

Resolução numérica das equações de Navier-Stokes usando o método das interfaces imersas com correção do salto implícita em malha deslocada

Gabriela A. dos Reis, **Italo V. M. Tasso,**
Leandro F. de Souza, **José A. Cuminato.**

Depto de Matemática Aplicada e Estatística, ICMC, USP,
13566-590, São Carlos, SP

E-mail: gareis@icmc.usp.br, italo@tasso.com.br, lefraso@icmc.usp.br, jacumina@icmc.usp.br.

Resumo: *O método das interfaces imersas (MII) tem sido utilizado para simular problemas de escoamento de fluidos em geometrias complexas. Uma das dificuldades do MII é calcular os saltos nos valores da função e em suas derivadas, pois nem sempre esses valores são conhecidos analiticamente. Em [4], foi desenvolvido um MII que calcula numericamente os saltos, por meio de um processo iterativo explícito que utiliza os pontos vizinhos da descontinuidade. Seguindo essa ideia, foi desenvolvida em [5] uma versão implícita não-iterativa desse método para resolver uma equação de Poisson. O presente trabalho apresenta a resolução das equações de Navier-Stokes, pelo método da projeção, usando o MII desenvolvido em [5] em variáveis primitivas e em uma malha deslocada.*

Palavras-chave: *Navier-Stokes, malha deslocada, interface imersa, métodos de alta ordem.*

1 Introdução

O método das interfaces imersas (MII) tem sido empregado em simulações de escoamento de fluidos em geometrias altamente complexas, que são frequentemente encontradas em muitos campos da ciência e engenharia. Esse método permite a imposição de uma grande variedade de condições de contorno [8], incluindo superfícies livres, interfaces fluido-fluido, fluido-sólido, etc. A principal vantagem da técnica de interface imersa está na alta ordem de aproximação na modelagem de geometrias complexas, sem a necessidade de transformação de coordenadas, necessitando simplesmente de uma malha cartesiana uniforme.

O MII-analítico (MII-A) foi proposto por [3] e então permitia obter segunda ordem de precisão. Com o MII-implícito (MII-E) [4], quarta ordem foi obtida na simulação das equações de Navier-Stokes utilizando a formulação corrente-vorticidade. Os valores dos saltos no valor da função e em suas derivadas, necessários para o MII, são calculados numericamente, por meio de um processo iterativo explícito, ao invés de analiticamente como na versão original.

O presente trabalho apresenta o MII-implícito (MII-I), construído a partir do MII-E, com algumas modificações. O MII-E tem um inconveniente que é a utilização de processos iterativos para a estimativa dos valores dos saltos na resolução da equação de Poisson. O sistema linear tem que ser resolvido várias vezes. A cada iteração, o valor do salto é atualizado até convergir. Isso gera um processo caro computacionalmente. Portanto, é proposto o MII-I, uma versão do MII-E que trata as condições de salto de forma implícita, diretamente como parte da solução do sistema linear. Isso permite resolver o sistema linear uma única vez, proporcionando um ganho significativo em tempo de processamento. Além disso, o presente trabalho resolve as equações de Navier-Stokes em uma malha deslocada com variáveis primitivas, ao contrário do MII-E que usava a formulação corrente-vorticidade com malha colocalizada.

O método das soluções manufaturadas é utilizado nos testes de análise de convergência.

2 Equações governantes

A dinâmica de um escoamento Newtoniano incompressível descrito pelas equações de Navier-Stokes é dada por

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{g}, \quad (1)$$

$$\nabla \cdot \mathbf{u} = 0, \quad (2)$$

onde ν é viscosidade cinemática, $\mathbf{u} = (u, v)$ o vetor velocidade, p a pressão sobre densidade constante e \mathbf{g} o campo gravitacional. No presente trabalho, as condições de contorno adotadas são do tipo Dirichlet para a velocidade e Neumann homogênea para a pressão. O método de Euler explícito é usado para avançar (1) no tempo. O sistema de EDPs (1) e (2) é resolvido utilizando a versão incremental do método da projeção [1].

