IV Journeys in Multiphase Flows (JEM2015)
March 23-27, 2015 - Campinas, Brazil
Copyright © 2015 by ABCM
PaperID: JEM-2015-0067

HIGH-ORDER IMMERSED INTERFACE METHOD TO SOLVE THE PRESSURE POISSON EQUATION

Marilaine Colnago
Leandro Franco de Souza
Departamento de Matemática Aplicada e Estatística, ICMC, USP - 13566-590, São Carlos, SP
colnago@icmc.usp.br, lefraso@icmc.usp.br

Abstract. *Immersed interface methods are becoming increasingly popular in fluid mechanics. In particular, the Immersed Interface Method (IIM) figures among the most effective approaches, where traditionally the technique is used to simulate the flow behavior over complex bodies immersed on a Cartesian mesh. In this work, a numerical approach based on the IIM is proposed to solve the pressure equation in two-phase flow on a Cartesian grid, which is guaranteed to produce high order accuracy. The effectiveness of the proposed technique is attested by simulating the Poisson equation in a variety of irregular domains.*

Keywords: *Immersed Interface Method, pressure Poisson equation, computational fluid dynamics*

1. INTRODUCTION

The basic idea of the IIM is to discretize the fluid equations on a uniform Cartesian grid while still imposing the jumps of the function into the finite difference (FD) formulas in order to produce a solution. Its correction terms require jump conditions to the first and second derivatives to reach a second-order accuracy. In (Xu and Wang, 2005), the authors have suggested the use of IIM to simulate the interaction of a fluid with moving boundaries. Li and Lai (Li and Lai, 2001) and Lee and Leveque (Lee and LeVeque, 2003) employ the Immersed Interface Method to model a few number of two-dimensional flow cases, achieving second-order spatial accuracy.

In (Zhong, 2006), the author presents a new high-order Immersed Interface Method to solve multi-phase flow. Although the author claims that the method can be applied to simulate general multi-phase flows, in his work the method is only used to solved elliptic equations on irregular interfaces. Linnick and Fasel (Linnick and Fasel, 2005) propose a high-order Immersed Interface Method for simulating unsteady incompressible flow in an irregular domain, more specifically, to compute incompressible flow over a cylinder. They compute the jump conditions obtaining a fourth-order compact scheme.

The goal of this work is to obtain high-order immersed interface method to solve the pressure Poisson equation for two-phase flow.

2. METHODOLOGY

In the following, we present the equations used to model a two-phase flow and the proposed methodology to solve the pressure equation using the IIM.

2.1 FLUID EQUATIONS

Many two-phase fluid flows can be described by using the incompressible Navier-Stokes equations, which can be written as follows:

$$\rho\left[\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}.\bigtriangledown \mathbf{u}\right] = -\bigtriangledown p + \bigtriangledown.[\mu(\bigtriangledown \mathbf{u} + \bigtriangledown \mathbf{u}^T] + \rho \mathbf{g} + \mathbf{f}, \tag{1}$$

$$\frac{\partial \rho}{\partial t} + \bigtriangledown.(\rho \mathbf{u}) = 0, \tag{2}$$

where ρ and μ are the fluid density and the fluid viscosity, respectively, p is the pressure term, $\mathbf{u}$ is the fluid velocity field, $\mathbf{f}$ is the field of external force and $\mathbf{g}$ represents the gravity.

Once we have a set of equations to simulate an incompressible two-phase flow, let ρ be a constant assuming distinct values for each one of the phases. The superscripts $+$ and $-$ are used to indicate both phases (see Fig. 1). Expliciting the density and the viscosity in the above-mentioned phases, we have: $\rho^+, \quad \rho^-, \quad \mu^+, \quad \mu^-$. So, introducing the jump conditions across the interface Γ is needed, in mathematical words,

$$[\rho]_\Gamma = \rho^+ - \rho^-, [\mu]_\Gamma = \mu^+ - \mu^-. \tag{3}$$

In the absence of mass transfer between the two phases, the velocity field is continuous along the interface, that is,

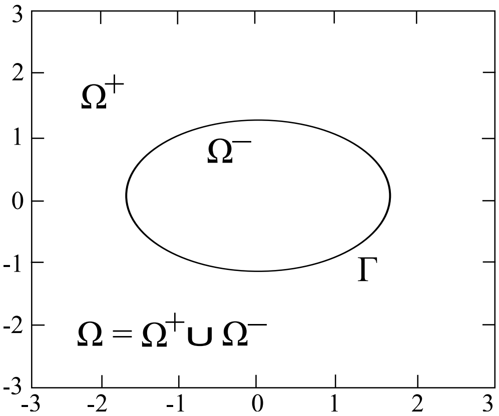

Figure 1. A rectangular domain $\Omega = \Omega^+ \cup \Omega^-$ with an irregular immersed interface Γ.

