

# Acompanhamento de fronteira livre-de-malhas e o método de mínimos-quadrados-móveis algébricos

João Paulo Gois, Anderson Nakano,  
Luis Gustavo Nonato, Gustavo C. Buscaglia  
Instituto de Ciências Matemáticas e de Computação, ICMC, USP,  
13560-970, São Carlos, SP  
E-mail: {jpgois,nakano,nonato,gustavo.buscaglia}@icmc.usp.br

**Resumo:** Neste trabalho apresentamos um método livre-de-malhas (meshless) de acompanhamento de fronteiras móveis. A interface é representada apenas por um conjunto de partículas desconectadas utilizando uma versão aprimorada do método de mínimos-quadrados-móveis algébricos (MQMA). Essas partículas são transportadas de forma Lagrangeana e regeneradas periodicamente com o intuito de manter a densidade e a distribuição de partículas controladas. Os resultados demonstram a precisão do método e também sua habilidade em lidar com grandes distorções na geometria da interface.

## 1 Introdução

A simulação numérica de fenômenos relacionados com interfaces que se alteram com o tempo necessita de duas tarefas distintas e independentes: a primeira, foco deste trabalho, é a representação da interface durante sua evolução e a segunda é a solução das equações que governam o problema durante a variação do tempo.

Partículas não-organizadas têm sido utilizadas no âmbito de simulação numérica em duas vertentes. Na primeira, diversos métodos já foram propostos, tal como o *Smoothed Particle Hydrodynamics* – SPH, métodos *meshless* e métodos baseados em partículas [9, 15, 10]. Esses métodos usam partículas em todo o domínio, de forma que as duas tarefas anteriores estão conectadas. Contudo, a quantidade de trabalhos baseados em partículas não-organizadas para a tarefa de representação da interface separada do *solver* (Euleriano, por exemplo) é bem menor [24, 3, 8], mas não menos importante.

Assim, propomos um método para o proble-

ma de modelagem de uma interface  $\mathcal{S}(t)$  que evolui com o tempo  $t$ , encarregada de separar dois fluidos  $A$  e  $B$ , com a possibilidade de o fluido  $B$  ser o ambiente, por exemplo, o ar. Além disso, o problema ocorre no interior de um domínio finito  $\Omega$  com bordo  $\partial\Omega$ .

Os métodos *puramente Lagrangeanos* para modelagem de interfaces móveis consistem em definir um conjunto de partículas sobre a interface e movê-las conforme um campo de velocidades. Esses métodos têm se mostrado eficientes em termos de precisão numérica [12, 26, 25, 19], porém, apresentam três problemas básicos:

- Dificuldade em simular processos de rompimento e junção (*merging*) da superfície: é difícil lidar com mudanças topológicas;
- Problema em manter a densidade de partículas consistente com o desejável nível de discretização: regiões da superfície podem conter um número excessivo de pontos, acarretando um custo computacional desnecessário, ou podem conter, por outro lado, uma pequena quantidade, levando ao comprometimento da precisão do método. Para contornar tal problema, essas abordagens exigem métodos eficientes para criação e remoção de partículas;
- Custo computacional em manter uma estrutura de dados topológica: métodos puramente Lagrangeanos exigem que as partículas estejam conectadas (dependência da malha) [26, 25, 19].

Todos os problemas previamente citados acerca dos métodos Lagrangeanos têm feito os

métodos Eulerianos – tais como o método *Volume-of-Fluid* (VOF) [11, 17] ou o método de conjunto de nível (*Level-Set* – LS) [21, 16, 18] ou, ainda, combinações desses [20, 22] – os preferidos para modelagem de interfaces complexas com significantes alterações topológicas, tais como simulação de fusão de bolhas ou quebra de ondas [23, 27, 2]. Entretanto, apesar de os métodos Eulerianos tratarem da topologia da interface de forma natural, eles apresentam problemas de difusão da interface.

