

3

Metodologia numérica

Neste capítulo serão abordados os principais tópicos relativos à solução numérica do escoamento na superfície de uma gota, utilizando a formulação integral de contorno. De um ponto de vista global, deseja-se determinar a forma da superfície de uma gota sujeita a um determinado escoamento $\mathbf{u}_\infty$ e o campo de velocidade sobre a mesma. Para isso será usado o *Método Integral de Contorno*. Vários métodos numéricos são candidatos a resolver esse problema, como os métodos de diferenças, volumes ou elementos finitos, por exemplo. No entanto, diversos pontos indicam que a escolha mais adequada para a solução numérica do presente problema é o Método Integral de Contorno. A linearidade das equações governantes, a existência de um teorema recíproco e de uma solução fundamental permitem que se obtenha uma expressão fechada, em termos de integrais de superfície, para o campo de velocidade sobre a superfície da gota. A expressão integral é, por assim dizer, a solução analítica do problema. Sob esse aspecto, o método numérico presta-se basicamente a solução de integrais de superfície, eventualmente associadas à solução de um sistema linear de equação algébricas, cujo erro numérico está relacionado apenas à regra de integração numérica. Por outro lado, no método de elementos finitos, por exemplo, busca-se uma solução aproximada do problema, baseada em um espaço de funções que não necessariamente contempla todas as direções da solução exata. Outro aspecto atrativo do Método Integral de Contorno é que a solução pode ser obtida com a discretização apenas da superfície das partículas, não havendo a necessidade de se resolver o escoamento interior ou exterior à mesma. Dessa forma, há uma redução dimensional, desde que um problema em três dimensões é resolvido por meio de integrais sobre uma superfície no espaço. Esse fato reflete-se drasticamente no número pontos de controle necessários para a discretização do problema, já que em uma discretização por elementos finitos seria preciso resolver também os escoamentos dentro e fora da gota. Sendo assim, o tempo computacional necessário para uma simulação utilizando o Método Integral de Contorno é muitas vezes menor do que quando uma técnica tradicional é empregada. Vale lembrar que, em geral, esse método

pode apenas ser empregado em problemas governados por operadores lineares. Assim sendo, seu escopo de aplicação fica restrito quando comparado às demais técnicas mencionadas. Além disso, há questões relacionadas à singularidade das integrais de superfície e a obtenção de sistemas lineares cheios e, muitas vezes, mal condicionados. Esses pontos serão abordados durante este capítulo.

O problema numérico do ponto de vista do Método Integral de Contorno pode ser visto como o de se resolver a equação (2-80)¹, dadas posições iniciais para partículas materiais posicionadas sobre a superfície da gota. Dessa forma, é natural que um método de Euler seja empregado para realizar a marcha no tempo. Se $\mathbf{u}_\infty$ for permanente a evolução temporal deve acontecer até que não haja mais variação da forma da superfície. Se, por outro lado, o escoamento não-perturbado for oscilatório, então é preciso simular um intervalo de tempo suficientemente grande para que o transiente inicial possa ser descartado e para que seja possível extrair pelo menos um período de oscilação completo. A superfície em sua forma inicial pode ser discretizada em pontos de controle sobre os quais elementos geométricos, como o vetor normal e a curvatura média local, além da velocidade superficial, devem ser conhecidos. Um pós-processamento é necessário para que sejam extraídas quantidades associadas à geometria, como deformação e orientação em relação ao escoamento. O mesmo vale para as funções viscométricas. Como validação dos cálculos numéricos, comparações com resultados de outros pesquisadores e com teorias assintóticas desenvolvidas neste trabalho serão apresentadas.

3.1

Discretização espacial

A discretização espacial da superfície da gota é feita por meio do posicionamento de pontos de controle sobre a mesma. Considerando uma configuração inicial esférica, o método empregado inicia-se com a geração de um icosaedro regular. As faces desse sólido são subdivididas em triângulos equiláteros e cada novo vértice é projetado sobre a esfera que circunscreve o icosaedro original. O refinamento da discretização é controlado por um parâmetro f que é igual ao número de subdivisões das aresta do icosaedro, como sugere a figura (3.1).

As quantidades da malha são função de f , de sorte que $N_\Delta = 20f^2$, $N_\bullet = 10f^2 + 2$ e $N_\ell = 30f^2$, em que N_Δ , $N_\bullet$ e N_ℓ são o número de

¹ $\frac{d\mathbf{x}}{dt} = \mathbf{u}(\mathbf{x})$: Equação de evolução da posição das partículas fluidas que compõem a superfície da gota.

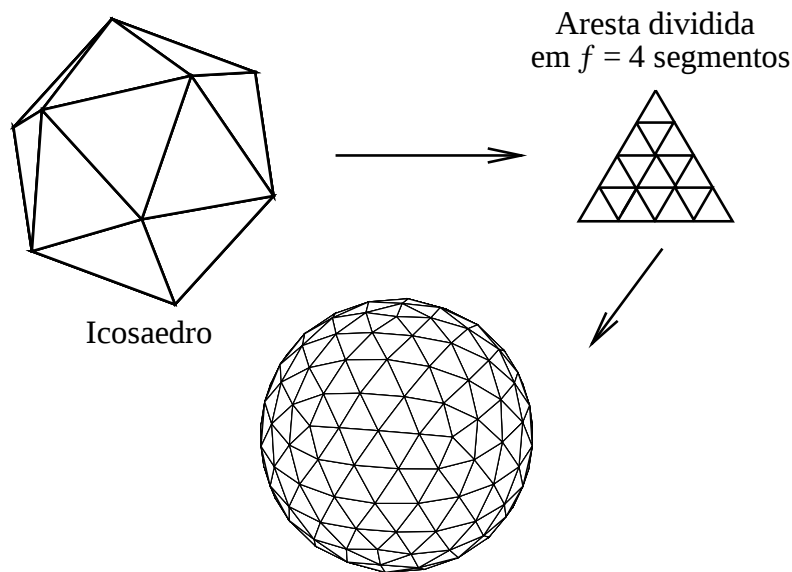


Figura 3.1: Processo de geração de malha sobre uma superfície esférica a partir de um icosaedro regular.

triângulos, nós e arestas, respectivamente. A subdivisão da superfície em triângulos é conveniente ao cálculo das integrais associadas ao Método Integral de Contorno. Dessa forma, os pontos de controle são conectados utilizando uma estrutura de dados composta por uma tabela de coordenadas dos nós e uma tabela de conexão dos elementos triangulares, na qual são listados os nós que compõem cada triângulo sobre a superfície, na ordem destrógera². São necessárias ainda estruturas para os nós vizinhos de cada nó e para os elementos vizinhos de cada nó. Também essas tabelas foram geradas na ordem destrógera. Vale comentar que o número de elementos e de nós vizinhos de cada ponto de controle pode variar entre 5 e 6, em virtude do método de geração da malha.

Esse processo de discretização é bastante difundido na literatura (Loewenberg & Hinch (1996), Zinchenko *et. al.* (1997), Cunha & Loewenberg (2003^a), Cunha *et. al.* (2003^b) & Bazhlekova *et. al.* (2004)) por sua simplicidade e pela boa qualidade da malha, isto é, triângulos praticamente equiláteros e com pouca variação do número de vizinhos.

²Isto é, no sentido anti-horário.

3.2

Elementos geométricos da superfície

Para o cálculo do vetor normal e da curvatura média local em cada nó da malha é possível utilizar uma variação do teorema de Stokes tal que vale a identidade

$$\bar{\kappa}(\mathbf{x}_c)\mathbf{n}(\mathbf{x}_c) = -\frac{1}{S_x} \int_C \mathbf{t} d\ell, \quad (3-1)$$

em que S_x é uma superfície que contém o ponto de controle $\mathbf{x}_c$, C é o contorno desta superfície e $\mathbf{t}$ um vetor unitário tangente à face triangular e ortogonal à trajetória C , conforme ilustra a figura (3.2). Esse método será referenciado como *método da integral de linha* no decorrer deste texto e foi empregado de forma pioneira por Loewenberg & Hinch (1996).