$[u]_\Gamma = 0$. In contrast, we have a surface tension force that induces to a discontinuity in the normal stresses at the fluid-fluid interface which produces a pressure jump that can be expressed as:

$$[p]_\Gamma = \sigma\kappa + 2[\mu]_\Gamma \mathbf{n}^T . \bigtriangledown \mathbf{u}.\mathbf{n}, \tag{4}$$

where σ is the surface tension coefficient, κ is the curvature of the phase-interface, and $\mathbf{n}$ is the phase-interface normal.

2.2 NUMERICAL ALGORITHM

We firstly start by considering the one-dimensional problem

$$\bigtriangledown.(\beta \bigtriangledown p) = \bigtriangledown.\mathbf{u}, \tag{5}$$

on the interval $[0, 1]$. Function $\beta = 1/\rho$ is allowed to be discontinuous at point x_α. Once β assumes constants values in both sides of the interface, we may write:

$$\begin{array}{rcl} \bigtriangledown.\mathbf{u} & = & \bigtriangledown.(\beta \bigtriangledown p) \\ \bigtriangledown.\mathbf{u} & = & \beta_x p_x + \beta p_{xx} \\ \bigtriangledown.\mathbf{u} & = & \beta p_{xx}. \end{array}$$

From Eq. (6), we are able to solve the originated Poisson equation.

Figure 2 shows a function $f(x)$ with discontinuity at $x = x_\alpha$.

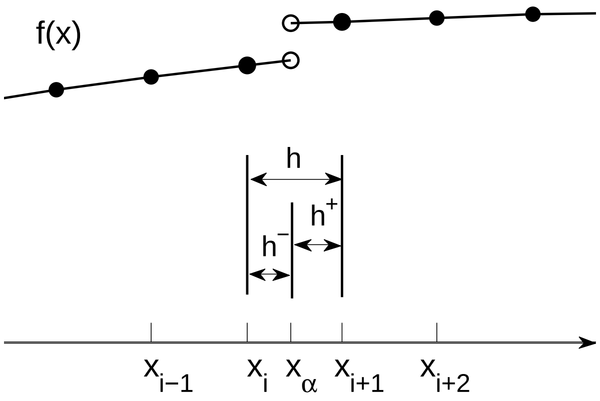

Figure 2. A function $f(x)$ discontinuous at $x = x_\alpha$.

If $x_i < x_\alpha$, the expansion of Taylor series can not be applied to approximate $f(x_{i+1})$, unless a correction term J_α be added:

$$f(x_{i+1}) = f(x_i) + f'(x_i)h + f''(x_i)\frac{h^2}{2!} + \ldots + J_\alpha, \tag{6}$$

where

$$J_\alpha = [f]_\alpha + [f']_\alpha h^+ + \frac{1}{2!}[f'']_\alpha (h^+)^2 + \ldots, \tag{7}$$

$h = x_{i+1} - x_i$, $h^+ = x_{i+1} - x_\alpha$ and $[f]_\alpha$ represents the jump in the function value at $x = x_\alpha$, that is:

$$[f]_\alpha = \lim_{x\to x_\alpha^+} f(x) - \lim_{x\to x_\alpha^-} f(x), \tag{8}$$

$[f']_\alpha$ the jump in the value of the first derivative of the function at $x = x_\alpha$, and so on.

A finite difference scheme to solve the Poisson equation without discontinuity can be expressed by:

$$L_{i-1}^2 f_{i-1}^{(2)} + L_i^2 f_i^{(2)} + L_{i+1}^2 f_{i+1}^{(2)} = R_{i-1}^2 f_{i-1} + R_i^2 f_i + R_{i+1}^2 f_{i+1}, \tag{9}$$

where the term L_i refers to the finite-difference coefficients appearing on the left side of the FD scheme and R_i the coefficients on the right side. So, considering the following scheme

$$\frac{1}{12}(f_{i-1}^{(2)} + 10 f_i^{(2)} + f_{i+1}^{(2)}) = \frac{1}{h^2}(f_{i-1} - 2f_i + f_{i+1}), \tag{10}$$

we have $L_{i-1} = L_{i+1} = 1/12$, $L_i = 10/12$, $R_{i-1} = R_{i+1} = 1/h^2$, and $R_i = -2/h^2$ (more details about the compact fouth-order FD scheme, we refer (Lele, 1992)).

If the correction is needed, we can change Eq. (10) adding the term $L_I^2 J_{\alpha 2} - R_I^2 J_{\alpha 0}$ in its right side:

$$L_{i-1}^2 f_{i-1}^{(2)} + L_i^2 f_i^{(2)} + L_{i+1}^2 f_{i+1}^{(2)} = R_{i-1}^2 f_{i-1} + R_i^2 f_i + R_{i+1}^2 f_{i+1} + (L_I^2 J_{\alpha 2} - R_I^2 J_{\alpha 0}), \tag{11}$$

where $J_{\alpha 2}$ and $J_{\alpha 0}$ represent the correction terms to the second derivative and the zero order derivative. Both jump correction terms are truncated by taking only enough terms to ensure order $O(h^4)$.