Portanto, as considerações prévias motivaram o desenvolvimento de métodos que sejam capazes de combinar a precisão das técnicas Lagrangeanas com a flexibilidade dos procedimentos Eulerianos. Torres e Brackbill [24] desenvolveram o *Point-Set Method* no qual o acompanhamento de fronteira também é livre de malhas. Tal método tem por finalidade evoluir um conjunto de partículas, conforme um campo de velocidade, para, então, construir uma função de nível  $\phi$ , através da Equação de Laplace em  $\Omega$ . Eles prescrevem  $\phi = 1$  nas células contendo partículas e  $\phi = 0$  no bordo  $\partial\Omega$ . A função resultante  $\phi$  é igual a 1 dentro de  $S$ , identificando a região ocupada pelo fluido  $A$ . Um procedimento de suavização baseado em *B-splines*, seguido por uma etapa de correção é, então, aplicado a  $\phi$  para que um conjunto de nível passe pelas partículas marcadoras. A técnica, então, regeira os pontos da interface como sendo projeções dos centros das células de uma grade computacional mais fina sobre o conjunto de nível  $\phi$ .

Enright et al. [3] propuseram um método denominado *Particle Level Set* (PLS), que combina uma abordagem Euleriana de LS e informações sobre a localização da interface dada por partículas Lagrangeanas. Em maiores detalhes, a informação Lagrangeana é incorporada por meio de operações de *blending*, da interface resultante do método Euleriano com esferas centradas em tais partículas Lagrangeanas que, no início do passo no tempo, são todas tangentes à interface. O PLS tem se mostrado eficiente em muitas aplicações [14, 16], pois a topologia da interface pode alterar naturalmente devido à parte Euleriana e por não necessitar de *solvers* altamente precisos para a equação do transporte do conjunto de nível [4], já que a parte Lagrangeana é capaz de corrigir as imperfeições da interface.

Hieber e Koumoutsakos [8] propuseram outra abordagem que também utiliza partículas Lagrangeanas para atualizar a interface. O método baseia-se em conjunto de nível (*Lagrangian Level-Set* – LSL) e faz uso de *Smooth Particle Hydrodynamics* (SPH).

Este trabalho está organizado da seguinte maneira: na Seção 2 apresentamos as contribuições que asseguram o potencial de nosso método de acompanhamento de fronteiras. Já na Seção 3, definimos formalmente o método de MQMA que utilizamos para a representação baseada em partículas da interface. Na Seção 4 descrevemos o nosso algoritmo de acompanhamento de fronteiras livre de malhas que faz uso de partículas geradas pelo método de MQMA. Experimentos numéricos que confirmam a eficiência do método estão presentes na Seção 5. Por fim, concluímos este trabalho e apresentamos direções futuras na Seção 6.

## 2 Contribuições

Nossa abordagem utiliza o potencial do método de mínimos-quadrados-móveis [13] em sua versão implícita para a representação de superfícies por pontos não-organizados no contexto de modelagem de interfaces em movimento. Assim como o método *Point-set* de Torres e Brackbill [24], definimos uma movimentação Lagrangeana de partículas que, no início de cada passo no tempo, são colocadas na interface  $S(t)$ . Todavia, no tempo  $t + \Delta t$  não se garante que as partículas obrigatoriamente estejam na interface  $S(t + \Delta t)$ , mas sim na superfície definida por elas, por meio da função implícita de MQMA.

Essa metodologia evita o custo de resolver os sistemas lineares de Torres e Brackbill [24] para a construção de  $\phi$ . Ainda mais, para a representação de MQMA de  $S(t + \Delta t)$ , simultaneamente executamos a re-geração dos pontos e a construção da função de nível  $\phi$ , que indica a região ocupada por cada fluido. Para tal tarefa, desenvolvemos uma técnica semelhante ao método de traçado-de-raios de Adamson e Alexa [1] para a detecção de pontos sobre  $S(t + \Delta t)$ , definidos pelas interseções de uma grade computacional (“de raios”) cartesiana tridimensional  $\mathcal{T}_h$ . O espaçamento  $h$  da grade  $\mathcal{T}_h$  é o único parâmetro que governa a aproximação de  $S(t)$ .

Isso significa que mantemos a distância entre as partículas marcadoras da ordem de  $h$ , e conseqüentemente, características menores que  $h$  são descartadas, como nos métodos Eulerianos. Porém, o acompanhamento da superfície se mantém Lagrangeano, portanto, preciso.

