplicity of the method.

1 The IBM

While we are using a more sophisticated time stepping scheme [Pes02], let us start with the basic projection scheme introduced by Chorin [Cho68] for the incompressible Navier-Stokes equations:
1- *Prediction step:*

$$\rho\left[\frac{V^{*}-V^{n}}{\Delta t}+\left(V^{n}.\nabla\right)V^{n}\right]-\nu\Delta V^{*}=F^{n},\quad V^{*}_{\partial\Omega}=V^{n+1}_{\partial\Omega}; \tag{1}$$

2- *Pressure correction step:*

$$\Delta P^{n+1}=\frac{\rho}{\Delta t}\nabla.V^{*},\quad \left(\frac{\partial P^{n+1}}{\partial\eta}\right)_{|\partial\Omega}=0; \tag{2}$$

3- *Correction step:*

$$\rho\left[\frac{V^{n+1}-V^*}{\Delta t}\right]+\nabla P^{n+1}=0. \qquad (3)$$

The notations are as follow: V, P, ρ and ν are respectively the velocity, pressure, uniform density and viscosity coefficient of the fluid. F is the force term, Δt, the time step, Ω, the domain. η is the outward normal vector to $\partial\Omega$, the boundary of the domain.
In this scheme we have a non-conservative convection term, an explicit force term and a semi-implicit diffusion term. Let $f(s,t)$ be the elastic force density along Γ. The boundary immersed in the fluid is represented in the Cartesian mesh by $X(s,t)$ where $0 \leq s \leq 1$ is the curvilinear coordinate and $0 \leq t \leq T$ is time. The force term F in Eq. 1 is obtained as follows:

$$F(x,t)=\int_{\Gamma} f(s,t)\delta(x-X(s,t))ds, \ \ (x,t)\in\Omega\times[0,T]. \qquad (4)$$

It is ideally zero everywhere except along Γ. In the computations, the δ function is regularized by a discrete Dirac delta function of compact support. Let us describe the force term in the two-dimensional case after discretization of the immersed boundary without considering the time dependency:

$$F_h(x)=h_\Gamma\sum_{j=1}^{M} f(s_j)\delta_h(x-X(s_j)), \ \ x\in\Omega. \qquad (5)$$

The immersed boundary is then a one dimensional line with this discrete mesh: $s_j=\frac{j-1}{M-1}=(j-1)h_\Gamma, \ \ 1\leq j\leq M.$
The Navier-Stokes equations implemented with a finite-difference method is of order two in space because of the discretization error but the order is reduced in the IBM by the discretization of the force term.
If we look at the prediction step (Eq. 1) of the projection scheme for the Navier-Stokes equations, we have:

$$(I-\nu\Delta t\Delta.)V^*=RHS^n, \qquad (6)$$

where the right-hand side contains singular components essentially due to the discrete force term that is a sum of discrete Dirac delta functions.
If we look at the pressure correction step (Eq. 2), we have:

$$\Delta(\delta P)^n=RHS^*, \qquad (7)$$

where the right-hand side also contains singular components but in the form of dipoles. Consequently we will focus our study on elliptic equations with singular source terms and more specifically these two operators $I-k^2\Delta$ $(k>0)$ and Δ. The standard IBM is first order in space. Our main goal is to get an order of accuracy larger than one and fast solvers for problems (6) and (7).

2 The discrete Dirac delta function δ_h

Let us introduce the discrete Dirac delta functions. They are usually written in this form in 1D : $\delta_h(x) = \frac{1}{h}\phi(\frac{x}{h})$. The function ϕ needs to satisfy several compatibility conditions:

1. $\phi \in C^0(\mathbb{R})$.
2. ϕ has to be of finite support, knowing that the computational cost of the method is proportional to its width.
3. If we are using the staggered mesh, as introduced by Harlow and Welch [Har65], which requires a regular rather than a wide stencil for the Laplace operator in the pressure equation, we just need to have:
$$\sum_{i\in\mathbb{Z}} \phi(r-i) = 1 \ \ \forall r \in \mathbb{R},$$
which guarantees that constant functions are interpolated exactly by δ_h. If we are not using the staggered mesh we would have this condition:
$$\sum_{i(even)} \phi(r-i) = \sum_{i(odd)} \phi(r-i) = \frac{1}{2} \ \ \forall r \in \mathbb{R}.$$
4. $\sum_{i\in\mathbb{Z}} [\phi(r-i)]^2 = C \ \ \forall r \in \mathbb{R}$ where C is a constant. That ensures that
$$\sum_{i\in\mathbb{Z}} \phi(r_1 - i)\phi(r_2 - 1) \le C \ \ \forall (r_1, r_2) \in \mathbb{R}^2.$$
J. M. Stockie wrote [Sto97] that it "is analogous to the physically reasonable requirement that when two fibre points interacts, the effect of one boundary point on the other is maximized when the points coincide".
5. $\sum_{i\in\mathbb{Z}}(r-i)\phi(r-i) = 0 \ \ \forall r$, which ensures along with property 3 that linear functions are interpolated exactly by δ_h.

The minimal width support of a function satisfying these requirements on a traditional mesh is $2h$. It is then defined uniquely, as presented by Peskin [Pes02]. For the staggered mesh a function with support $\frac{3}{2}h$ is uniquely determined too [Rom96]:

$$\phi(r) = \begin{cases} \frac{1}{6}(5 - 3|r| - \sqrt{-3(1-|r|)^2 + 1}, & 0.5 \le |r| \le 1.5; \\ \frac{1}{3}(1 + \sqrt{-3r^2 + 1}), & |r| \le 0.5; \\ 0, \ \ \text{otherwise}. \end{cases} \tag{8}$$

This function described in Eq. 8 gives an IBM that is somewhat faster computationally due to the fact that the support is $\frac{3}{2}h$ instead of $2h$. Engquist and Tornberg showed [Eng04] that the discretization error is proportional to the number of moment conditions satisfied by the function. But adding some moment conditions requires increasing the support. If we keep a discrete delta function that has a $2h$ support with the staggered mesh instead of a $\frac{3}{2}h$ [Rom96], we can then increase the accuracy of the method using this piecewise cubic function [Eng04]:

$$\phi(r) = \begin{cases} 1 - \frac{1}{2}|r| - |r|^2 + \frac{1}{2}|r|^3, \ 0 \le |r| \le 1; \\ 1 - \frac{11}{6}|r| + |r|^2 - \frac{1}{6}|r|^3, \ 1 < |r| \le 2; \\ 0, \ \text{otherwise}. \end{cases} \tag{9}$$

This function satisfies the properties above as well as these extra moment properties:

$$\sum_{i \in \mathbb{Z}} (r-i)^2 \phi(r-i) = 0 \ \ \forall r \in \mathbb{R}, \ \sum_{i \in \mathbb{Z}} (r-i)^3 \phi(r-i) = 0 \ \ \forall r \in \mathbb{R} \tag{10}$$

Our numerical experiment will use extensively this new discretization of the δ function, adapted to the staggered meshes.

3 The multigrid τ–extrapolation

The τ–extrapolation [Ber97, Rud04] is a modified multigrid method that improves the convergence order of a discrete problem. It is based on the Richardson extrapolation technique. It combines two solutions obtained on different grids in order to correct the fine grid solution but requires the knowledge of the order of the first asymptotic expansion term, which can be evaluated experimentally. If this order is α, we combine the fine solution u_h and the coarse solution u_H with this linear combination (u^* is the analytic solution):

$$\hat{u}_h = \left(\frac{2^\alpha}{2^\alpha - 1}\right) u_h + \left(1 - \frac{2^\alpha}{2^\alpha - 1}\right) u_H = u^* + o(h^\alpha). \tag{11}$$

Here is the τ–extrapolation multigrid algorithm for the problem $Au = f$:

- 1 - pre-smoothing step : $u_h = S^{\nu 1}(A_h, u_h, f_h)$,
- 2 - $u_h = u_h + I_H^h A_H^{-1} \left(\left(\frac{2^\alpha}{2^\alpha-1}\right) \hat{I}_h^H (f_h - A_h u_h) + \left(1 - \frac{2^\alpha}{2^\alpha-1}\right)(f_H - A_H I_h^H u_h)\right)$,
- 3 - post-smoothing step : $u_h = S^{\nu 2}(A_h, u_h, f_h)$,

with these characteristics in most cases:
- I_H^h is a trilinear interpolation prolongation operator,
- $\hat{I}_h^H$ is a full weighting restriction operator,
- I_h^H is a full injection prolongation operator,
- (ν_1, ν_2), the number of smoothing steps per iteration, are small (≤ 2).

The good convergence order property of the multigrid methods is due to the fact that the smoothing iterations improves the high frequency modes of the discrete solution while the coarse grid correction improves its low frequency modes. This is especially true for the stiff elliptic problems solved in the IBM. In the τ–extrapolation technique, the Richardson extrapolation linear combination significantly improves the discretization order of the coarse grid correction. This is the idea of the double discretization. A high order discretization scheme is used on the coarse grid, different from the scheme used for calculating the residuals transferred to the coarse grid. The smoothing process uses the low order discretization scheme too, which implies that two discrete problems with slightly different fixed points are solved. The τ–extrapolation is a special case of the double discretization method where we use the Richardson extrapolation technique to change the discretization order of the coarse grid. The analytic solution needs to be smooth enough and the restrictions operator needs to be chosen carefully enough, for the τ–extrapolation to improve the regular multigrid method.

A specificity of the τ–extrapolation applied to problems with singular source points is that we use f_H instead of $\hat{I}_h^H f_h$ at the coarse grid correction step. f_H is the discretization of the right-hand side using the discrete Dirac delta functions that have a $2H = 4h$ support, while f_h is evaluated using the same kind of delta function but with a $2h$ support. This is easy to implement and saves an interpolation process per multigrid iteration.

4 Numerical results

4.1 The 1D Helmholtz operator

Let us compare the different behaviors of both elliptic operators introduced in Sec. 1 with a singular source point at the right-hand side. We solve at first the 1D problem [Wal99]:

$$\frac{d^2u}{dx^2}(x) - \alpha^2 u(x) = -2\alpha\delta(x - x_0), \quad x \in [-0.5, 0.5], \;\; \alpha \in \mathbb{R}_+^*; \tag{12}$$

$$x_0 \in [-0.5, 0.5]; \;\; u(-0.5) = e^{-\alpha|-0.5-x_0|} \;\text{ and }\; u(0.5) = e^{-\alpha|0.5-x_0|}.$$

The domain is divided in N equidistant intervals. Finite-differences and a classic stencil for the second order derivative are implemented in all of our computations. The computed solution is compared to the exact solution: $u_{ex}(x) = e^{-\alpha|x-x_0|}$, taking $x_0 = 0$ and $\alpha = 60$.

The number of operations represents the number of time the values at the nodes are updated but does not take into account the extrapolation and interpolation operations made in the multigrid algorithms in order to switch from one grid to another. The multigrid algorithm implemented is a classic V shaped algorithm with only two levels. We can see on Fig. 2 that the τ–extrapolation significantly improves the convergence order for this 1D problem with Dirac point load.

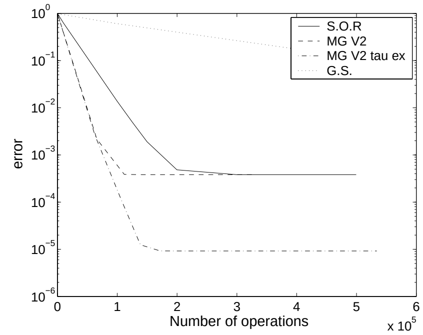

Fig. 1. *Error in L2-norm with respect to the number of operations with the 4 solvers S.O.R., Multigrid V2, Multigrid V2/ τ ex. and Gauss-Seidel. $N = 1000$, $x_0 = 0$ and we use the piec. cub. delta func.*