So, if the jump singularity occurs at $x_\alpha : x_i < x_\alpha < x_{i+1}$, $I = i + 1$, then the correction is required for the point x_i so that the jump correct terms are computed as follows:

$$\begin{aligned} J_{\alpha 0} &= [f^{(0)}]_\alpha + [f^{(1)}]_\alpha h^+ + \frac{1}{2!}[f^{(2)}]_\alpha (h^+)^2 + \frac{1}{3!}[f^{(3)}]_\alpha (h^+)^3 + \\ &\quad \frac{1}{4!}[f^{(4)}]_\alpha (h^+)^4 + \frac{1}{5!}[f^{(5)}]_\alpha (h^+)^5, \qquad (12) \\ J_{\alpha 2} &= [f^{(2)}]_\alpha + [f^{(3)}]_\alpha h^+ + \frac{1}{2!}[f^{(4)}]_\alpha (h^+)^2 + \frac{1}{3!}[f^{(5)}]_\alpha (h^+)^3, \qquad (13) \end{aligned}$$

for the zero and second derivative, respectively, and $h^+ = x_{i+1} - x_\alpha$.

If the jump singularity occurs for $x_{i-1} < x_\alpha < x_i$, $I = i - 1$, and

$$\begin{aligned} J_{\alpha 0} &= -[f^{(0)}]_\alpha + [f^{(1)}]_\alpha h^- - \frac{1}{2!}[f^{(2)}]_\alpha (h^-)^2 + \frac{1}{3!}[f^{(3)}]_\alpha (h^-)^3 \\ &\quad -\frac{1}{4!}[f^{(4)}]_\alpha (h^-)^4 + \frac{1}{5!}[f^{(5)}]_\alpha (h^-)^5, \qquad (14) \\ J_{\alpha 2} &= -[f^{(2)}]_\alpha + [f^{(3)}]_\alpha h^- - \frac{1}{2!}[f^{(4)}]_\alpha (h^-)^2 + \frac{1}{3!}[f^{(5)}]_\alpha (h^-)^3, \qquad (15) \end{aligned}$$

where $h^- = x_\alpha - x_{i-1}$.

In order to hold fourth order of accuracy, each one of the one-sided stencils employed to calculate the jumps should contain six points. So, additional terms $[f^{(n)}]_\alpha$ can be computed from

$$[f^{(n)}]_\alpha = f_{F.D.^+}^{(n)} - f_{F.D.^-}^{(n)}, \tag{16}$$

where

$$\begin{aligned} f_{F.D.^+}^{(n)} &= c_{n_\alpha^+} f_\alpha^+ + c_{n_{i+2}} f_{i+2} + c_{n_{i+3}} f_{i+3} + c_{n_{i+4}} f_{i+4} + c_{n_{i+5}} f_{i+5} + c_{n_{i+6}} f_{i+6} \qquad (17) \\ f_{F.D.^-}^{(n)} &= c_{n_\alpha^-} f_\alpha^- + c_{n_{i-1}} f_{i-1} + c_{n_{i-2}} f_{i-2} + c_{n_{i-3}} f_{i-3} + c_{n_{i-4}} f_{i-4} + c_{n_{i-5}} f_{i-5}, \qquad (18) \end{aligned}$$

if the jump singularity occurs for $x_i < x_\alpha < x_{i+1}$. The superscripts $+$ and $-$ indicate right and left limits, respectively. In order to avoid issues relate to numerical instability, points x_i and x_{i+1} are not used into the scheme.

If the jump singularity occurs for $x_{i-1} < x_\alpha < x_i$, the points x_i and x_{i-1} are not used and the derivative terms $[f^{(n)}]_\alpha$ can be obtained as:

$$[f^{(n)}]_\alpha = f_{F.D.^+}^{(n)} - f_{F.D.^-}^{(n)}, \tag{19}$$

where

$$\begin{aligned} f_{F.D.^+}^{(n)} &= c_{n_\alpha^+} f_\alpha^+ + c_{n_{i+1}} f_{i+1} + c_{n_{i+2}} f_{i+2} + c_{n_{i+3}} f_{i+3} + c_{n_{i+4}} f_{i+4} + c_{n_{i+5}} f_{i+5} \qquad (20) \\ f_{F.D.^-}^{(n)} &= c_{n_\alpha^-} f_\alpha^- + c_{n_{i-2}} f_{i-2} + c_{n_{i-3}} f_{i-3} + c_{n_{i-4}} f_{i-4} + c_{n_{i-5}} f_{i-5} + c_{n_{i-6}} f_{i-6}, \qquad (21) \end{aligned}$$

In order to obtain a numerical approximation to the *n*th derivative of f, the coefficients c_{n_i} are adopted as a linear combination of the f_i, for instance:

$$f_\alpha^+ = \lim_{x\to\alpha^+} f(x), \quad \text{and} \quad f_\alpha^- = \lim_{x\to\alpha^-} f(x). \tag{22}$$

Aiming at finding the constants $c's$, we employ the following explicit difference finite scheme to the *n*th derivative:

$$f_\alpha^{(n)} = c_\alpha f_\alpha + c_i f_i + c_{i+1} f_{i+1} + c_{i+2} f_{i+2} + c_{i+3} f_{i+3} + c_{i+4} f_{i+4}. \tag{23}$$