3 Método numérico

A Figura 1 ilustra o domínio que será utilizado para simular a solução das equações (1) e (2). A malha é gerada de forma que o contorno rígido seja representado por uma interface imersa. A solução em Ω é de interesse, fora disso é considerada nula. É importante notar que a interface não coincide com pontos de malha.

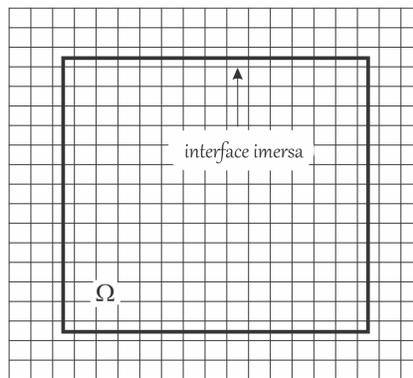


Figura 1: Ilustração do domínio de interesse nesse trabalho. O contorno rígido é tratado como uma interface imersa, e esse não coincide com pontos de malha.

3.1 Método das interfaces imersas

O MII-I, apresentado neste trabalho, é inspirado nas ideias presentes em [4]. Os autores desenvolveram o MII-E para resolver as equações de Navier-Stokes utilizando a formulação corrente-vorticidade. No MII-E, as condições de salto são calculadas numericamente por meio de um processo iterativo explícito e quarta ordem é obtida. A seguir são apresentadas as ideias desse método.

Considere uma função $f(x)$ com uma descontinuidade no ponto $x = x_\alpha$. Esse ponto está entre os pontos de malha x_i e x_{i+1} . Ao se realizar cálculos no ponto x_i , o valor de $f(x_{i+1})$ não pode ser usado diretamente, devido à presença da descontinuidade. Deve-se extrapolar um valor para $f(x_{i+1})$. O MII usa uma expansão em série de Taylor no ponto x_α para aproximar $f(x)$ no ponto x_{i+1} .

Assumimos que $f(x)$ seja analítica em todos os pontos do domínio $D = \{x | x_i \leq x \leq x_{i+1}\}$, exceto no ponto x_α onde há um salto (descontinuidade) no valor da função ou suas derivadas. A expansão em série de Taylor padrão em torno de x_i não pode ser usada para aproximar $f(x_{i+1})$,

a menos que um termo de correção J_α seja adicionado:

$$f(x_{i+1}) = f(x_i) + f^{(1)}(x_i)h + f^{(2)}(x_i)\frac{h^2}{2!} + \dots + J_\alpha, \quad (3)$$

onde

$$J_\alpha = [f^{(0)}]_\alpha + [f^{(1)}]_\alpha(h^+) + \frac{1}{2!}[f^{(2)}]_\alpha(h^+)^2 + \frac{1}{3!}[f^{(3)}]_\alpha(h^+)^3 + \dots + \frac{1}{n!}[f^{(n)}]_\alpha(h^+)^n, \quad (4)$$

$h = x_{i+1} - x_i$ e $h^+ = x_{i+1} - x_\alpha$. O termo $[f^{(n)}]_\alpha$ representa o salto no valor da n -ésima derivada da função em $x = x_\alpha$, isto é,

$$[f^{(n)}]_\alpha = \lim_{x \rightarrow x_\alpha^+} f^{(n)}(x) - \lim_{x \rightarrow x_\alpha^-} f^{(n)}(x). \quad (5)$$

Usando o termo de correção J_α , pode-se modificar qualquer método de diferenças finitas, e o método corrigido pelo salto deverá manter a ordem de precisão do original quando o estêncil envolver uma singularidade ou salto na função.