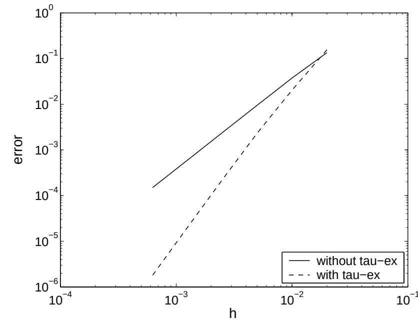

Fig. 2. *Error in L2-norm of the method for the multigrid algo. with or without the τ−extrapolation. and using the piec. cub. delta func. The order is improved from 2.0 to 3.4.*

Since the point loads in the IBM can be located anywhere in a cell, it is relevant to study the behavior of the error depending on the distance between the point load and the nodes of the mesh. In the following graph, the error relative to the exact solution is plotted as a function of d, the minimum distance between x_0 and the nodes of the mesh , from 0 to $\frac{h}{2}$:

$$d(x_0) = min_{i=1,..,N+1} \left| (-0.5 + (i-1)h) - x_0 \right| . \tag{13}$$

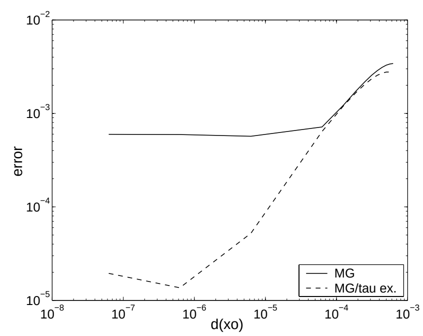

Fig. 3. *Error in L2-norm with respect to $d(x_0)$, the min. distance between x_0 and the nodes of the mesh, using the piec. cub. delta func. and the multigrid solver with or without the τ−extrapolation. $N = 800$, $x_0 = 0$.*

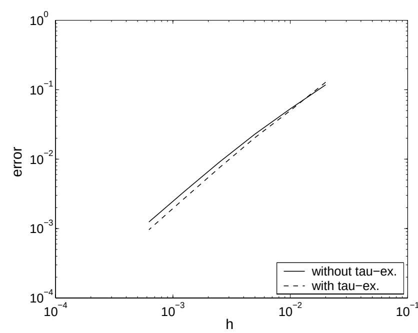

Fig. 4. *Error in L2-norm of the method using the multigrid algorithm with or without the τ−extrapolation and using the piec. cub. delta func. centered in the middle of a cells. The order is then 1.4*

We can see on Fig. 3 that the accuracy strongly depends on the distance $d(x_0)$. If we measure the convergence order of the method when $d(x_0) = \frac{h}{2}$, using the L2-norm, we get only 1.4 (Fig. 4).

4.2 The 2D Laplace operator

Let us study the following benchmark problem:

$$-\Delta u(x,y) = \delta(x,y,\Gamma), \quad (x,y) \in \Omega = [-1,1]^2; \tag{14}$$
$$\Gamma = \left\{(x,y) \in \Omega / x^2 + y^2 = r^2\right\}, \quad r < 1, \quad u_{|\partial\Omega} = u_{ex|\partial\Omega}$$
$$u_{ex}(x,y) = \begin{cases} 1 - \frac{1}{2} ln\left(\frac{1}{r}\sqrt{x^2+y^2}\right), & \text{if } x^2+y^2 > r^2; \\ 1, & \text{if } x^2+y^2 \le r^2. \end{cases}$$

In this case the source term is distributed along a circle centered at the origin and with radius $r < 1$, which makes this problem closer to those in the IBM. This time we need to use a discrete collection of M Dirac delta functions along the line Γ. M is usually a large number so that the discretization error of the delta functions along Γ is minimized:

$$\delta_h(x,y,\Gamma) = \frac{1}{M}\sum_{i=1}^{M} \delta_h\left(x - r\,cos\left(\frac{2(i-1)\pi}{M}\right)\right)\delta_h\left(y - r\,sin\left(\frac{2(i-1)\pi}{M}\right)\right) \tag{15}$$