Por outro lado, se comparado ao método PLS, o custo do passo Euleriano é evitado e também o número de partículas é significativamente menor, bem como são re-gerados de forma mais fácil, pois são aqueles que realmente estão sobre a superfície e não os centros de esferas tangentes.

### 3 Mínimos-Quadrados-Móveis Algébricos

Consideremos um conjunto finito de pontos  $\mathcal{P} = \{\mathbf{p}_1, \dots, \mathbf{p}_m\}$  quase regularmente espaçados a uma distância média  $h$  e uma base polinomial  $\{q_k\}$  para os quais adotamos a base de esferas algébricas (i.e.;  $q_1(\mathbf{x}) = 1$ ,  $q_2(\mathbf{x}) = x_1$ ,  $q_3(\mathbf{x}) = x_2$ ,  $q_4(\mathbf{x}) = x_3$ ,  $q_5(\mathbf{x}) = x_1^2 + x_2^2 + x_3^2$ ). Definimos a função:

$$f(\underline{\alpha}, \mathbf{x}) = \sum_{k=1}^5 \alpha_k q_k(\mathbf{x}) \quad (1)$$

onde  $\underline{\alpha} = (\alpha_1, \dots, \alpha_5) \in \mathbb{R}^5$ . Consideremos uma função de localização contínua não-negativa  $\psi(s) = e^{-s^2}$  e um raio de influencia  $\Delta = 1.5h$ , escolhido após diversos testes numéricos. Um conjunto de funções pesos é definida como  $w_i(\mathbf{x}) = \psi\left(\frac{\|\mathbf{x} - \mathbf{p}_i\|}{\Delta}\right)$ . Os seguintes dois passos definem nosso método de MQMA:

**Passo 1:** *Calculando os vetores normais  $\mathbf{N}_i$  para cada  $\mathbf{p}_i \in \mathcal{P}$ .* Consideremos

$$\begin{aligned} Q &= \{ \underline{\alpha} \in \mathbb{R}^5 : \|\nabla_{\mathbf{x}} f(\underline{\alpha}, \mathbf{x})\| = 1 \\ &\quad \forall \mathbf{x} \text{ s.t. } f(\underline{\alpha}, \mathbf{x}) = 0 \} \\ &= \{ \underline{\alpha} \in \mathbb{R}^5 : \alpha_2^2 + \alpha_3^2 + \alpha_4^2 - 4\alpha_1\alpha_5 = 1 \} \end{aligned} \quad (2)$$

Para cada ponto  $\mathbf{p}_i \in \mathcal{P}$ , calcule  $\underline{\alpha}_{(i)} \in Q$  resolvendo o problema de minimização com restrição:

$$\underline{\alpha}_{(i)} = \arg \min_{\underline{\beta} \in Q} \sum_{j=1}^m w_j(\mathbf{p}_i) |f(\underline{\beta}, \mathbf{p}_j)|^2, \quad (3)$$

que se reduz a um problema de autovalor generalizado em  $\mathbb{R}^5$ .

A normal aproximada  $\mathbf{N}_i$  em  $\mathbf{p}_i$  é dada por

$$\mathbf{N}_i = \nabla_{\mathbf{x}} f(\underline{\alpha}_{(i)}, \mathbf{p}_i) / \|\nabla_{\mathbf{x}} f(\underline{\alpha}_{(i)}, \mathbf{p}_i)\|. \quad (4)$$

**Passo 2:** *Calculando a interface.* A interface  $\mathcal{S}$  por MQMA é definida como o conjunto de nível zero da função implícita  $F(\mathbf{x}) = \sum_{k=1}^5 \gamma_k(\mathbf{x}) q_k(\mathbf{x})$ . A última tarefa é o cálculo de  $\underline{\gamma}(\mathbf{x})$  para todo  $\mathbf{x}$ . Este cálculo é feito pela solução de um problema de minimização sem restrição [7]:

$$\begin{aligned} \underline{\gamma}(\mathbf{x}) &= \arg \min_{\underline{\beta} \in \mathbb{R}^5} \left( \sum_{i=1}^m w_i(\mathbf{x}) |f(\underline{\beta}, \mathbf{p}_i)|^2 + \right. \\ &\quad \left. + K \sum_{i=1}^m w_i(\mathbf{x}) \|\nabla_{\mathbf{x}} f(\underline{\beta}, \mathbf{p}_i) - \mathbf{N}_i\|^2 \right). \end{aligned} \quad (5)$$

onde  $K$  é um parâmetro de penalidade definido pelo usuário.