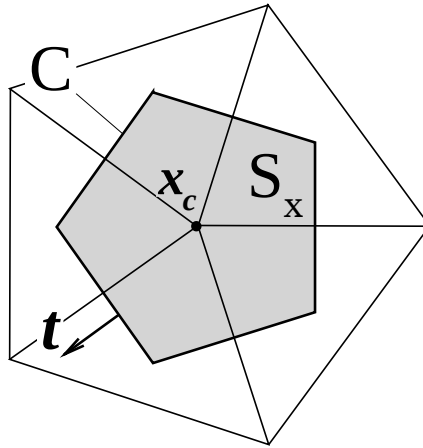


Figura 3.2: Definições geométricas para o cálculo da curvatura média local e do vetor normal pelo método da integral de linha.

A definição precisa da trajetória C é muito relevante na precisão obtida no cálculo da curvatura pelo método da integral de linha. Loewenberg & Hinch (1996) definiram C como a trajetória que passa pelas linhas bisectoras³ de cada triângulo vizinho do nó em $\mathbf{x}_c$. No entanto, esse procedimento torna-se impreciso quando o triângulo é obtuso, já que o circuncentro, neste caso, fica fora do perímetro do triângulo. Sendo assim, essa definição não é robusta o suficiente para ser utilizada em um código para simulação de gotas deformando

³Bisectora: Segmento que une o ponto médio de uma aresta do triângulo ao seu circuncentro.

arbitrariamente. De maneira alternativa, o contorno de S_x pode ser definido pela trajetória composta dos segmentos que unem os pontos médios dos lados do triângulo ao seu baricentro $\mathbf{x}_o$ (Bazhlekov *et. al.*, 2004). Dessa forma, e considerando a nomenclatura da figura (3.3), o ponto $\mathbf{x}_o$ é sempre interno ao triângulo.

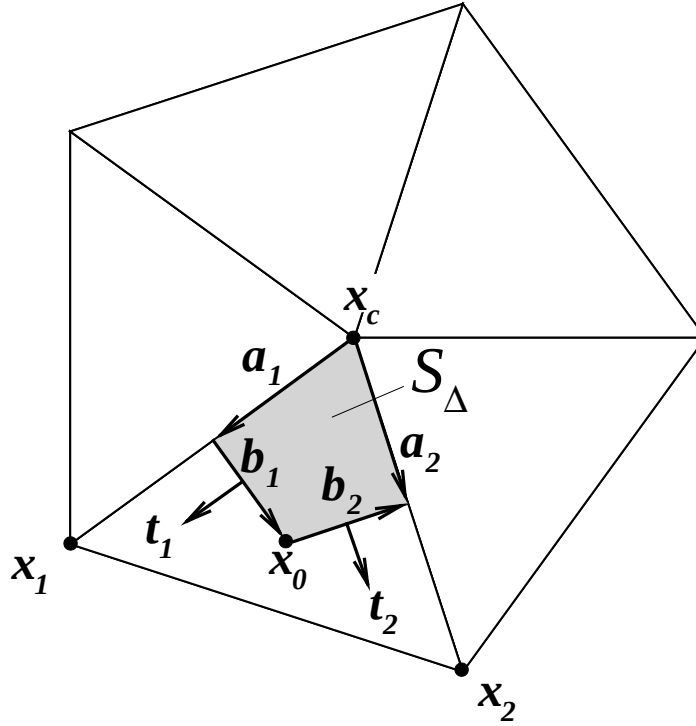


Figura 3.3: Detalhe dos vetores auxiliares para o cálculo da curvatura média local pelo método da integral de linha.

A integral de linha da equação (3-1) é calculada somando-se as contribuições de cada triângulo vizinho do nó. Considerando a nomenclatura da figura (3.3) e que são conhecidas as coordenadas no espaço dos ponto de controle $\mathbf{x}_c$, $\mathbf{x}_1$ e $\mathbf{x}_2$ temos que $\mathbf{x}_o = (\mathbf{x}_c + \mathbf{x}_1 + \mathbf{x}_2)/3$, $\mathbf{a}_1 = (\mathbf{x}_1 - \mathbf{x}_c)/2$, $\mathbf{a}_2 = (\mathbf{x}_2 - \mathbf{x}_c)/2$, $\mathbf{b}_1 = \mathbf{x}_c - \mathbf{x}_o - \mathbf{a}_1$ e $\mathbf{b}_2 = \mathbf{x}_o - \mathbf{x}_c - \mathbf{a}_2$. O vetor normal ao triângulo é dado por $\mathbf{n} = (\mathbf{a}_1 \times \mathbf{a}_2) / \|\mathbf{a}_1 \times \mathbf{a}_2\|$. Os vetores tangentes são calculados como $\mathbf{t}_1 = \mathbf{b}_1 \times \mathbf{n}$ e $\mathbf{t}_2 = \mathbf{b}_2 \times \mathbf{n}$. Assim sendo, esses têm a orientação desejada e norma igual ao comprimento do segmento que une $\mathbf{a}_1$ a $\mathbf{x}_o$, no caso de $\mathbf{t}_1$, e $\mathbf{a}_2$ a $\mathbf{x}_o$, no caso de $\mathbf{t}_2$. Sendo assim,

$$\int_C \mathbf{t} d\ell = \sum \mathbf{t}_i, \quad (3-2)$$

em que o somatório é realizado sobre todos os vetores tangentes em todos os elementos vizinhos do nó.

Outro método disponível para o cálculo das curvaturas consiste em ajustar uma superfície aos pontos vizinhos do nó $\mathbf{x}_c$ e determinar o valor da curvatura usando a equação da superfície ajustada. Para isso, um sistema de coordenadas cartesianas local $O\hat{x}\hat{y}\hat{z}$, orientado segundo o triedro $(\mathbf{e}_{\hat{x}}, \mathbf{e}_{\hat{y}}, \mathbf{e}_{\hat{z}})$, em que $\mathbf{e}_{\hat{z}} = \mathbf{n}$ e $\mathbf{x}_c$ é a origem pode ser definido, como indica a figura (3.4). Os vetores $\mathbf{e}_{\hat{x}}$ e $\mathbf{e}_{\hat{y}}$ são gerados arbitrariamente, mas respeitando a convenção da regra da mão direita, tal que $\mathbf{e}_{\hat{x}} \times \mathbf{e}_{\hat{y}} = \mathbf{e}_{\hat{z}}$. Nesse sistema de coordenadas, uma superfície quádrlica genérica, de equação $\hat{z} = A\hat{x} + B\hat{y} + C\hat{x}^2 + D\hat{x}\hat{y} + E\hat{y}^2$ é ajustada aos pontos $\mathbf{x}_c$ e seus vizinhos, usando o método dos mínimos quadrados. A curvatura média local em $\mathbf{x}_c = (0, 0, 0)_{O\hat{x}\hat{y}\hat{z}}$ é dada por

$$\bar{\kappa} = \nabla_s \cdot \mathbf{n} / 2 = -(C + E). \quad (3-3)$$

Para facilitar a referência, o método descrito neste parágrafo será denominado *método do ajuste*.

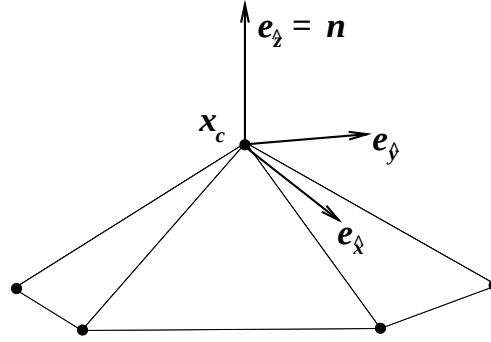


Figura 3.4: Sistema local de coordenadas utilizado para a definição da superfície quádrlica ajustada pelo ponto $\mathbf{x}_c$ e seus vizinhos.

Diversas versões do método do ajuste podem ser concebidas variando-se o tipo e/ou a representação da superfície e a quantidade de nós vizinhos utilizada no ajuste. Essa família de métodos é bastante utilizada em aplicações de reconhecimento de imagem, visão, topografia, entre outros (Mokhtarian *et al.*, 2001). Caracterizam-se por serem métodos robustos, capazes de calcular corretamente a curvatura local em superfícies descritas por malhas de elementos distorcidos ou sem malha conectada (“mesh free methods”).