3.1.1 Discretização espacial

As derivadas presentes em (1) e (2) serão discretizadas utilizando o método de diferenças finitas compactas de quarta ordem [2]. Na presença de descontinuidades, uma aproximação para a primeira derivada pode ser escrita como

$$L_{i-1}^1 f_{i-1}^{(1)} + L_i^1 f_i^{(1)} + L_{i+1}^1 f_{i+1}^{(1)} = R_{i-1}^1 f_{i-1} + R_i^1 f_i + R_{i+1}^1 f_{i+1} + (L_I^1 J_{\alpha 1} - R_I^1 J_{\alpha 0}), \quad (6)$$

e para a segunda derivada,

$$L_{i-1}^2 f_{i-1}^{(2)} + L_i^2 f_i^{(2)} + L_{i+1}^2 f_{i+1}^{(2)} = R_{i-1}^2 f_{i-1} + R_i^2 f_i + R_{i+1}^2 f_{i+1} + (L_I^2 J_{\alpha 2} - R_I^2 J_{\alpha 0}), \quad (7)$$

onde L_i^n e R_i^n são, respectivamente, os coeficientes dos lados esquerdo e direito da aproximação para a n -ésima derivada e $J_{\alpha n}$ são as expansões dos saltos em série de Taylor de $f^{(n)}$ no ponto $x = x_\alpha$. $I = i + 1$ ou $I = i - 1$ dependendo da posição da interface. Observe que $J_{\alpha 1}$, $J_{\alpha 2}$ são as expansões dos saltos em relação a derivada (valor desconhecido) e $J_{\alpha 0}$ em relação a função (valor conhecido).

Em uma malha colocada, para o cálculo de $J_{\alpha n}$ como em (4), o valor de (h^+) é sempre o mesmo. No entanto, em uma malha deslocada, dois casos devem ser considerados, como visto na Figura 2.

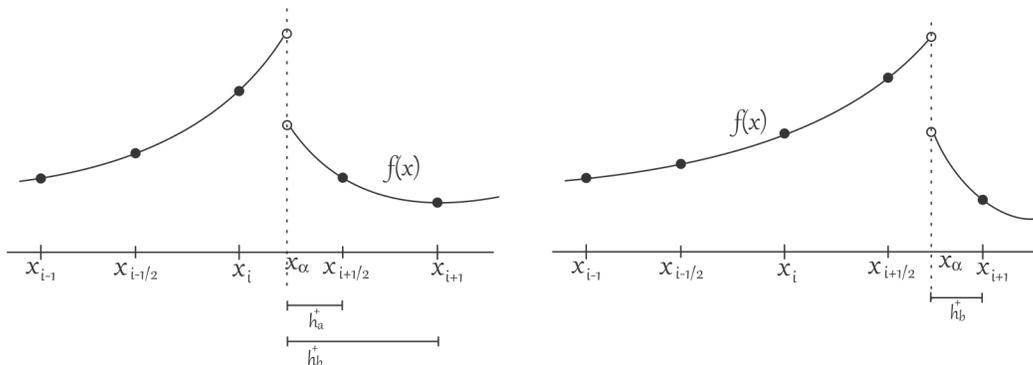


Figura 2: Casos possíveis onde a interface intersecta a malha. Na esquerda, tanto a quantidade conhecida (valor da função) quanto a desconhecida (derivadas) estão depois da descontinuidade. Na direita, apenas a quantidade desconhecida está depois da descontinuidade.

No caso descrito na Figura 2, as quantidades desconhecidas (primeira ou segunda derivada) estão nos pontos com índices inteiros ($i, i + 1, \dots$). As quantidades conhecidas (valor da função)

estão nos pontos com índices semi-inteiros $(i + 1/2, i + 3/2, \dots)$. Quando tanto a quantidade conhecida quanto a desconhecida estão depois da descontinuidade, são necessários dois $J_{\alpha n}$, cada um com seu valor apropriado para (h^+) . Esse caso pode ser visto no lado esquerdo da Figura 2. Se apenas a quantidade desconhecida estiver depois da descontinuidade, apenas um valor de $J_{\alpha n}$ é necessário, como pode ser visto no lado direito da Figura 2.