É importante mencionar que, antes de executar o Passo 2, deve-se orientar todos os vetores normais  $\mathbf{N}_i$  consistentemente para fora [6]. Assim, os valores positivos de  $F$  estarão no exterior de  $\mathcal{S}$  e os negativos no interior.

Recentemente, propomos a eliminação do parâmetro  $K$ , aplicando o limite  $K \rightarrow \infty$ . Com isso, definimos um método livre de parâmetros [6] que é capaz de aprimorar a robustez do método original de MQMA [7] (detalhes do método original de MQMA são encontrados em [6]).

### 4 Advecção da interface definida pelo MQMA

A atualização da interface  $\mathcal{S}(t)$  se dá pela sua evolução por um campo de velocidades:

$$\frac{d\mathbf{p}_i}{dt} = \mathbf{v}(\mathbf{p}_i(t), t), \quad \mathbf{p}_i(0) = \text{pontos iniciais.}$$

Desta forma,  $\mathcal{S}(t)$  denota a interface definida pelos pontos  $\mathcal{P}(t) = \{\mathbf{p}_i(t)\}_{i=1, \dots, m}$ . Com o intuito de manter os pontos bem distribuídos, é conveniente redistribuir estes pontos periodicamente. Para este propósito, aplicamos o algoritmo ilustrado pela Figura 1. Nós definimos um conjunto de “raios”  $R_{1h} \cup R_{2h} \cup R_{3h}$

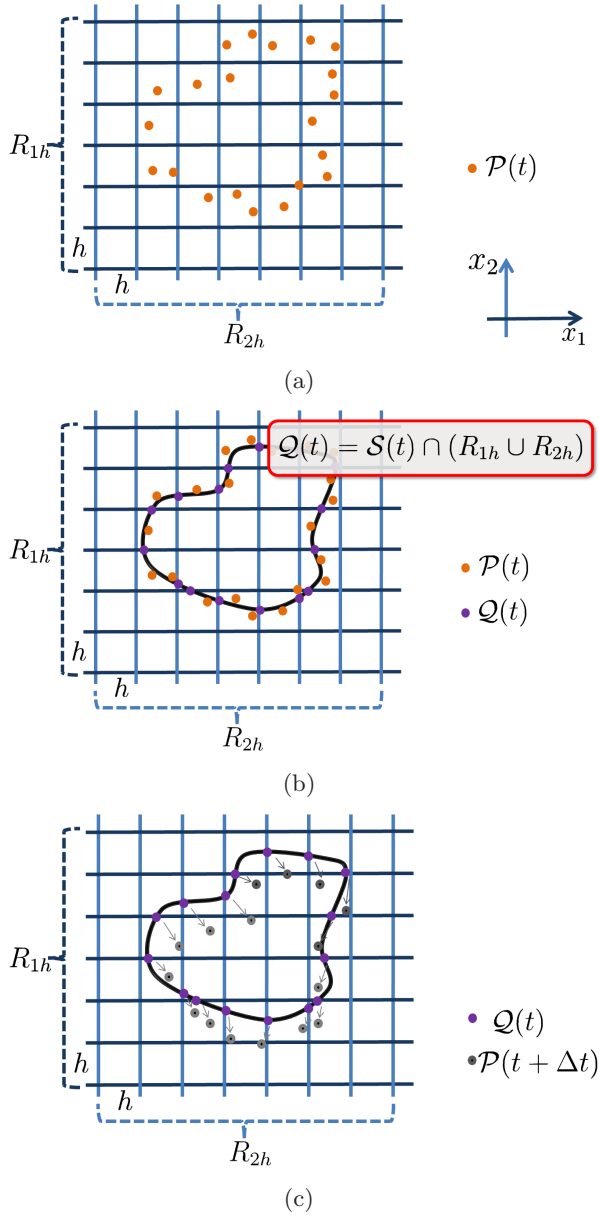


Figura 1: (a) Esboço do conjunto de raios de  $\mathcal{T}_h$  e do conjunto de pontos  $\mathcal{P}(t)$  – exemplo bidimensional. (b) O conjunto de pontos  $\mathcal{P}(t)$  define então a superfície  $\mathcal{S}(t)$ , no qual o conjunto de pontos  $\mathcal{Q}(t)$  é gerado nas interseções de  $\mathcal{S}(t)$  com os raios de  $\mathcal{T}_h$ . (c) Os pontos em  $\mathcal{Q}(t)$  são então advectados, definindo o novo conjunto de pontos  $\mathcal{P}(t + \Delta t)$ .