Em testes preliminares observou-se que o erro associado ao cálculo do vetor normal utilizando a média dos vetores normais à faces vizinhas do nó é sempre menor do que o erro associado ao cálculo da curvatura. Sendo assim, optou-se por essa técnica, independentemente do método de cálculo de curvatura empregado. Para realizar uma comparação entre os métodos de cálculo de curvatura foram gerados elipsóides cujo vetor normal e a curvatura média local são conhecidos analiticamente. Para isso, dada uma superfície inicialmente esférica, cada ponto $\mathbf{x} = (x, y, z)$ da malha tem suas coordenadas multiplicadas

por $a, b, c \in \mathbb{R}_+^*$, respectivamente. Assim, a nova posição dos pontos de controle é tal que $\mathbf{x} = (ax, by, cz)$, de maneira que a superfície é um elipsóide do tipo $x^2/a^2 + y^2/b^2 + z^2/c^2 = 1$. Definido $g(x, y, z) = x^2/a^2 + y^2/b^2 + z^2/c^2 - 1$, temos que $\mathbf{n} = \nabla g / \|\nabla g\|$ e $\bar{\kappa} = 1/2(\nabla \cdot \mathbf{n})$. Considerando a expressão de $g(x, y, z)$,

$$\mathbf{n} = \frac{\left(\frac{x}{a^2}, \frac{y}{b^2}, \frac{z}{c^2}\right)}{\left(\frac{x^2}{a^4} + \frac{y^2}{b^4} + \frac{z^2}{c^4}\right)^{1/2}} \quad \text{e} \quad \bar{\kappa} = \frac{\left(\frac{1}{c^2} + \frac{1}{b^2}\right)x^2 + \left(\frac{1}{a^2} + \frac{1}{c^2}\right)y^2 + \left(\frac{1}{a^2} + \frac{1}{b^2}\right)z^2}{2\left(\frac{x^2}{a^4} + \frac{y^2}{b^4} + \frac{z^2}{c^4}\right)^{3/2}}. \quad (3-4)$$

Na figura (3.5a) são comparados os métodos de cálculo de curvatura para uma malha esférica para diferentes refinamentos da malha. A medida de erro usada nas comparações é a máxima diferença relativa definida por $\varepsilon_{\kappa} = \max\{|\bar{\kappa}_n - \bar{\kappa}_e|/\bar{\kappa}_e\}$, para a curvatura e $\varepsilon_{\mathbf{n}} = \max\{\|\mathbf{n}_n - \mathbf{n}_e\|/\|\mathbf{n}_e\|\}$ para o vetor normal, em que os sub-índices n e e denotam a solução numérica e exata, respectivamente. No caso da malha esférica, o método da integral de linha com

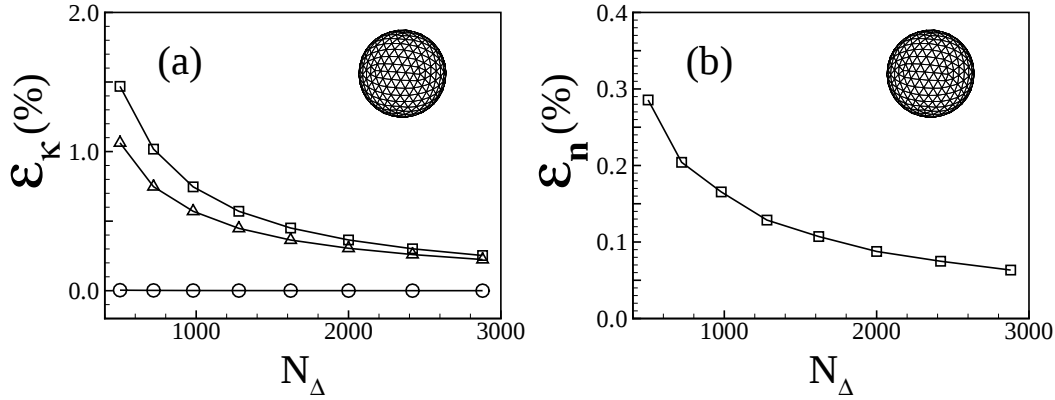


Figura 3.5: Malha esférica: $a = 1$, $b = 1$ e $c = 1$. (a) Erro percentual máximo associado ao cálculo de curvatura: $\square$ – método do ajuste; $\triangle$ – método da integral de linha em que $\mathbf{x}_o$ = baricentro do elemento; $\circ$ – método da integral de linha em que $\mathbf{x}_o$ = circuncentro do elemento. (b) Erro percentual máximo associado ao cálculo do vetor normal.

$\mathbf{x}_o$ igual ao circuncentro recupera exatamente o valor analítico da curvatura local, como mostra a figura (3.5a). Em (3.5b) observa-se que o erro associado ao cálculo do vetor normal é menor do que o cometido no cálculo da curvatura para os outros dois métodos. Para um elipsóide de pequena excentricidade, simulamos o comportamento dos métodos de cálculo de curvatura em regimes de pequenas deformações. Nesse caso o método da integral de linha com $\mathbf{x}_o$ igual ao circuncentro passa a apresentar pequenos desvios da solução exata, como pode ser observado na figura (3.6). Aumentando-se a excentricidade do elipsóide chegamos ao que se poderia chamar de um regime de deformações moderadas. Nesse caso, a performance do método da integral de linha em que $\mathbf{x}_o$ igual ao circuncentro fica prejudicada devido ao fato de existirem triângulos obtusos que compõem a malha. O erro associado ao cálculo da

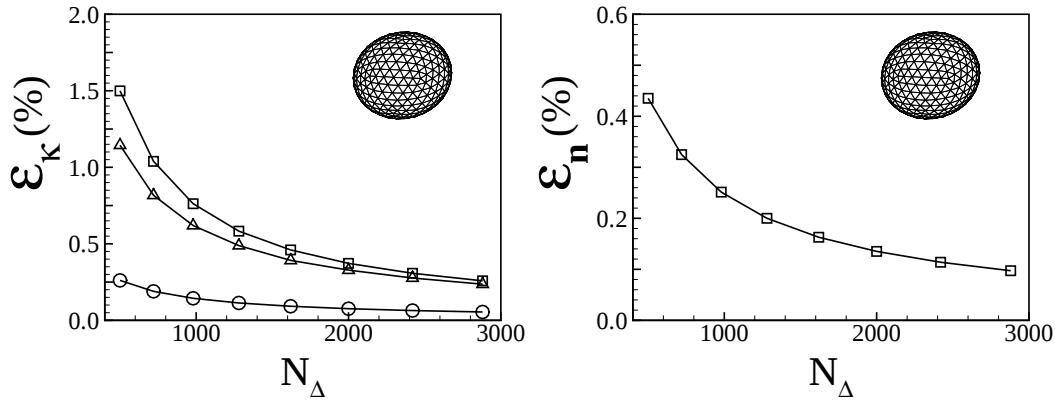


Figura 3.6: Elipsóide - 1: $a = 1$, $b = 0,9$ e $c = 0,8$. (a) Erro percentual máximo associado ao cálculo de curvatura: $\square$ – método do ajuste; $\triangle$ – método da integral de linha em que $\mathbf{x}_o$ = baricentro do elemento.; $\circ$ – método da integral de linha em que $\mathbf{x}_o$ = circuncentro do elemento. (b) Erro percentual máximo associado ao cálculo do vetor normal.

curvatura por esse método sequer converge à zero à medida que se aumenta o refinamento da discretização ficando o mesmo em torno de 10%, como pode ser visto na figura (3.7). Já os demais métodos apresentam bons resultados,

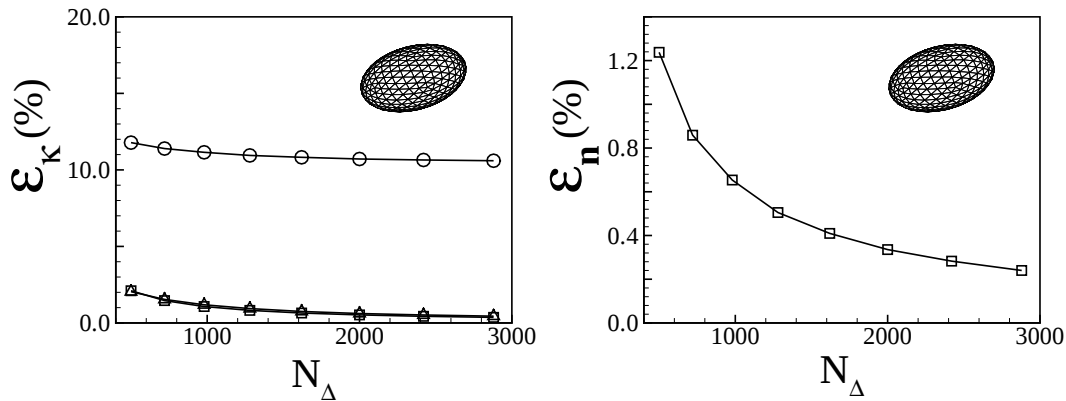


Figura 3.7: Elipsóide - 1: $a = 1$, $b = 0,5$ e $c = 0,25$. (a) Erro percentual máximo associado ao cálculo de curvatura: $\square$ – método do ajuste; $\triangle$ – método da integral de linha em que $\mathbf{x}_o$ = baricentro do elemento; $\circ$ – método da integral de linha em que $\mathbf{x}_o$ = circuncentro do elemento. (b) Erro percentual máximo associado ao cálculo do vetor normal.