So:

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & h_i & h_{i+1} & h_{i+2} & h_{i+3} & h_{i+4} \\ 0 & h_i^2 & h_{i+1}^2 & h_{i+2}^2 & h_{i+3}^2 & h_{i+4}^2 \\ 0 & h_i^3 & h_{i+1}^3 & h_{i+2}^3 & h_{i+3}^3 & h_{i+4}^3 \\ 0 & h_i^4 & h_{i+1}^4 & h_{i+2}^4 & h_{i+3}^4 & h_{i+4}^4 \\ 0 & h_i^5 & h_{i+1}^5 & h_{i+2}^5 & h_{i+3}^5 & h_{i+4}^5 \end{bmatrix} \begin{bmatrix} c_\alpha \\ c_i \\ c_{i+1} \\ c_{i+2} \\ c_{i+3} \\ c_{i+4} \end{bmatrix} = \begin{bmatrix} 1\delta_{n0} \\ 1\delta_{n1} \\ 2!\delta_{n2} \\ 3!\delta_{n3} \\ 4!\delta_{n4} \\ 5!\delta_{n5} \end{bmatrix} \tag{24}$$

where $h_i = x_i - x_\alpha$ and δ_{ij} is the Kronecker delta function

$$\delta_{ij} = \begin{cases} 1, & \text{if } i = j, \\ 0, & \text{if } i \neq j. \end{cases} \tag{25}$$

To illustrate the above-described scheme, we explain below how to find $c's$ to obtain $f_{F.D.+}^{(1)}$ when $x_i < x_\alpha < x_{i+1}$. The explicit FD is:

$$f_{F.D.+}^{(1)} = c_{1_\alpha^+} f_\alpha^+ + c_{1_{i+2}} f_{i+2} + c_{1_{i+3}} f_{i+3} + c_{1_{i+4}} f_{i+4} + c_{1_{i+5}} f_{i+5} + c_{1_{i+6}} f_{i+6}. \tag{26}$$

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & h_{i+2} & h_{i+3} & h_{i+4} & h_{i+5} & h_{i+6} \\ 0 & h_{i+2}^2 & h_{i+3}^2 & h_{i+4}^2 & h_{i+5}^2 & h_{i+6}^2 \\ 0 & h_{i+2}^3 & h_{i+3}^3 & h_{i+4}^3 & h_{i+5}^3 & h_{i+6}^3 \\ 0 & h_{i+2}^4 & h_{i+3}^4 & h_{i+4}^4 & h_{i+5}^4 & h_{i+6}^4 \\ 0 & h_{i+2}^5 & h_{i+3}^5 & h_{i+4}^5 & h_{i+5}^5 & h_{i+6}^5 \end{bmatrix} \begin{bmatrix} c_{1_\alpha} \\ c_{1_i+2} \\ c_{1_i+3} \\ c_{1_i+4} \\ c_{1_i+5} \\ c_{1_i+6} \end{bmatrix} = \begin{bmatrix} 1\delta_{10} \\ 1\delta_{11} \\ 2!\delta_{12} \\ 3!\delta_{13} \\ 4!\delta_{14} \\ 5!\delta_{15} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{27}$$

In summary, if the jump singularity occurs at point $x_\alpha : x_i < x_\alpha < x_{i+1}$, we have:

1. At the point x_{i-1}, there is no correction term. So:

$$L_{i-1}^2 f_{i-1}^{(2)} + L_i^2 f_i^{(2)} + L_{i+1}^2 f_{i+1}^{(2)} = R_{i-1}^2 f_{i-1} + R_i^2 f_i + R_{i+1}^2 f_{i+1}. \tag{28}$$

2. At the point x_i:

$$L_{i-1}^2 f_{i-1}^{(2)} + L_i^2 f_i^{(2)} + L_{i+1}^2 f_{i+1}^{(2)} = R_{i-1}^2 f_{i-1} + R_i^2 f_i + R_{i+1}^2 f_{i+1} + (L_I^2 J_{\alpha 2} - R_I^2 J_{\alpha 0}). \tag{29}$$

And, if the jump singularity occurs at $x_\alpha : x_{i-1} < x_\alpha < x_i$:

1. At the point x_{i+1}, the stencil does not intersect the discontinuity points, then it is no needed to compute the correction term, that is:

$$L_{i-1}^2 f_{i-1}^{(2)} + L_i^2 f_i^{(2)} + L_{i+1}^2 f_{i+1}^{(2)} = R_{i-1}^2 f_{i-1} + R_i^2 f_i + R_{i+1}^2 f_{i+1}. \tag{30}$$

2. At the point x_i:

$$L_{i-1}^2 f_{i-1}^{(2)} + L_i^2 f_i^{(2)} + L_{i+1}^2 f_{i+1}^{(2)} = R_{i-1}^2 f_{i-1} + R_i^2 f_i + R_{i+1}^2 f_{i+1} + (L_I^2 J_{\alpha 2} - R_I^2 J_{\alpha 0}). \tag{31}$$

From Eq. (28 - 31), we obtain a system of linear equations that can be solved using a direct or iterative algorithm.