Os saltos são definidos como

$$[f^{(n)}]_{\alpha} = f_{+}^{(n)} - f_{-}^{(n)}, \tag{8}$$

em que $f_{+}^{(n)}$ e $f_{-}^{(n)}$ podem ser obtidos numericamente por

$$f_{+}^{(n)} = c_{n_{\alpha+}} f_{\alpha} + c_{n_{i+2}} f_{i+2} + c_{n_{i+3}} f_{i+3} + c_{n_{i+4}} f_{i+4} + c_{n_{i+5}} f_{i+5} + c_{n_{i+6}} f_{i+6}, \tag{9}$$

$$f_{-}^{(n)} = c_{n_{\alpha-}} f_{\alpha} + c_{n_{i-1}} f_{i-1} + c_{n_{i-2}} f_{i-2} + c_{n_{i-3}} f_{i-3} + c_{n_{i-4}} f_{i-4} + c_{n_{i-5}} f_{i-5}. \tag{10}$$

Note que, os pontos x_i e x_{i+1} foram evitados para contornar problemas de instabilidade. Essa aproximação possui quarta ordem de precisão.

Os coeficientes c_n para calcular

$$f_{\alpha}^{(n)} = c_{\alpha} f_{\alpha} + c_i f_i + c_{i+1} f_{i+1} + c_{i+2} f_{i+2} + c_{i+3} f_{i+3} + c_{i+4} f_{i+4}, \tag{11}$$

são obtidos da resolução do sistema linear

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & h_i & h_{i+1} & h_{i+2} & h_{i+3} & h_{i+4} \\ 0 & h_i^2 & h_{i+1}^2 & h_{i+2}^2 & h_{i+3}^2 & h_{i+4}^2 \\ 0 & h_i^3 & h_{i+1}^3 & h_{i+2}^3 & h_{i+3}^3 & h_{i+4}^3 \\ 0 & h_i^4 & h_{i+1}^4 & h_{i+2}^4 & h_{i+3}^4 & h_{i+4}^4 \\ 0 & h_i^5 & h_{i+1}^5 & h_{i+2}^5 & h_{i+3}^5 & h_{i+4}^5 \end{bmatrix} \begin{bmatrix} c_{\alpha} \\ c_i \\ c_{i+1} \\ c_{i+2} \\ c_{i+3} \\ c_{i+4} \end{bmatrix} = \begin{bmatrix} 1\delta_{n0} \\ 1\delta_{n1} \\ 2!\delta_{n2} \\ 3!\delta_{n3} \\ 4!\delta_{n4} \\ 5!\delta_{n5} \end{bmatrix}, \tag{12}$$

em que $h_i = x_i - x_{\alpha}$ e δ_{ij} é a função delta de Kronecker

$$\delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}. \tag{13}$$

3.2 Equação de Poisson

Ao resolver a equação de Poisson,

$$\nabla^2 f = g, \tag{14}$$

obtida no método da projeção, a equação (7) torna-se

$$R_{i-1}^2 f_{i-1} + R_i^2 f_i + R_{i+1}^2 f_{i+1} = L_{i-1}^2 f_{i-1}^{(2)} + L_i^2 f_i^{(2)} + L_{i+1}^1 f_{i+1}^{(2)} - (L_I^2 J_{\alpha 2} - R_I^2 J_{\alpha 0}). \tag{15}$$

Nesse caso, o valor da segunda derivada $(f^{(2)})$ é conhecido e o valor da função (f) passa a ser o desconhecido. Note que o termo de correção é adicionado no lado direito da equação, como proposto pelo MII-E [4]. Uma solução, adotada pelos autores, é começar com um valor tentativo para f e melhorá-lo iterativamente

$$f = \beta f_{\text{new}} + (1 - \beta) f. \tag{16}$$

Uma desvantagem dessa formulação é que se trata de um processo iterativo. O parâmetro de relaxação β influencia na convergência e é calibrado manualmente para cada problema. Uma escolha ruim pode significar uma convergência lenta ou não convergência.