sendo que o método da integral de linha com $\mathbf{x}_o$ igual ao baricentro é sempre ligeiramente mais preciso do que o método do ajuste. Levando em consideração que um código de MIC tem sua principal aplicação em regimes de moderadas e grandes deformações, elegeu-se esse procedimento para realizar os cálculos de curvatura sobre a malha. Além disso, sua implementação computacional é mais simples bem como o tempo de processamento cerca de um quarto do tempo gasto pelo método do ajuste. No entanto, acerca desse último comentário, vale dizer que o tempo de processamento necessário para o cálculo

da curvatura média local e do vetor normal é, em qualquer caso, proporcional à N_Δ . Já o cálculo das velocidades (seção 3.3) é proporcional à N_Δ^2 , de sorte que otimizações no tempo de processamento do cálculo da curvatura e do vetor normal são pouco significativas no desempenho global do código. Considerando uma situação típica das aplicações exploradas nesta tese, para uma malha com 1280 elementos, o tempo necessário para o cálculo do vetor normal e da curvatura representa aproximadamente 1,8% do tempo total de cálculo por iteração. Para malhas mais refinadas, essa proporção diminui quadraticamente.

3.3

Cálculo da velocidade

A velocidade sobre a superfície da gota é determinada solucionando a equação (2-85) após a subtração das singularidades e da regularização da dupla camada potencial como descrito na seção 2.7. A principal tarefa nesse caso é a solução de integrais de superfície de equações vetoriais do tipo $\mathbf{F}(\mathbf{x}_o) = \int_S \mathbf{f}(\mathbf{x}, \mathbf{x}_o) dS(\mathbf{x})$, em que $\mathbf{f}(\mathbf{x})$ neste contexto é uma função vetorial de $\mathbf{x}$ e de parâmetro $\mathbf{x}_o$. Após a discretização da superfície da gota em triângulos, a solução da integral sobre S pode ser realizada pela soma das integrais sobre os elementos delimitados pelas superfícies S^e , $e = 1 \dots N_\Delta$, de maneira que

$$\int_S \mathbf{f}(\mathbf{x}, \mathbf{x}_o) dS(\mathbf{x}) = \sum_{e=1}^{N_\Delta} \int_{S^e} \mathbf{f}(\mathbf{x}, \mathbf{x}_o) dS(\mathbf{x}). \quad (3-5)$$

Para a integração nas superfícies S^e utilizou-se a regra do trapézio para o cálculo de integrais sobre um triângulo de vértices $\mathbf{x}_1$, $\mathbf{x}_2$ e $\mathbf{x}_3$ no espaço, de maneira que (Press *et. al.*, 1992)

$$\int_S \mathbf{f}(\mathbf{x}, \mathbf{x}_o) dS(\mathbf{x}) \approx \sum_{e=1}^{N_\Delta} \frac{S^e}{3} \sum_{i=1}^3 \mathbf{f}(\mathbf{x}_i, \mathbf{x}_o). \quad (3-6)$$

O uso da fórmula de integração (3-6) introduz um erro $\mathcal{O}(N_\Delta^{-1})$ que é o mesmo cometido no cálculo da curvatura pelo método da integral de linha com elementos triangulares planos.

A maneira natural de se calcular uma integral do tipo de (3-6) é visitando elemento por elemento e somando suas contribuições. No entanto, considerando que em geral cada nó pertence à seis triângulos, o valor do integrando em cada nó é calculado repetidamente seis vezes, na grande maioria dos casos. Uma forma de contornar esse problema é realizar uma varredura por nós (Rallison, 1981) de tal forma que

$$\int_S \mathbf{f}(\mathbf{x}, \mathbf{x}_o) dS(\mathbf{x}) \approx \frac{1}{3} \sum_{i=1}^{N_\bullet} S_i^{viz} \mathbf{f}(\mathbf{x}_i, \mathbf{x}_o), \quad (3-7)$$

em que S_i^{viz} é o somatório das áreas dos elementos vizinhos do nó na posição $\mathbf{x}_i$. Realizando o cálculo das integrais de superfície dessa forma o tempo por iteração diminui para cerca de 17% do tempo que seria necessário se a integração fosse realizada segundo (3-6).

Em geral, o cálculo da velocidade sobre a superfície S da gota é realizado pela solução do sistema linear representado, algebricamente, pela equação (2-97). Para isso um algoritmo de substituições sucessivas foi implementado. Conhecendo-se $\mathbf{w}^p$ e considerando a tolerância tol_w o algoritmo para determinar $\mathbf{w}^{p+1}$ é

$$\left[\begin{array}{l} 1. \quad \mathbf{w}^{p+1} \leftarrow \beta [\mathbf{L}\mathbf{w}^p - \alpha_1 \mathbf{v}_1^p(\mathbf{r}^p, \mathbf{w}^p) - \alpha_2 \mathbf{s}(\mathbf{v}_2, \mathbf{w}^p)] + \mathbf{b}; \\ 2. \quad \text{Se } \varepsilon_w = \max\{\|\mathbf{w}^{p+1}(\mathbf{x}) - \mathbf{w}^p(\mathbf{x})\|\} < tol_w \text{ então } \rightarrow \text{FIM} \\ 3. \quad \mathbf{w}^p(\mathbf{x}) \leftarrow \mathbf{w}^{p+1}(\mathbf{x}) \end{array} \right] \quad (3-8)$$

Uma vez que os valores de $\mathbf{w}(\mathbf{x})$ são atualizados à medida que são calculados durante a iteração, o algoritmo anterior reproduz o método de Gauss-Seidel (Kreyszig, 1999) para solução de sistemas lineares. A convergência de $\mathbf{w}$ dentro de uma tolerância $tol_w = 10^{-6}$ para altas razões de viscosidade e altos números capilaridade ($\lambda = 100$ e $Ca = 50$, por exemplo), ocorre por volta de 6 iterações. Nos casos ordinários, com deformações mais significativas ($\lambda = 3$ e $Ca = 1$), a convergência é atingida por volta de 3 iterações. O refinamento da malha não afetou de forma significativa o número de iterações nos casos estudados neste trabalho. Após a determinação de $\mathbf{w}$ a velocidade $\mathbf{u}$ é obtida pela equação (2-98) tal que

$$\mathbf{u} = \mathbf{w}^{p+1} + \frac{\beta\alpha_1}{1 - \beta\alpha_1} \mathbf{v}_1^{p+1}(\mathbf{r}^{p+1}, \mathbf{w}^{p+1}).$$

Depois do processo iterativo na variável $\mathbf{w}$ ter convergido, o valor de $\mathbf{u}$ é usado no lado direito da equação (2-85) em que apenas a subtração de singularidades é realizada, gerando uma solução sem a remoção dos autovaleores espúrios, $\mathbf{u}_{padrão}$. Se $\varepsilon_u = \max\{\|\mathbf{u}(\mathbf{x}) - \mathbf{u}(\mathbf{x})_{padrão}\|\} > tol_u$, então o valor de tol_w é momentaneamente diminuído (tipicamente dividido por 10) e as iterações em $\mathbf{w}$ repetidas. Neste trabalho utilizou-se $tol_u = 10^{-4}$. Essa segunda conferência é feita com a intenção de garantir que apenas as auto-soluções associadas a movimentos de bolha invíscida e movimentos de corpo rígido são extraídas do espectro de soluções da representação integral. Apenas esporadicamente a re-definição de tol_w foi necessária durante as simulações

realizadas. Vale comentar que mesmo com um chute inicial preciso tal que $\varepsilon_u < tol_u$, um processo iterativo em que a formulação padrão sem a remoção dos autovalores espúrios é diverge para $\lambda \geq 5$.