The error between the computed and analytic solutions is measured along the x-axis in the L2-norm. We get, for the convergence order using the L2-norm, 2.0 without the τ–extrapolation and 2.8 with. Since the discrete Dirac delta functions are located along the circle, the distance between them and the nodes of the mesh varies between 0 and $\frac{h}{\sqrt{2}}$. The error is an average of the errors we would get with the delta functions centered at the nodes or at the mesh cell center.

5 Conclusion

We have shown that one can improve dramatically the accuracy of the IBM solvers by combining the τ–extrapolation technique with the piecewise cubic discrete Dirac delta function presented by Engquist and Tornberg [Eng04]. Our current experiments with fluid-structure interactions extends these preliminary results using the IBM on staggered grid meshes.

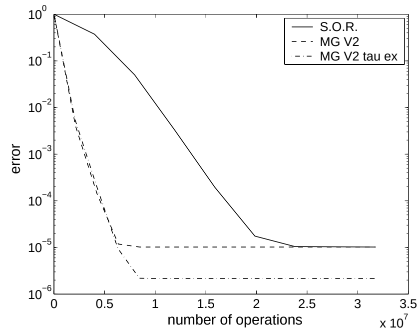

Fig. 5. *Error in L2-norm with respect to the number of operations with the 3 solvers S.O.R., Multigrid V2, Multigrid V2/ τ ex. $N = 200$, $r = 0.5$ and we use the piec. cub. delta func.*

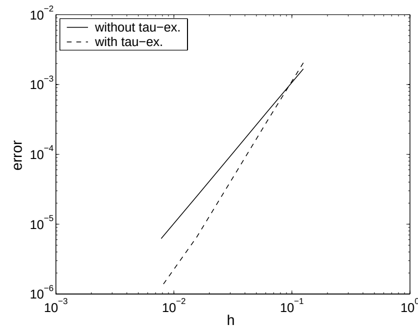

Fig. 6. *Error in L2-norm of the method for the multigrid algorithm with or without the τ–extrapolation and using the piece. cub. delta func. The order is improved from 2.0 to 2.8.*

References

[Rud04] Rüde, U., Köstler, H.: Accurate Techniques for Computing Singular Solutions of Elliptic Problems. Technical report, Friedrich-Alexander-Universität Erlangen-Nürnberg (2004)

[Pes02] Peskin, C.S.: The Immersed Boundary Method. Acta Numerica, 1–39 (2002)

[Eng04] Engquist, B., Tornberg A.-K.: Numerical Approximation of Singular Source Terms in Differential Equations. Jour. of Comp. Phys., **200**, 462–488 (2004)

[Har65] Harlow, F.H., Welch, E.: Numerical calculation of time-dependent viscous incompressible flow of fluids with free surface. Phys. Fluids, **8**, 2182–2189 (1965)

[Cho68] Chorin, A.J.: Numerical solution of the Navier–Stokes equations. Math. Comp., **22**, 745-762 (1968)

[Ber97] Bernert, K.: τ-Extrapolation—Theoretical Foundation, Numerical Experiment, and Application to Navier–Stokes Equations. SIAM Jour. on Sci. Comp., **18-2**, 460–478 (1997)

[Wal99] Waldén, J.: On the approximation of singular source terms in differential equations. Numer. Methods Partial Differential Equations, **15**, 503–520 (1999)

[Pes72] Peskin, C.S.: Flow patterns around heart valves: A numerical method. Jour. of comp. phys., **10**, 252–271 (1972)

[Sto97] Stockie, J.M.:Analysis and Computation of Immersed Boundaries, with Applications to Pulp Fibres.PHD Thesis, University of British Columbia (1997)

[Rom96] Roma, A.M.: A Multilevel Self Adaptive Version of the Immersed Boundary Method. PHD Thesis, New York University (1996)