Uma alternativa é calcular as correções como parte da solução diretamente. Assim, a equação a ser resolvida passa a ser

$$R_{i-1}^2 f_{i-1} + R_i^2 f_i + R_{i+1}^2 f_{i+1} + (L_I^2 J_{\alpha 2}^* - R_I^2 J_{\alpha 0}^*) = L_{i-1}^2 f_{i-1}^{(2)} + L_i^2 f_i^{(2)} + L_{i+1}^1 f_{i+1}^{(2)} - (L_I^2 J_{\alpha 2}^{**} - R_I^2 J_{\alpha 0}^{**}). \tag{17}$$

As condições de salto $J_{\alpha 2}, J_{\alpha 0}$ podem ser incorporadas na matriz do sistema linear, pois elas são aproximadas como funções lineares dos valores discretos de f . O termo $J_{\alpha n}^*$ envolve valores desconhecidos de f , geralmente dentro do domínio, e o termo $J_{\alpha n}^{**}$ envolve valores conhecidos de f , geralmente condições de contorno. Esse é o MII-I [5].

4 Resultados

A análise dos resultados é feita por meio do método das soluções manufaturadas [6]. No primeiro caso, é mostrada a análise de convergência no cálculo das derivadas ao utilizar o método das interfaces imersas de quarta ordem apresentado anteriormente (MII-I). No segundo caso, é simulado o escoamento em uma cavidade. Em ambos os casos, é utilizada uma malha deslocada, onde a pressão está no centro da célula e as velocidades nas arestas.

4.1 Cálculo das derivadas

Sejam as soluções prescritas

$$U = \sin(2\pi x) \sin(2\pi y), \quad V = \cos(2\pi x) \cos(2\pi y), \quad P = \sin(2\pi x) \cos(2\pi y),$$

no domínio $0 \leq x, y \leq 1$. O contorno está descolado de $\frac{1}{3}h$ em relação aos pontos de malha, portanto não passa pelos pontos onde são amostradas as velocidades ou pressão. h é a largura da célula. A Figura 3 mostra a evolução dos erros absolutos na norma L_∞ com o refinamento da malha. Observe que todas as derivadas convergem com ordem 4.

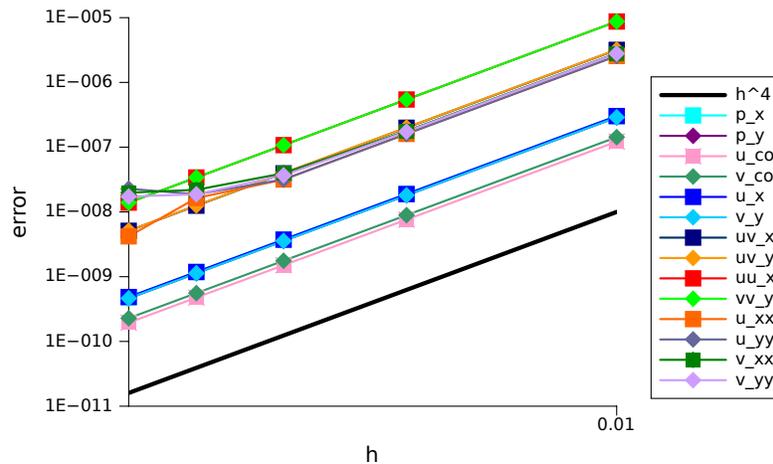


Figura 3: Gráfico $h \times |error|_\infty$. Análise de convergência no cálculo das derivadas. Os pontos correspondem a $h = 1/100, 1/200, 1/300, 1/400$ e $1/500$. Os subscritos em ϕ_x, ϕ_{xx} e ϕ_{co} referem-se, respectivamente, à primeira derivada, à segunda derivada e às interpolações das velocidades para o canto da célula. Essa última é necessária para o cálculo do termo convectivo.