Em particular, quando $\lambda = 1$, a representação integral do escoamento sobre a superfície da gota reduz-se simplesmente camada potencial, de forma que

$$\mathbf{u}(\mathbf{x}_o) = \frac{2c_\lambda}{1+\lambda} Ca \mathbf{u}^\infty(\mathbf{x}_o) - \frac{c_\lambda}{4\pi(1+\lambda)} \int_S 2\kappa \mathbf{G}(\mathbf{x} - \mathbf{x}_o) \cdot \mathbf{n}(\mathbf{x}) dS(\mathbf{x}).$$

Assim sendo, nesse caso, a expressão para $\mathbf{u}(\mathbf{x}_o)$ é explícita e não há necessidade de processo iterativo de solução. Tipicamente, a aproximação inicial para $\mathbf{u}(\mathbf{x}_o)$ necessária quando $\lambda \neq 1$ é obtida desse caso.

3.4

Evolução da superfície da gota

A evolução da forma da superfície da gota é feita por um algoritmo de Euler aplicado à solução da equação de evolução (2-80). No entanto, o caráter lagrangeano da discretização espacial, tal que cada ponto da malha é uma partícula material, dá origem a um problema de instabilidade da malha. Mesmo que a forma da gota não varie mais, ainda há escoamento em sua superfície (como há dentro e fora da gota também). Assim sendo, os pontos de controle estão sempre em movimento, distorcendo os elementos até que a malha entre em colapso. A cura para esse comportamento patológico são os chamados métodos de relaxação. Tais métodos são procedimentos que buscam retirar a dependência temporal da posição dos nós sobre a gota, de forma que a malha não dependa mais da história do escoamento. Tanzosh *et al.* (1992) reportam esse problema e sugerem que a evolução dos pontos da malha seja feita usando, apenas, a componente normal à superfície do vetor velocidade. De fato, esse procedimento reduz significativamente a distorção da malha, mas não resolve o problema, sobretudo em simulações de longo tempo. Loewenberg & Hinch (1996) convectam os pontos de controle alterando a equação evolutiva para $d\mathbf{x}/dt = \mathbf{u} + \mathbf{\Pi}$, em que $\mathbf{\Pi}$ é uma velocidade tangencial arbitrariamente definida com o intuito de conter a distorção da malha. A velocidade tangencial $\mathbf{\Pi}$ é calculada localmente, de forma que o tempo de processamento necessário ao procedimento de relaxação é $\mathcal{O}(N_\Delta)$. Esse método evita que a malha se distorça durante simulações de longo tempo. Além disso, a definição de $\mathbf{\Pi}$ pode levar em consideração fatores como a curvatura média local, variações na área dos elementos, distância entre duas ou mais

gotas, velocidade sobre a superfície, concentração de surfactantes entre outros, caracterizando assim um método ativo de relaxação. Portanto, a recolocação dos pontos de controle sobre a malha pode privilegiar regiões de alta curvatura ou de altos gradientes de concentração de surfactantes, por exemplo. Zinchenko *et. al.* (1997) utilizam um algoritmo baseado na minimização de uma energia definida em função do comprimento das arestas da malha. Seu método é independente de qualquer parâmetro físico do problema (como curvatura local e concentração de surfactantes, por exemplo) e por isso é classificado como um método passivo. Na medida do possível, tende a manter a malha com elementos do mesmo tamanho. Segundo os autores, na sua forma otimizada, o procedimento é $\mathcal{O}(N_\Delta)$ apesar do método ser global.

Neste trabalho empregou-se um método similar ao de Loewenberg & Hinch (1996). Dessa forma, ao invés de se resolver a equação (2-80) utilizou-se

$$\frac{d\mathbf{x}}{dt} = (\mathbf{u} \cdot \mathbf{n})\mathbf{n} + \mathbf{\Pi}, \quad (3-9)$$

com o valor de $\mathbf{\Pi}$, para cada nó $\mathbf{x}_i$ da malha, dado por (Loewenberg & Hinch, 1996)

$$\mathbf{\Pi}(\mathbf{x}_i) = \psi^{-1} \frac{N_\Delta^{3/2}}{1 + \lambda} [\mathbf{I} - \mathbf{n}(\mathbf{x}_i)\mathbf{n}(\mathbf{x}_i)] \sum_j (C_{r1} + C_{r2}|\bar{\kappa}_j|^{3/2}) S_j^e(\mathbf{x}_i - \mathbf{x}_j), \quad (3-10)$$

em que ψ , C_{r1} e C_{r2} são constantes positivas, j é um índice que corre sobre os nós vizinhos do ponto de controle $\mathbf{x}_i$ e S_j^e a área do elemento à esquerda da aresta formada por $\mathbf{x}_i - \mathbf{x}_j$. A expressão (3-10) é definida de forma heurística e otimizações cuidadosas são desnecessárias, segundo os autores. O termo $\mathbf{I} - \mathbf{n}(\mathbf{x}_i)\mathbf{n}(\mathbf{x}_i)$ multiplicando o somatório garante que $\mathbf{\Pi}$ seja uma velocidade tangente à superfície. A constante C_{r1} controla a tendência da malha em manter os elementos com mesma área e C_{r2} controla a tendência da relaxação em concentrar pontos nas regiões de maior curvatura. Para dar sensibilidade à velocidade de relaxação à outros parâmetros é preciso adicionar termos similares. Por exemplo, se há a necessidade de concentrar pontos em regiões próximas de outras gotas, o termo $C_{r3}\|\mathbf{x}_i - \mathbf{x}_j\|/h(\mathbf{x}_j)$, em que $h(\mathbf{x}_j)$ é a distância do ponto $\mathbf{x}_i$ à gota mais próxima, deve ser incluído. A constante ψ está associada à magnitude de $\mathbf{\Pi}$. Neste trabalho foram utilizados $\psi = 300$, $C_{r1} = 10$ e $C_{r2} = 5$.

Sejam os pontos sobre a superfície da gota em determinado instante p denominados por $\mathbf{x}^p$, N_t o número de passos de tempo e $\mathbf{m}(\mathbf{x}^p) = (\mathbf{u} \cdot \mathbf{n})\mathbf{n} + \mathbf{\Pi}$. A utilização de um método de Euler de segunda ordem para a evolução da equação (3-9), em que uma condição inicial $\mathbf{x}^0$ é conhecida, segue o algoritmo (3-11).

A partir de $p = 0$ até $p = N_t$ faça:

$$\left[\begin{array}{l} 1. \mathbf{k}_1 \leftarrow \mathbf{m}(\mathbf{x}^p) \\ 2. \mathbf{x}' \leftarrow \mathbf{x}^p + \Delta t \mathbf{k}_1 \\ 3. \mathbf{k}_2 \leftarrow \mathbf{m}(\mathbf{x}') \\ 4. \mathbf{x}^{p+1} \leftarrow \mathbf{x}^p + \frac{1}{2}(\mathbf{k}_1 + \mathbf{k}_2). \end{array} \right. \quad (3-11)$$

Vale comentar que nos passos 1 e 3 do algoritmo anterior é preciso calcular o vetor normal, a curvatura, a velocidade $\mathbf{u}$ e a velocidade de relaxação $\mathbf{\Pi}$. O passo de tempo adimensional é ajustado levando-se em conta os parâmetros físicos do problema, tal que $\Delta t = \min \left\{ \frac{\Delta x}{(\lambda + 1)Ca}, \omega \Delta x, 10^{-2} \right\}$, em que $\Delta x = \sqrt{4\pi/N_\Delta}$, Ca é o número de capilaridade e ω a frequência de excitação do escoamento, quando for o caso. Se a evolução temporal for realizada simplesmente usando o algoritmo (3-11) não há garantia de que o processo de relaxação seja completamente realizado no sentido de tornar a malha independente da história do escoamento. Para isso, os pontos sobre a superfície são submetidos a um novo processo de “evolução temporal” em que apenas a velocidade de relaxação é utilizada. Definem-se então $\mathbf{m}^*(\mathbf{x}) = \mathbf{\Pi}$ e o parâmetro de evolução da relaxação t^* . Esse processo é realizado em um contexto virtual, sendo que seus incrementos Δt^* são fictícios e não somam no avanço temporal real da simulação. Para sinalizar o fim da relaxação da malha utilizou-se a derivada em relação à t^* do desvio padrão das áreas dos elementos da malha. O algoritmo da relaxação pode ser descrito como se segue