Similar to 1D case, we can discretize the 2D case by using fourth-order compact finite-difference schemes. If there is no discontinuity, the scheme results to solve:

$$\bigtriangledown f(x, y) = \rho(x, y) \tag{32}$$

can be expressed by:

$$\begin{aligned} L_{xx} f_{xx_{ij}} &= R_{xx_i} f_{ij} && (33) \\ L_{yy} f_{yy_{ij}} &= R_{yy_j} f_{ij}. && (34) \end{aligned}$$

where L_{xx}, L_{yy}, R_{xx} and R_{yy} denote the coefficients in the FD scheme and f_{xx} and f_{yy} represent the numerical approximations to the second partial derivatives in the x and y directions.

Using the Eq. (32-34), the 2D compact stencils at point (i, j) can be obtained as:

$$\begin{aligned} f_{xx_{ij}} + f_{yy_{ij}} &= \rho_{ij} && (35) \\ L_{xx_i} L_{yy_j} (f_{xx_{ij}} + f_{yy_{ij}}) &= L_{xx_i} L_{yy_j} \rho_{ij} \\ L_{yy_j}(L_{xx_i} f_{xx_{ij}}) + L_{xx_i}(L_{yy_j} f_{yy_{ij}}) &= L_{xx_i} L_{yy_j} \rho_{ij} \\ L_{yy_j}(R_{xx_i} f_{ij}) + L_{xx_i}(R_{yy_j} f_{ij}) &= L_{xx_i} L_{yy_j} \rho_{ij} \\ (L_{yy_j} R_{xx_i} + L_{xx_i} R_{yy_j}) f_{ij} &= L_{xx_i} L_{yy_j} \rho_{ij} \\ L_{ij} f_{ij} &= R_{ij} \rho_{ij} && (36) \end{aligned}$$

where $L_{ij} = L_{yy_j} R_{xx_i} + L_{xx_i} R_{yy_j}$ and $R_{ij} = L_{xx_i} L_{yy_j}$, resulting in a nine-point 2D stencil.

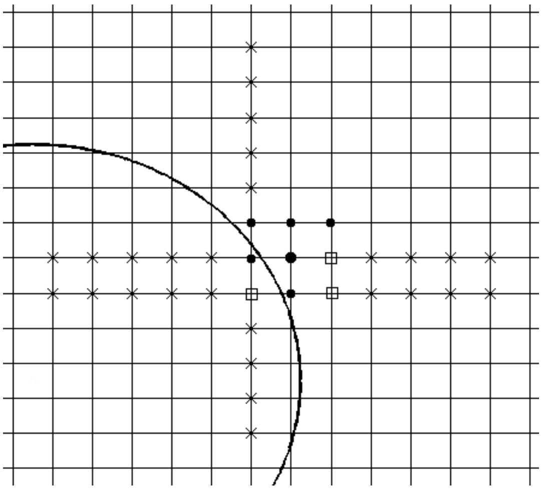

Figure 3. The nine-point stencil (●) given in Eq. (36) and the jump corrections terms (×). The points where (●) and (×) overlap are represented by (□).

If corrections are required, we have:

$$\begin{aligned} L_{xx} f_{xx_{ij}} &= R_{xx_i} f_{ij} - J_{\alpha x I} && (37) \\ L_{yy} f_{yy_{ij}} &= R_{yy_j} f_{ij} - J_{\alpha y K}, && (38) \end{aligned}$$

where

$$\begin{aligned} J_{\alpha x I} &= -(L_{xx_I} J_{\alpha 2x} - R_{xx_I} J_{\alpha 0x}) && (39) \\ J_{\alpha y K} &= -(L_{yy_K} J_{\alpha 2y} - R_{yy_K} J_{\alpha 0y}), && (40) \end{aligned}$$

and

$$I = \begin{cases} i-1, & x_{i-1} < x_\alpha < x_i, \\ i+1, & x_i < x_\alpha < x_{i+1}, \end{cases} \qquad (41)$$

and similary for K. The jump corrections $J_{\alpha 2x}$, $J_{\alpha 0x}$, $J_{\alpha 2y}$ and $J_{\alpha 0y}$ are defined as in 1D case. Finally, the resulting of 2D discretizations for Eq. (32) with jump corrections becomes:

$$(L_{yy_j} R_{xx_i} + L_{xx_i} R_{yy_j}) f_{ij} = L_{xx_i} L_{yy_j} \rho_{ij} + (L_{yy_j} J_{\alpha x I_j} + L_{xx_i} J_{\alpha y K_i}). \qquad (42)$$

Fig. 3 portrays an stencil where jump corrections are needed.