correspondentes às linhas de uma grade cartesiana  $\mathcal{T}_h$  com espaçamento  $h$ . No tempo  $t$ , o conjunto de pontos  $\mathcal{P}(t)$ , que define a interface  $\mathcal{S}(t)$ , é substituído por um conjunto de novos pontos  $\mathcal{Q}(t)$  calculados como as interseções dos raios com  $\mathcal{S}(t)$ . Estes pontos são então transportados, definindo um novo conjunto de pontos  $\mathcal{P}(t + \Delta t)$  para a nova iteração, onde  $\Delta t$  é o passo de tempo. Detalhes computacionais são encontrados em [6]. Este procedimento automaticamente mantém a densidade de pontos consistente com  $\mathcal{T}_h$ . De fato, a idéia é que  $\mathcal{T}_h$  seja a malha no qual um problema físico seja modelado.

É possível mostrar que o erro cometido durante a simulação é dado por  $\mathcal{O}(h^2/\Delta t) + \mathcal{O}(\Delta t^s)$ , onde  $\mathcal{O}(h^2)$  determina a ordem de aproximação da superfície e  $\mathcal{O}(\Delta t^s)$  é a ordem do método de integração numérica utilizado [6].

Programamos o nosso método com a linguagem C++ sob o sistema operacional Gentoo-Linux. Para o cálculo dos autovalores generalizados, utilizamos o método da Decomposição de Schur, disponível na biblioteca de domínio público *GNU Scientific Library* [5].

Por fim, é importante enfatizar que o processamento de cada raio de  $\mathcal{T}_h$  para a determinação das interseções com a superfície  $\mathcal{Q}$  é independente. Isto implica que a paralelização do nosso método é imediata.

## 5 Experimentos Numéricos

Demonstramos os potenciais dos métodos de MQMA e de acompanhamento de fronteiras através de dois experimentos numéricos: no primeiro mostramos como o método de MQMA lida com a reconstrução de uma superfície de classe  $C^0$ . Já no segundo, mostramos a capacidade do método de acompanhamento de fronteiras em lidar com interfaces que sofrem grandes distorções durante sua evolução no tempo.

Para o primeiro teste, consideramos pontos originados do 3/4 de uma esfera (Figura 2). Mostramos que, mesmo quando o número de pontos que representa a superfície é aumentado, o método original de MQMA [7] não é capaz de diferenciar entre o interior e o exterior corretamente nas “quinas” do modelo geométrico (Figura 2 - topo), que comprometeria o processo de simulação. Já o método aqui