4.2 Escoamento em uma cavidade

O problema clássico de escoamento laminar dentro de uma cavidade quadrada na qual a tampa se move é simulado. Uma solução analítica conhecida é dada por [7]. O modelo matemático consiste das equações de Navier-Stokes com algumas simplificações: estado estacionário, escoamento laminar bidimensional nas direções x e y , fluido incompressível. As condições de contorno para as velocidades u e v são do tipo Dirichlet: sendo nula em todos os pontos, exceto em $y = 1$, onde temos

$$u(x, 1) = 16(x^4 - 2x^3 + x^2). \tag{18}$$

Um termo fonte S está presente somente na equação da quantidade de movimento na direção y e é dado como em [7]:

$$S = -8\nu[24F(x) + 2f^{(1)}(x)g^{(2)}(y) + f^{(3)}(x)g(y)] - 64[F_2(x)G_1(y) - g(y)g^{(1)}(y)F_1(x)], \quad (19)$$

onde

$$\begin{aligned} f(x) &= x^4 - 2x^3 + x^2, \\ g(y) &= y^4 - y^2, \\ F(x) &= \int f(x)dx = 0.2x^5 - 0.5x^4 + x^3/3, \\ F_1(x) &= f(x)f^{(2)}(x) - [f^{(1)}(x)]^2 = -4x^6 + 12x^5 - 14x^4 + 8x^3 - 2x^2, \\ F_2(x) &= \int f(x)f^{(1)}(x)dx = 0.5[f(x)]^2, \\ G_1(y) &= g(y)g^{(3)}(y) - g^{(1)}(y)g^{(2)}(y) = -24y^5 + 8y^3 - 4y. \end{aligned}$$

A solução analítica é dada por

$$u(x, y) = 8f(x)g^{(1)}(y) = 8(x^4 - 2x^3 + x^2)(4y^3 - 2y), \quad (20)$$

$$v(x, y) = -8f^{(1)}(x)g(y) = -8(4x^3 - 6x^2 + 2x)(y^4 - y^2), \quad (21)$$

$$p(x, y) = 8\nu[F(x)g^{(3)}(y) + f^{(1)}(x)g^{(1)}(y)] + 64F_2(x)\{g(y)g^{(2)}(y) - [g^{(1)}(y)]^2\}. \quad (22)$$

A Figura 4 e a Tabela 1 apresentam os resultados, considerando: $\nu = 1$, $\Delta t = 10^{-6}$. Observe na figura que o campo de velocidades é semelhante ao problema clássico, no qual a tampa se move com velocidade constante igual a 1. A análise de convergência é observada na tabela. Apesar de os cálculos das derivadas feitos separadamente apresentarem quarta ordem, o método como um todo aparenta se aproximar com terceira ordem de precisão. Além disso, para esse problema, a interface foi colocada muito próxima aos pontos de malha (distância de 10^{-10} ao invés de $\frac{1}{3}h$).

Esses são, no entanto, resultados iniciais. Espera-se melhorar o método de forma que apresente quarta ordem para qualquer posicionamento da interface.

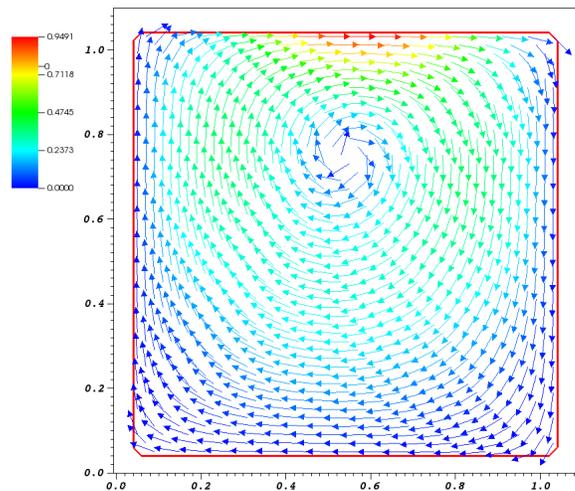


Figura 4: Escoamento em uma cavidade. Campo de velocidades. $\nu = 1$, $\Delta t = 10^{-6}$.