3. RESULTS AND NUMERICAL SIMULATIONS

Example 1. In order to demonstrate the robustness of the IIM to simulate two-phase flows, we consider the following equation:

$$\beta_x p_x + \beta p_{xx} = u, \qquad (43)$$

where

$$p(x) = \begin{cases} \sin(2x), & \text{if } x \le x_\alpha \\ \cos(2x), & \text{if } x > x_\alpha, \end{cases} \tag{44}$$

$$\beta(x) = \begin{cases} 2, & \text{if } x \le x_\alpha \\ 1, & \text{if } x > x_\alpha, \end{cases} \tag{45}$$

$$u(x) = \beta_x p_x + \beta p_{xx}. \tag{46}$$

and the jump occurs in $x_\alpha = 0.755882$.

Figure 4 shows a comparison between the original and our numerical solution while Fig. 5 depicts the obtained the error. Both Figures show that the presence of the discontinuity does not affect the error, even for the points near to the interface. In this case, the maximum error is located in the region without discontinuities.

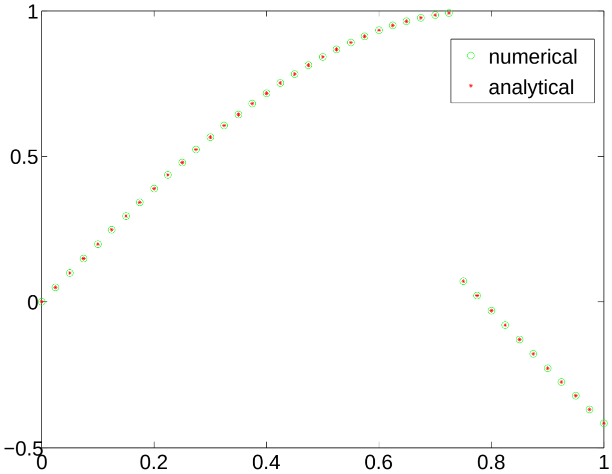

Figure 4. Comparison between analytical and numerical solutions (Example 1).

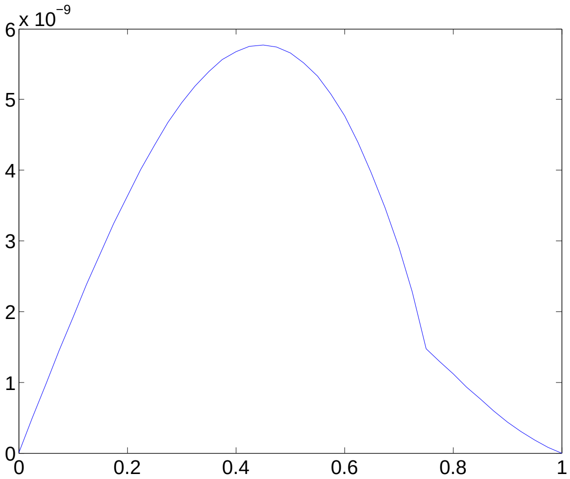

Figure 5. Error computed when applying Immersed Interface Method (Example 1).

Figure 6 shows the error quantities obtained from the max-norm, wich is computed as:

$$\| E \|_\infty = \max_{1 \le j \le m} | E_j | = \max_{1 \le j \le m} | P_j - p(x_j) |, \tag{47}$$

where p and P are the analytical and numerical solutions, respectively, plotted against h on a log-log scale, where fourth-order accuracy can be observed.

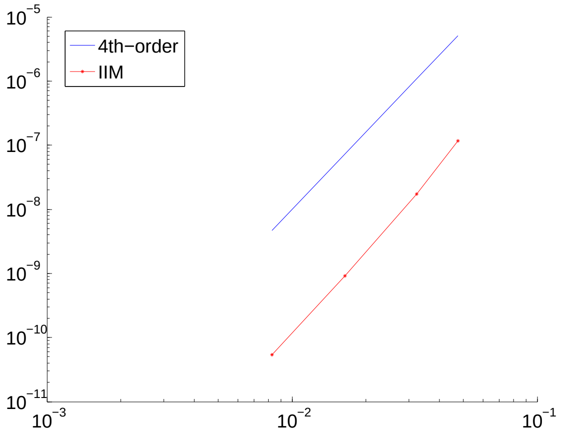

Figure 6. Error values plotted against h on a log-log scale (Example 1).

Table 1 summarizes the errors obtained from max-norm, 1-norm and 2-norm (letting h varying) and the order calculated from

$$p = \frac{\log\left(\frac{error(h_1)}{error(h_2)}\right)}{\log\left(\frac{h_1}{h_2}\right)}. \tag{48}$$

Table 1. Error and order of convergence (Example 1).

n	$\|E_n\|_\infty$	Order	$\|E_n\|_1$	Order	$\|E_n\|_2$	Order
20	1.18951e-07	-	7.31575e-08	-	8.21768e-08	-
40	5.55442e-09	4.2	2.19903e-09	4.3	2.72315e-09	4.3
80	3.12079e-10	4.1	1.31890e-10	3.9	1.67452e-10	4.0
160	1.93619e-11	4.0	1.08269e-11	3.7	1.20385e-11	3.8

It is clear from Table 1 that a fourth-order accuracy was reached, as expected.