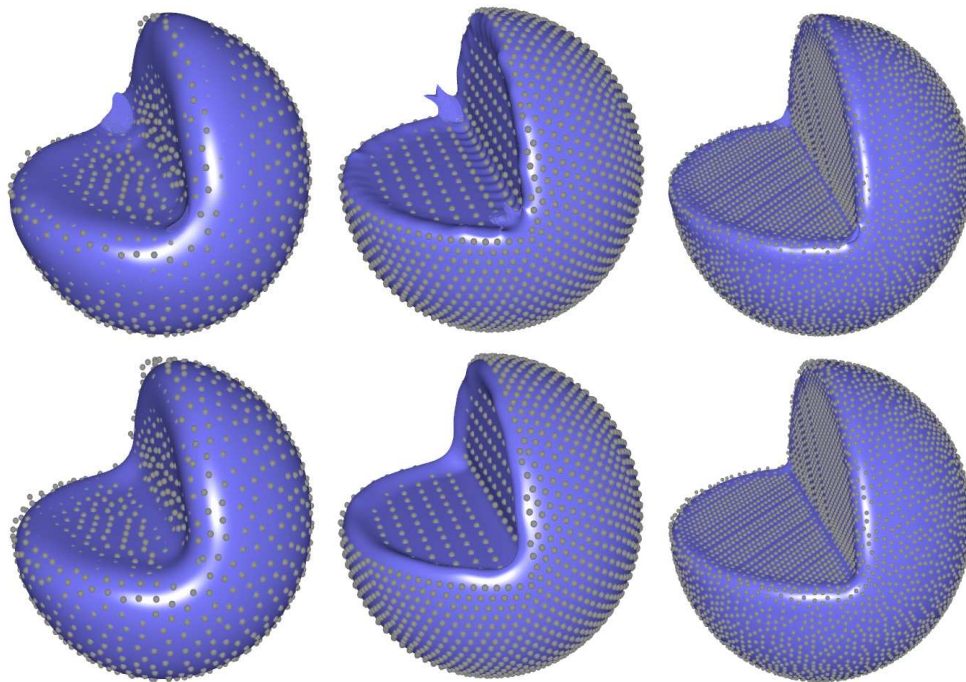


Figura 2: Comparação do MQMA original [7] (topo) e do método proposto (base): as imperfeições que ocorrem nas quinas das interfaces quando se utiliza o método de MQMA original são evitadas pelo método aqui proposto. Da esquerda para a direita, tanto no topo quanto na base, utilizamos 1030, 2131 e 4748 pontos.

proposto é capaz de representar a superfície de forma robusta e precisa (Figura 2 - base).

Submetemos a nossa abordagem a um problema clássico que inferiu a sua capacidade em representar interfaces sujeitas a grandes distorções e em preservar massa. Tal teste é denominado na literatura por torção tridimensional de uma esfera [3]. Na Figura 3 mostramos os resultados que obtemos mediante tal teste. É possível observar qualitativamente que os resultados demonstram o potencial de nosso método quando comparado com metodologias do estado-da-arte de representação de interfaces móveis [3, 8]. Também é possível notar que, com uma quantidade reduzida de pontos (ver Figura 3), o nosso método é capaz de preservar a topologia da interface mesmo sob severas distorções. Quantitativamente, a diferença entre o volume da esfera original e o da superfície obtida após o final da simulação é inferior a 0.1415%, demonstrando a precisão do método.

## 6 Conclusão

Apresentamos um método livre-de-malhas para a representação de interfaces que são advectadas conforme o tempo. Para isto, introduzimos o conceito de mínimos-quadrados-móveis algébricos, bem como uma variante que torna a técnica robusta. O método se mostrou eficiente em relação à preservação de massa e de geometria mesmo quando a interface é sujeita a grandes distorções.

Com respeito às direções futuras, estamos trabalhando para a incorporação desta técnica em *solvers* de simulação numérica de escoamento de fluidos multifásicos. Também pretendemos tornar público o código fonte deste método.

**Agradecimentos:** Este trabalho foi parcialmente financiado pela FAPESP e pelo CNPQ. Gustavo Carlos Buscaglia também está com o CNEA e CONICET (Argentina).