5 Conclusões

Foi apresentado o método das interfaces imersas implícito (MII-I), uma adaptação do método das interfaces imersas explícito (MII-E) desenvolvido por [4], que permite a simulação de escoamento de fluidos em geometrias complexas. Os resultados, ainda preliminares, são promissores. Foi obtida quarta ordem de aproximação para cálculo de derivadas e para a solução da equação de Poisson. No entanto, para a resolução das equações de Navier-Stokes por meio do método

Malha	Norma L_1		Norma L_2		Norma L_∞	
	erro	ordem	erro	ordem	erro	ordem
	velocidade u					
50x50	2.036261e-06	-	2.541972e-06	-	8.457417e-06	-
100x100	2.599824e-07	2.9694	3.274744e-07	2.9565	1.205539e-06	2.8105
150x150	7.759571e-08	2.9820	9.813933e-08	2.9719	3.890900e-07	2.7891
200x200	3.285584e-08	2.9873	4.165165e-08	2.9791	1.825710e-07	2.6302
	velocidade v					
50x50	1.132684e-06	-	1.959281e-06	-	1.104794e-05	-
100x100	1.456096e-07	2.9596	2.581196e-07	2.9242	1.664289e-06	2.7308
150x150	4.355497e-08	2.9766	7.795950e-08	2.9527	5.349789e-07	2.7991
200x200	1.846175e-08	2.9836	3.321987e-08	2.9652	2.396510e-07	2.7914

Tabela 1: Erro e ordem de convergência.

da projeção, ainda há obstáculos que impedem a obtenção de quarta ordem. Esses obstáculos provavelmente estão relacionados com o método da projeção em si, em particular com a resolução da equação de Poisson. Pretende-se investigar formas alternativas de se resolver a equação de Poisson e também outras formulações do método da projeção.

6 Agradecimentos

Pelo apoio financeiro, processos nº2012/04471-5 e nº2013/21501-8 Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP).

Referências

- [1] J. L. Guermond, L. Quartapelle, On stability and convergence of projection methods based on pressure poisson equation, *I. J. Numerical Methods in Fluids*, 26 (1998) 1039–1053.
- [2] S. K. Lele, Compact finite difference schemes with spectral-like resolution, *Journal of Computational Physics*, 103 (1992) 16–42.
- [3] R. J. LeVeque, Z. Li, The immersed interface method for elliptic equations with discontinuous coefficients and singular sources, *SIAM J. Numerical Analysis*, 31 (1994) 1019–1044.
- [4] M. N. Linnick, H. F. Fasel, A high-order immersed interface method for simulating unsteady incompressible flows on irregular domains, *J. Computational Physics*, 204 (2005) 157–192.
- [5] G. A. Reis, I. V. M. Tasso, L. F. Souza, J. A. Cuminato, “Fast and precise solution for the Poisson equation in the presence of interfaces with discontinuities”, 22nd International Congress of Mechanical Engineering, Ribeirão Preto, 2013.
- [6] C. J. Roy, Review of code and solution verification procedures for computational simulation, *Journal of Computational Physics*, 205 (2005) 131–156.
- [7] T. Shih, C. Tan, B. Hwang, Effects of grid staggering on numerical schemes, *International Journal for Numerical Methods in Fluids*, 9 (1989) 193–212.
- [8] D. A. von Terzi, M. N. Linnick, J. Seidel, H. F. Fasel, “Immersed boundary techniques for high-order finite-difference methods”, AIAA Fluid Dynamics Conference and Exhibit, Anaheim, CA, 2001.