Example 2. Now, we are interested in solving the Poisson equation on form

$$\beta_x p_x + \beta p_{xx} = u, \tag{49}$$

with two discontinuous points, $x_{\alpha 1} = 0.51378$, $x_{\alpha 2} = 0.723141$, and

$$p(x) = \begin{cases} \sin(2x), & x < x_{\alpha 1}, \\ \cos(2x), & x_{\alpha 1} \geq x \geq x_{\alpha 2}, \\ \sin(2x), & \text{otherwise}, \end{cases} \tag{50}$$

$$\beta(x) = \begin{cases} 1/1.2047, & x < x_{\alpha 1}, \\ 1/997.78, & x_{\alpha 1} \geq x \geq x_{\alpha 2}, \\ 1/1.2047, & \text{otherwise}, \end{cases} \tag{51}$$

Figure 7 illustrates a comparison between the numerical and analytical solutions while Fig. 8 depicts the max-norm error.

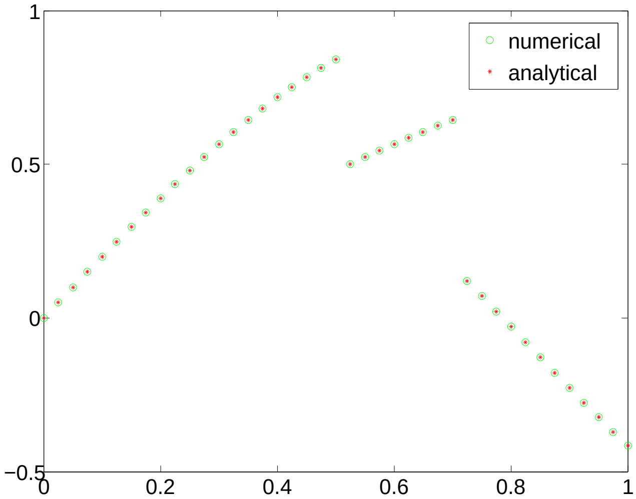

Figure 7. Comparison between the analytical and numerical solutions (Example 2).

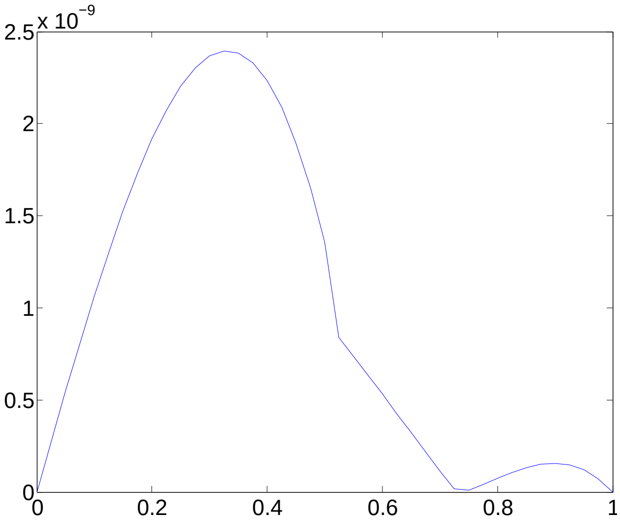

Figure 8. Error computed when applying the Immersed Interface Method (Example 2).

Table 1 summarizes the error obtained when applying the IIM for the Example 2.

Table 2. Error and order of convergence (Example 2).

n	$\|E_n\|_\infty$	Order	$\|E_n\|_1$	Order	$\|E_n\|_2$	Order
20	3.59957e-07	-	3.64580e-08	-	7.65209e-08	-
40	7.11242e-09	4.8	1.17188e-09	4.6	1.97306e-09	4.8
80	3.94013e-10	4.2	1.60084e-10	3.4	2.13229e-10	3.7
160	2.15280e-11	4.2	1.15359e-11	3.8	1.32448e-11	4.0

Example 3. In the next test case, we solve the following 2D Poisson equation:

$$(\beta p_x)_x + (\beta p_y)_y = -u(x, y) \tag{52}$$

defined in a square $[-1,1] \times [-1,1]$ with a circular interface $x^2 + y^2 = 1/4$. As provided by (LeVeque and Li, 1997), the analytical solution is

$$p(x) = \begin{cases} x^2 + y^2, & r \leq 1/2, \\ \frac{1}{4}\left(1 - \frac{1}{8b} - \frac{1}{b}\right) + \left(\frac{(x^2+y^2)^2}{2} + x^2 + y^2\right)/b, & \text{otherwise} , \end{cases} \tag{53}$$

with

$$\beta(x) = \begin{cases} 2, & r \leq 1/2, \\ b, & \text{otherwise} . \end{cases} \tag{54}$$

Let $b = 10$ be a constant and $p(x,y)$ be a continuous function throughout the domain. The experiment was performed assuming a five-points FD scheme and, as shown in Table 3, second order of accuracy has been achieved.