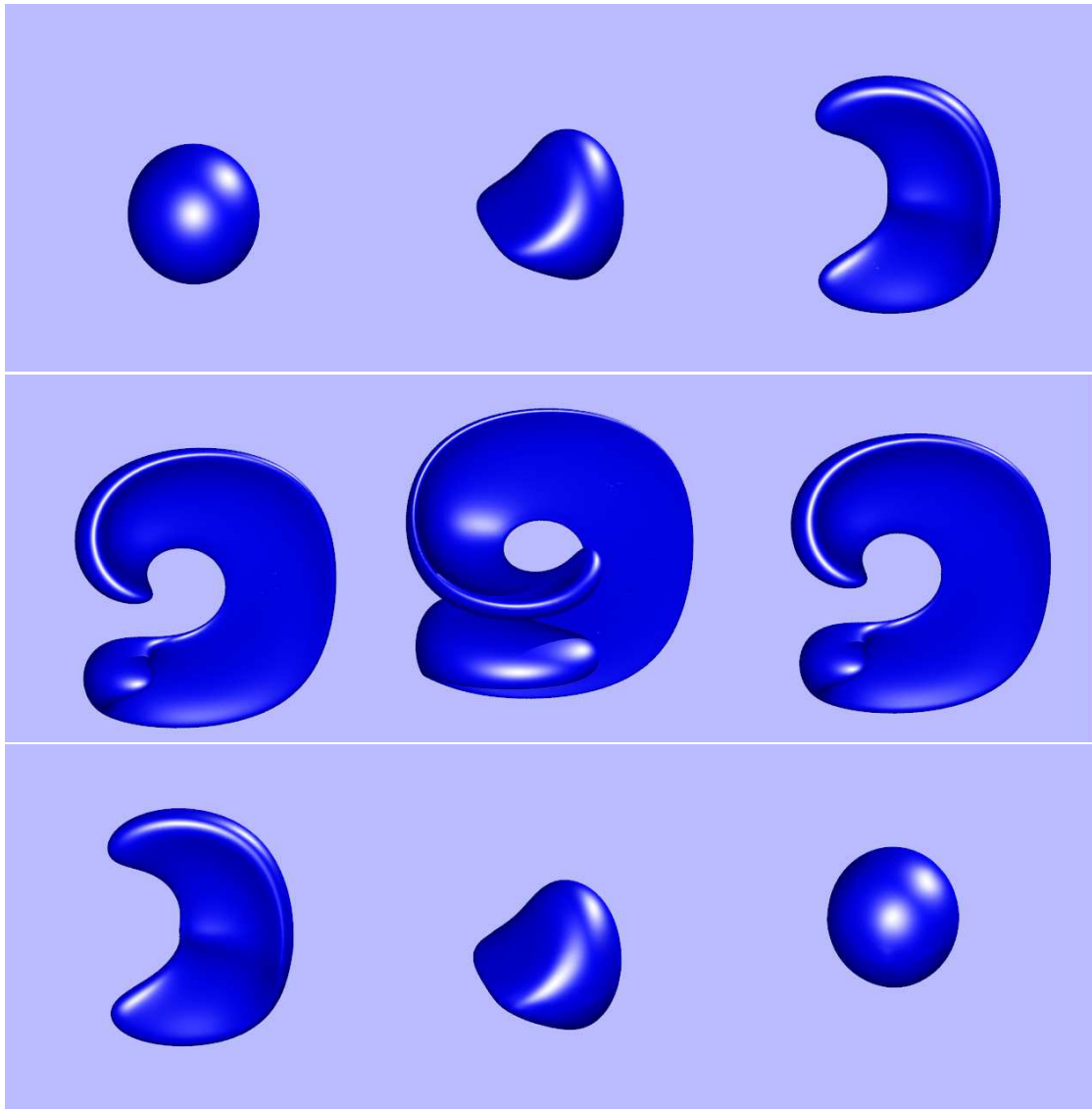


Figura 3: Torção tridimensional da esfera obtida numericamente em diferentes equispaçados passos de tempo. O domínio da simulação é o cubo unitário, o espaçamento da grade é  $h = 0.002$  e o passo de tempo é  $\Delta t = 0.0064$ . A simulação dura 628 passos de tempo. O número de pontos em cada um dos passos de tempo acima é (da esquerda para direita e de cima para baixo) 68020, 69314, 109728, 187571, 277021, 193238, 114898, 72774 e 68031 (que é uma quantidade significativamente menor quando comparada aos métodos de Enright et al. [3] e Hieber e Koumoutsakos [8]).