$$\left[\begin{array}{l} 1. \mathbf{k}_1 \leftarrow \mathbf{m}^*(\mathbf{x}, t) \\ 2. \mathbf{x}' \leftarrow \mathbf{x} + \Delta t^* \mathbf{k}_1 \\ 3. \mathbf{k}_2 \leftarrow \mathbf{m}^*(\mathbf{x}') \\ 4. \mathbf{x} \leftarrow \mathbf{x} + \frac{1}{2}(\mathbf{k}_1 + \mathbf{k}_2) \\ 5. \text{ Se } |std'(S^e)| < tol_r \text{ então } \rightarrow \mathbf{FIM} \\ 6. \text{ Retorna ao passo 1} \end{array} \right. \quad (3-12)$$

em que $std'(S^e)$ representa a derivada do desvio padrão em relação à t^* . Utilizou-se $\Delta t^* = \Delta t$, apesar de que aparentemente estes são parâmetros independentes. A tolerância utilizada foi $tol_r = 10^{-3}$. Qualitativamente, o resultado do processo de relaxação pode ser observado na figura (3.8). Em (3.8a) e (3.8b) são mostradas duas vistas de uma malha sobre uma gota que deformou durante determinado período sem relaxação (apenas utilizando a projeção normal da velocidade). Em (3.8c) e (3.8d) são mostradas as mesmas

vistas da malha resultante da relaxação da primeira utilizando-se o processo descrito em (3-12). Podemos observar que a malha relaxada apresenta uma distribuição de pontos mais uniforme sobre a superfície. Para obter esse resultado foram necessárias 250 iterações. Nesse exemplo, a relaxação atuou

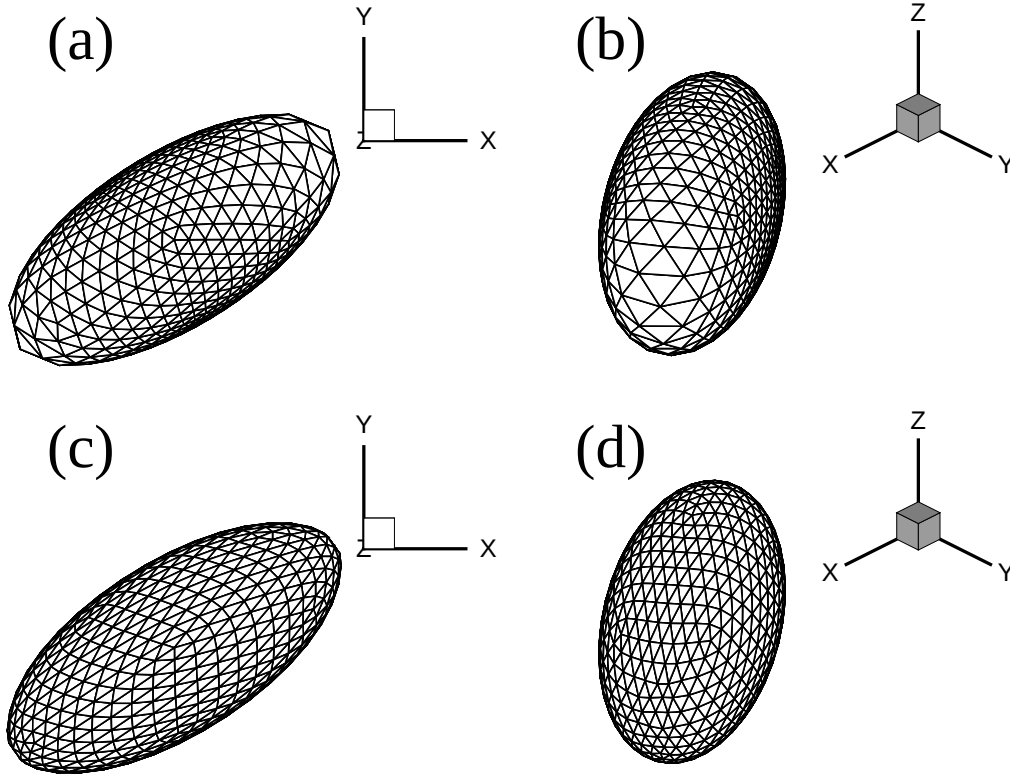


Figura 3.8: Malha sobre uma gota: $f = 8$, $N_{\Delta} = 1280$. (a) e (b) Vistas da malha resultante após breve evolução temporal sem relaxação ($Ca = 1$, $\lambda = 1$, após 100 iterações). (c) e (d) malha resultante depois da relaxação.

sobre uma malha que inicialmente evoluiu deformando sem relaxação alguma. Em contraste, nas simulações realizadas neste trabalho, a cada passo de tempo a malha passa pelo processo de relaxação. Para mostrar o efeito desse procedimento três malhas foram geradas nas mesmas condições da malha da figura (3.8). A primeira relaxava apenas pela adição de Π à $(\mathbf{u} \cdot \mathbf{n})\mathbf{n}$, como em (3-11). As demais, além disso, passavam pelo processo (3-12) com diferentes números de iterações (para isso o passo 5 é suprimido e o número de iterações controlado), a cada passo de tempo. Finalmente, as malhas resultantes foram completamente relaxadas e o desvio padrão da área dos elementos da malha, $sdt(S^e)$, monitorado em função do número iterações, n_{relax} , desse último processo de relaxação. O resultado dessa comparação é mostrado na figura (3.9). Observa-se em (3.9) que usando-se 10 iterações de relaxação por passo de tempo a distribuição de pontos sobre a malha, praticamente, não é alterada pela relaxação extra. Isso sugere que o caráter lagrangeano da malha é anulado

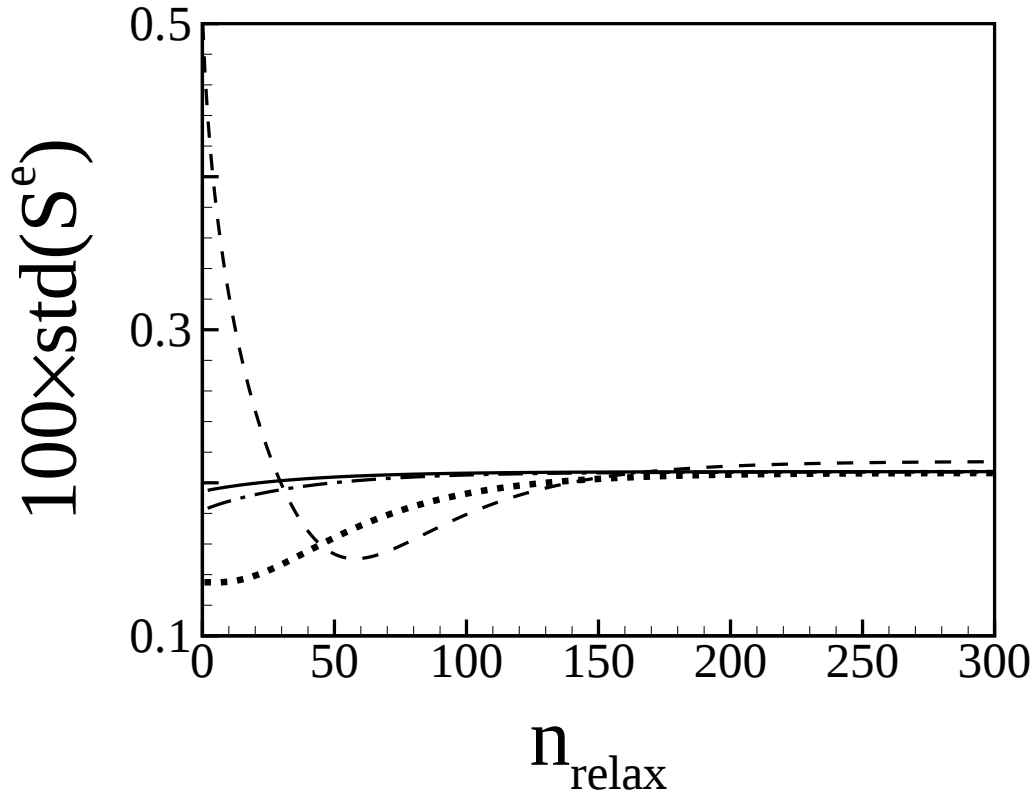


Figura 3.9: Desvio padrão das áreas dos elementos da malha em função do número de iterações do processo de relaxação. Linha tracejada: malha inicial gerada sem relaxação; Linha pontilhada: malha inicial gerada pela evolução usando apenas (3-11); Linha de traços e pontos: malha inicial gerada com 5 iterações de relaxação extras; Linha cheia: malha inicial gerada com 10 iterações de relaxação extras;

pelo processo de relaxação e que a posição dos nós sobre a superfície é independente da história do escoamento. Na prática, quando o critério de parada do passo 5 do algoritmo (3-12) é utilizado, o número de iterações de relaxação raramente ultrapassa 4.