Table 3. Error and convergence order for the IIM (Example 3).

n	$\|E_n\|_\infty$	Order	$\|E_n\|_1$	Order	$\|E_n\|_2$	Order
20	7.89759e-05	-	3.15031e-05	-	4.09409e-05	-
40	2.45938e-05	1.9	9.70220e-06	2.0	1.26339e-05	2.0
80	6.91275e-06	2.0	2.71758e-07	2.0	3.54093e-06	2.0
160	1.83601e-07	2.0	7.21229e-08	2.0	9.39694e-07	2.0

The obtained solution taking a 40×40 mesh is plotted in Fig. 9.

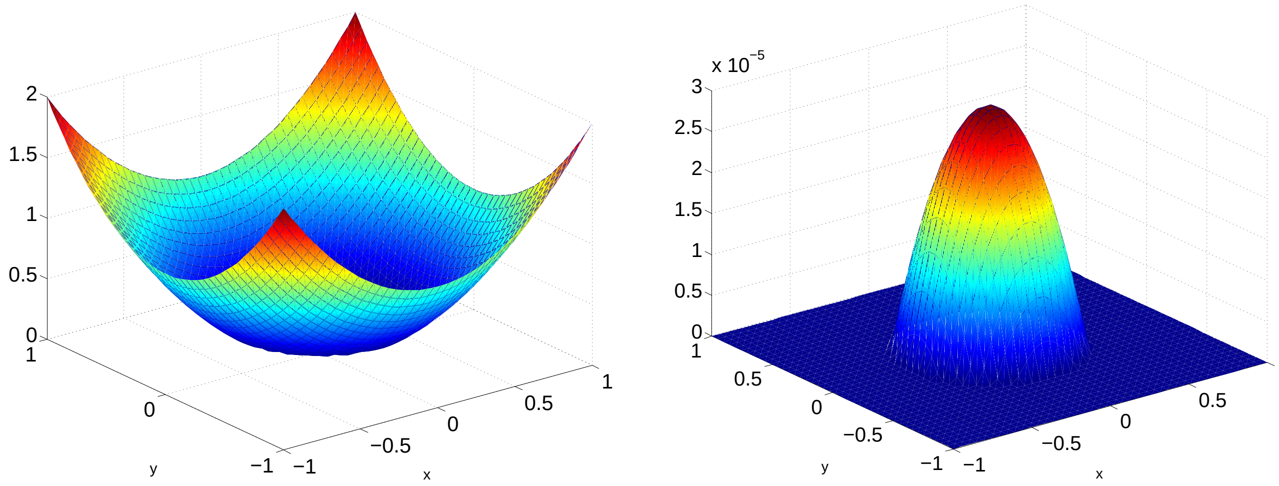

Figure 9. Computational solution (left) and the error (right) obtained from the Immersed Interface Method (Example 3).

4. CONCLUSION

In this work a high-order Immersed Interface Method to solve the pressure Poisson equation in two-phase flows has been presented. As shown in Section 3, for 1D case the fourth-order accuracy is reached. Moreover, a second-order accuracy is achieved using a five-points FD scheme to 2D case. Our experiments show that the convergence order is not affected by the presence of discontinuities in the simulating domain. As future work, we aim at adapting this scheme to simulate incompressible two-phase flows. Furthermore, we are also performing a variety of experiments taking a a nine-points scheme in order to reach fourth-order accuracy for 2D case.

5. ACKNOWLEDGMENTS

The authors would like to thank CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) for the financial support.

6. REFERENCES

Desjardins, O. and Moureau, V., 2009. "Methods for multiphase flows with high density ratio." *Chemical Engineering Science.*, Vol. 64, pp. 2941–2950.

Lee, L. and LeVeque, R., 2003. "An immersed interface method for incompressible Navier-Stokes equations." *Journal of Computational Physics*, Vol. 25, pp. 832–856.

Lele, S.K., 1992. "Compact finite difference schemes wit spectral-like resolution." *Journal of Computational Physics*, Vol. 103, pp. 16–42.

LeVeque, R. and Li, Z., 1997. "The immersed interface method for elliptic equations with discontinuous coefficients and singular sources." *SIAM J. Numerical Analysis*, Vol. 31, No. 4, pp. 1019–1044.

Li, Z. and Lai, M., 2001. "The Immersed Interface Method for the Navier–Stokes Equations with Singular Forces." *Journal of Computational Physics*, pp. 822–842.

Linnick, M.N. and Fasel, H.F., 2005. "A high-order immersed interface method for simulating unsteady incompressible flows on irregular domains." *Journal of Computational Physics*, Vol. 204, pp. 157–192.

Peskin, C.S., 1972. "Heart Valves: A Numerical Method." *Journal of Computational Physics*, Vol. 10, pp. 252–271.

Xu, S. and Wang, Z.J., 2005. "An immersed interface method for simulating the interaction of a fluid with moving boundaries." *Journal of Computational Physics*, Vol. 216, pp. 454–493.

Zhong, X., 2006. "A New High-order Immersed Interface Method for Multi-phase Flow." *AIAA Paper 2006-1294*.

7. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.