## Referências

- [1] A. Adamson e M. Alexa. Ray-tracing point set surfaces. In *Proceedings of the Shape Modeling International*, pág. 272–279, 2003.
- [2] P. Carrica, R. Wilson, e F. Stern. An unsteady single-phase level set method for viscous free surface flows. *Int. J. Numer. Meth. Fluids*, 53:229–256, 2007.
- [3] D. Enright, R. Fedkiw, J. Ferziger, e I. Mitchell. A Hybrid Particle Level Set Method for Improved Interface Capturing. *J. Comput. Phys.*, 183:83–116, 2002.
- [4] D. Enright, F. Losasso, e R. Fedkiw. A fast and accurate semi-Lagrangian particle level set method. *Computers and Structures*, 83:479–490, 2005.
- [5] Gsl – Gnu Scientific Library <http://www.gnu.org/software/gsl> (março/2008)
- [6] J. P. Gois, A. Nakano, L. G. Nonato, e G. C. Buscaglia. Front-tracking with moving-least-squares surfaces. *Journal of Computational Physics – sob revisão*, 2008.
- [7] G. Guennebaud e M. Gross. Algebraic point set surfaces. In *SIGGRAPH '07*, artigo 23, 2007. ACM.
- [8] S. Hieber e P. Koumoutsakos. A Lagrangian particle level set method. *J. Comput. Phys.*, 210:342–367, 2005.
- [9] A. Huerta, T. Belytschko, S. Fernandez-Mendez, e S. Rabczuk. *Meshfree methods*, volume 1, pág. 279–309. Wiley, 2004. in *Encyclopedia of Computational Mechanics*, Stein, E., de Borst, R. e Hughes T. J. R. (eds.).
- [10] P. Koumoutsakos. Multiscale flow simulations using particles. *Annu. Rev. Fluid Mech.*, 37:457–487, 2005.
- [11] B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, e G. Zanetti. Modelling Merging and Fragmentation in Multiphase Flows with SURFER. *J. Comput. Phys.*, 113:134–147, 1994.
- [12] D. Lakehal, M. Meier, e M. Fulgosi. Interface tracking towards the direct simulation of heat and mass transfer in multiphase flows. *International Journal of Heat and Fluid Flow*, 23:242–257, 2002.
- [13] D. Levin. The approximation power of moving least-squares. *Mathematics of Computation*, 67(224):1517–1531, 1998.
- [14] F. Losasso, R. Fedkiw, e S. Osher. Spatially adaptive techniques for level set methods and incompressible flow. *Computers and Fluids*, 35:995–1010, 2006.
- [15] J. Monaghan. Simulating free surface flows with SPH. *J. Comput. Phys.*, 110:399–406, 1994.
- [16] S. Osher e R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*, volume 153 of *Applied Mathematical Sciences*. Springer, 2003.
- [17] R. Scardovelli e S. Zaleski. Direct numerical simulation of free-surface and interfacial flow. *Annu. Rev. Fluid Mech.*, 31:567–603, 1999.
- [18] J. A. Sethian e P. Smereka. Level Set Methods for Fluids Interfaces. *Annual Rev. Fluid Mech.*, 35:341–372, 2003.
- [19] F. Sousa, N. Mangiavacchi, L. Nonato, A. Castelo, M. Tomé, V. Ferreira, J. Cuminato, e S. McKee. A front-tracking method for simulation of 3D multifluid flows with free surfaces. *J. Comput. Phys.*, 198:469–499, 2004.
- [20] M. Sussman. A second order coupled level set and volume-of-fluid method for computing growth and collapse of vapor bubbles. *J. Comput. Phys.*, 187:110–136, 2003.
- [21] M. Sussman e E. Fatemi. An efficient, interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow. *SIAM J. Sci Comput.*, 20(4):1165–1191, 1999.
- [22] M. Sussman e E. G. Puckett. A coupled level set and volume-of-fluid method for computing 3d and axisymmetric incompressible two-phase flows. *J. Comput. Phys.*, 162:301–337, 2000.
- [23] M. Sussman, P. Smereka, e S. Osher. A Level Set Approach for Computing Solutions to Incompressible Two-Phase Flow. *J. Comput. Phys.*, 114:146–159, 1994.
- [24] D. Torres e J. Brackbill. The point-set method: Front-tracking without connectivity. *J. Comput. Phys.*, 165:620–644, 2000.
- [25] G. Tryggvason, B. Bunner, A. Esmaeeli, D. Juric, N. Al-Rawahi, W. Tauber, J. Han, S. Nas, e Y. J. Jan. A Front-Tracking Method for the Computations of Multiphase Flow. *J. Comput. Phys.*, 169:708–759, 2001.
- [26] S. Unverdi e G. Tryggvason. A front-tracking method for viscous, incompressible, multi-fluid flows. *J. Comput. Phys.*, 100:25–37, 1992.
- [27] R. Wilson, P. Carrica, e F. Stern. Simulation of ship breaking bow waves and induced vortices and scars. *Int. J. Numer. Meth. Fluids*, 54(4):419–451, 2007.