Chama a atenção na figura (3.9) que $std(S^e)$ estabiliza-se em um valor acima do seu valor mínimo ($std(S^e)_{min} = 1,3 \times 10^{-3}$). Isso indica que a relaxação está levando a malha para uma configuração tal que distribuição de tamanhos dos elementos não é a mais homogênea possível. Esse fato está associado à tendência do processo em concentrar pontos nas regiões de maior curvatura. Para demonstrar isso, relaxou-se a malha de (3.8a) com $C_{r2} = 0$. Dessa maneira, o único efeito da relaxação é a homogeneização do tamanho dos elementos. Na figura (3.10) são comparadas as evoluções do desvio padrão de S^e , como função de n_{relax} , para $C_{r2} = 0$ e $C_{r2} = 5$. Verifica-se que no caso de $C_{r2} = 0$ a estabilização da malha acontece no valor mínimo de $std(S^e)$, indicando que desta forma, o procedimento distribui os pontos de maneira a

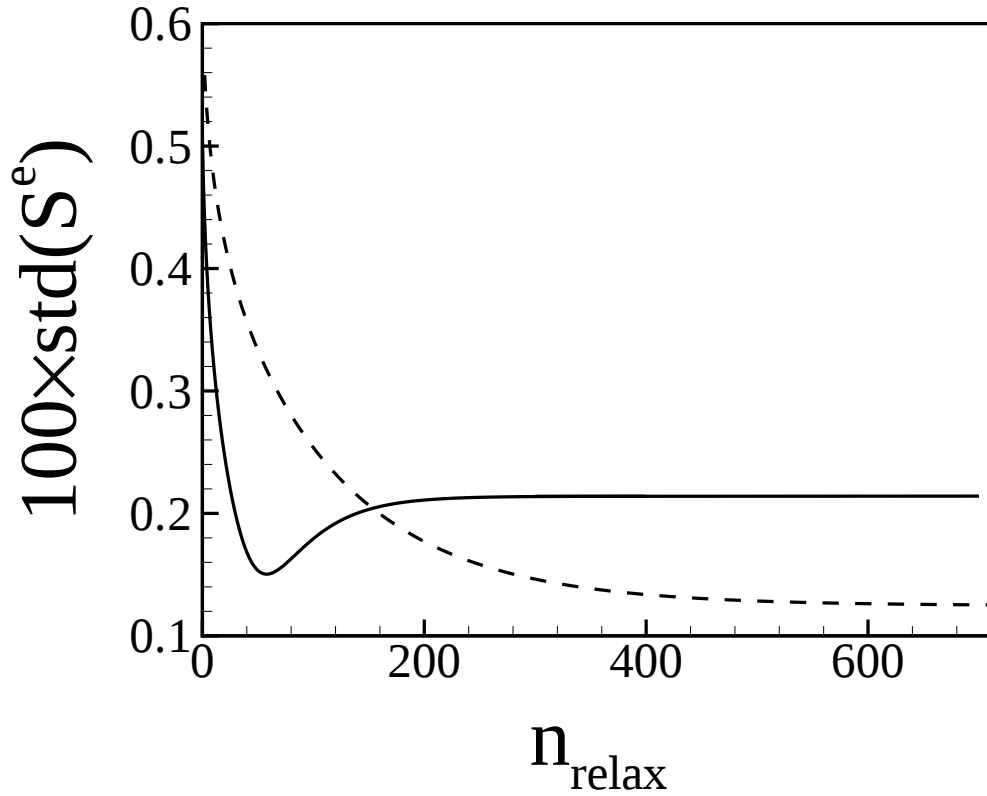


Figura 3.10: Evolução de $\text{std}(S^e)$ durante a relaxação da malha na figura (3.8). Linha cheia: $C_{r2} = 5$; Linha tracejada: $C_{r2} = 0$.

tornar malha a mais homogênea possível quanto ao tamanho dos elementos. Observa-se também que essa estabilização precisa de mais iterações para ser obtida. Na figura (3.11) observa-se que a diferença entre as malhas finais obtidas é discreta, mas perceptível, havendo maior concentração de pontos na região de maior curvatura no caso (b), em que $C_{r2} = 5$.

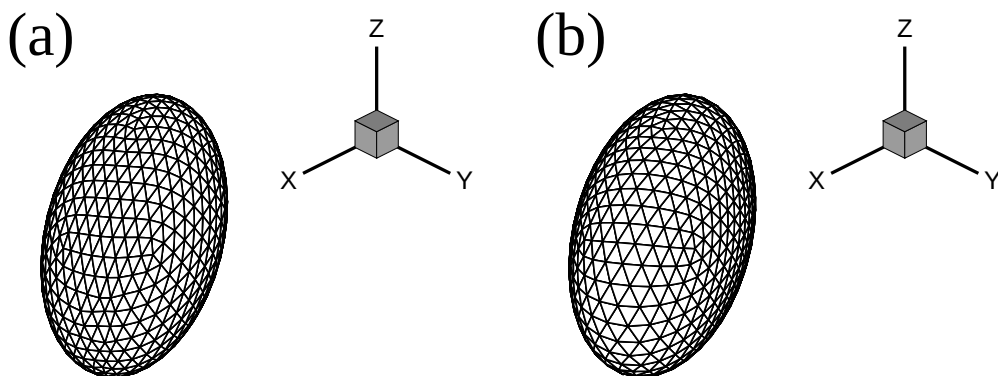


Figura 3.11: Malhas finais obtidas após a relaxação. (a) $C_{r2} = 5$, maior concentração de pontos nas regiões de curvatura mais intensa; (b) $C_{r2}=0$, distribuição homogênea de tamanhos de elementos.

A inflexão no ponto de mínimo do gráfico de $\text{std}(S^e)$ para o caso de

$C_{r2} = 5$ na figura (3.10) provavelmente acontece pelo fato da configuração da malha em (3.10b) ser intermediária entre as configurações da malha em (3.8a) e (3.11a). Dessa forma, antes de se estabilizar, a malha relaxando com $C_{r2} = 5$ passa por uma configuração tal que $std(S^e)$ é próximo do mínimo. No entanto, o processo continua movendo os pontos para promover o refinamento nas regiões de curvatura acentuada, o que eleva o valor de $std(S^e)$. Vale ainda comentar que para $C_{r2} = 0$ o procedimento de relaxação proposto nesse trabalho atua de forma semelhante ao método de Zinchenko *et. al.* (1997).

Otimizações em relação ao tempo de processamento podem ser realizadas no procedimento de relaxação. Uma possibilidade é não relaxar em todo passo de tempo. Outra seria procurar formas otimizadas para a expressão de Π ou mesmo fazer um estudo sistemático do valor das constantes C_{r1} , C_{r2} e ψ . No entanto, para uma simulação com uma malha de 1280 elementos ($f = 8$), com $\lambda = 1$, a relaxação consome menos de 0,2% do tempo total de processamento. Portanto, otimizações nesse sentido são desinteressantes.

3.5

Geometria e tensão induzida pela gota

As medidas associada à geometria da gota e a tensão adicional da fase dispersa da emulsão são informações obtidas em pós-processamento. Nessa seção, as principais técnicas utilizadas para obter essas quantidades são exploradas.

3.5.1

Quantidades geométricas da gota

Em um trabalho pioneiro Taylor (1934) define uma quantidade associada à geometria da gota em termos dos semi-eixos da projeção da gota em um plano paralelo à direção do escoamento imposto sobre a gota, $\mathbf{u}^\infty$. De acordo com a notação definida na figura (3.12), Taylor (1934) propôs

$$D_T = \frac{L - B}{L + B}, \quad (3-13)$$

de forma que para uma esfera $D_T = 0$ e à medida que gota se deforma, $D_T \rightarrow 1$. Para o cálculo da deformação de Taylor, D_T , a partir de uma superfície descrita por uma malha de elementos triangulares o procedimento adotado inicia com a determinação da orientação da gota, θ , em relação à direção do escoamento. Para de prover uma forma geral para esse fim, o tensor de inércia da gota é calculado usando-se (Kennedy *et. al.*, 1994)

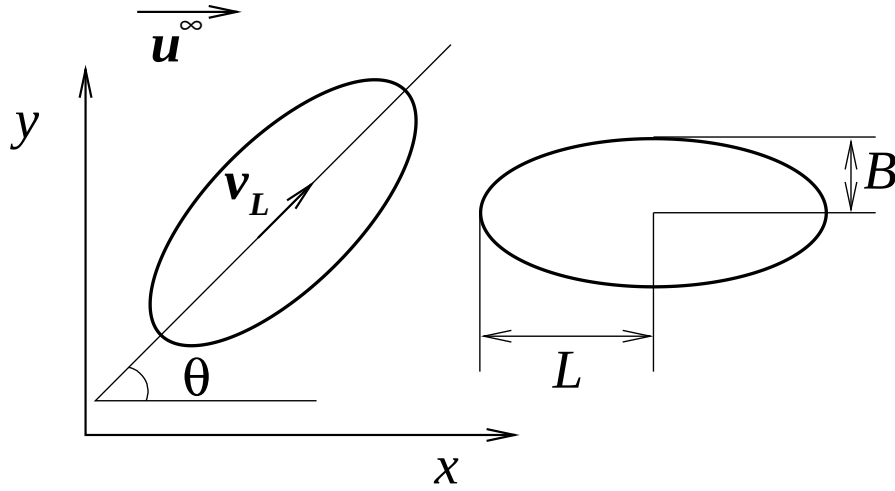


Figura 3.12: Orientação e semi-eixos principais da gota projetada sobre o plano paralelo à direção do escoamento $\mathbf{u}^\infty$.

$$\mathcal{D} = \frac{1}{5} \int_S [(\mathbf{x} \cdot \mathbf{x})\mathbf{I} - \mathbf{x}\mathbf{x}](\mathbf{x} \cdot \mathbf{n}) dS. \quad (3-14)$$

O menor autovalor de $\mathcal{D}$ está associado à direção que oferece menor resistência à rotação. Sendo assim, seu autovetor é paralelo ao vetor unitário $\mathbf{v}_L$, mostrado na figura (3.12). Dessa maneira, e sendo $\mathbf{e}_x$ e $\mathbf{e}_y$ as direções dos eixos ordenados, a orientação da gota é calculada pelo algoritmo (3-15).

$$\begin{cases} 1. & \theta' \leftarrow \cos^{-1}(\mathbf{v}_L \cdot \mathbf{e}_x) \\ 2. & \text{Se } \mathbf{v}_L \cdot \mathbf{e}_y < 0 \text{ então } \theta \leftarrow -\theta' \\ 3. & \text{Senão } \theta \leftarrow \theta' \end{cases} \quad (3-15)$$

Uma vez determinado $\mathbf{v}_L$, a direção do semi-eixo associado ao comprimento B da gota pode ser obtida diretamente por $\mathbf{v}_B = \mathbf{v}_L \times \mathbf{e}_z$ em que $\mathbf{e}_z$ é o vetor unitário da direção normal ao plano Oxy da figura (3.12). Para a determinação do comprimento dos semi-eixos L e B uma pesquisa sequencial é realizada em todos os pontos da malha buscando aqueles cuja projeção no plano Oxy e nas direções $\mathbf{v}_L$ e $\mathbf{v}_B$ são as maiores. A distância desses pontos à projeção no plano Oxy do centróide da malha definem os comprimentos L e B .

Em algumas teorias de pequenas deformações (Oliveira *et. al* (2005) & seção 4.1), a descrição da superfície da gota é definida em função de um tensor simétrico de distorção $\mathbf{A}$, como na equação (4-11), em que

$$S(t) = r(t) - (1 + \mathbf{A} : \mathbf{x}\mathbf{x}),$$

$r(t) = \sqrt{\mathbf{x} \cdot \mathbf{x}}$ e $\mathbf{x}$ representa a coordenada do ponto sobre a superfície adimensionalizada pelo raio da gota não-deformada a . O vínculo $S(t) = 0$ deve ser satisfeito, de sorte que $r(t) = 1 - \mathbf{A} : \mathbf{x}\mathbf{x}$. Nessa teoria, toda a geometria da superfície é função das componentes do tensor distorção $\mathbf{A}$. Deseja-se, portanto, determinar quais seriam essas componentes para uma superfície descrita por uma malha computacional, resultante das simulações. Para isso, introduzimos uma representação para o tensor $\mathbf{A}$ usando-se uma base para um espaço de tensores de segunda ordem simétricos, tal que

$$\mathbf{A} = C_i \mathbf{T}_i, \quad (3-16)$$

em que $\mathbf{T}_i$, $i = 1, \dots, 6$ são os diádicos $\mathbf{T}_1 = \mathbf{e}_1\mathbf{e}_1$, $\mathbf{T}_2 = \mathbf{e}_2\mathbf{e}_2$, $\mathbf{T}_3 = \mathbf{e}_3\mathbf{e}_3$, $\mathbf{T}_4 = \mathbf{e}_1\mathbf{e}_2 + \mathbf{e}_2\mathbf{e}_1$, $\mathbf{T}_5 = \mathbf{e}_1\mathbf{e}_3 + \mathbf{e}_3\mathbf{e}_1$ e $\mathbf{T}_6 = \mathbf{e}_2\mathbf{e}_3 + \mathbf{e}_3\mathbf{e}_2$. Na equação (3-16) vale a convenção soma de Einstein (Aris, 1962). Dessa forma, definindo $f = 1 - r(t) = \mathbf{A} : \mathbf{x}\mathbf{x}$, podemos escrever

$$f = C_i \mathbf{T}_i : \mathbf{x}\mathbf{x} = C_i Y_i, \quad (3-17)$$

em que $Y_i = \mathbf{T}_i : \mathbf{x}\mathbf{x}$. Assim sendo, a distorção em relação à forma esférica da gota f fica escrita em termos dos coeficientes lineares C_i . Define-se agora

$$\beta_i = \int_S f Y_i dS. \quad (3-18)$$

A quantidade β_i pode ser calculada a partir de uma superfície genérica S , proveniente de uma simulação numérica, por exemplo. Por outro lado, utilizando a equação (3-17) podemos escrever

$$C_j \int_S Y_j Y_i dS = \int_S f Y_i dS. \quad (3-19)$$

A equação (3-19) define um sistema linear na forma $[\mathcal{D}]\{\mathbf{C}\} = \{\beta\}$, em que $[\mathcal{D}]$ é uma matriz 6×6 cujas componentes são dadas por $\mathcal{D}_{ji} = \int_S Y_j Y_i dS$, $\{\beta\}$ uma matrix 6×1 de componentes dadas por (3-18) e cuja solução $\{\mathbf{C}\}$ são os coeficientes lineares C_j que melhor representam as componentes de $\mathbf{A}$ da superfície S . Esse procedimento é equivalente à um ajuste pela técnica dos resíduos ponderados. Usando essa nova abordagem teórica, é possível comparar diretamente os resultados obtidos a partir das teorias desenvolvidas no capítulo 4 com resultados numérico gerados pelas simulações.

3.5.2

Quantidades reológicas da emulsão

Em relação às grandezas geométricas associadas à forma da gota, as quantidades reológicas podem ser determinadas de forma mais simples. Isso

acontece porque uma forma integral para o cálculo da tensão induzida pela fase dispersa da emulsão está disponível nas equações (2-130) e (2-134), ou (2-130) e (2-136). No caso de uma formulação monodispersa, nenhum desenvolvimento adicional precisa ser feito para a determinação de $\tilde{\sigma}^d$ que é simplesmente dado por (2-135). Já para o caso polidisperso diluído, em geral, o procedimento envolve o cálculo das integrais associadas à distribuição de raios de gotas. Esse processo pode ser realizado sem dificuldades desde que se conheça $\tilde{\mathbf{B}}$ ou $\tilde{\mathbf{S}}$ como função de Ca . Se não houver uma expressão fechada, um tabelamento da função $\tilde{\mathbf{B}}(Ca)$ ou de $\tilde{\mathbf{S}}(Ca)$ deve ser empregado. Nesse caso, uma integração pela regra do trapézio ou por outro procedimento numérico pode ser usado.