

## 6

### Fluidos viscoplásticos

Nesse capítulo, apresentaremos um novo método de animação visualmente realística de fenômenos físicos como deformações plásticas e mudanças de fase sólida-líquida que ocorrem em materiais viscoplásticos, tais como: metal, plástico, cera, polímero, argila e lava. Essa técnica consiste na representação desses materiais como um fluido não-newtoniano que varia entre o estado sólido e líquido, e de alta para baixa viscosidade dependendo de sua própria temperatura ou de uma força externa aplicada sobre o material.

O primeiro objetivo desse capítulo é o de simular com fidelidade os efeitos viscosos de um fluido viscoplástico, onde a viscosidade de um fluido descreve a maneira que a velocidade do fluido se dissipa. Assim, criamos um modelo SPH baseado na formulação física de Fluido Newtoniano Generalizado proposta por Mendes *et al.* [45] (veja Seção 6.1). Nessa formulação, a viscosidade do material é dada através de uma função não-linear em termos da intensidade de uma tensão de deformação aplicada sobre ele.

O próximo objetivo consiste na simulação de materiais que variam entre o estado sólido e líquido devido a variações térmicas. Para simular essas transições criamos uma simples dependência da viscosidade com a temperatura do material. Na Seção 6.2, mostraremos com detalhes como é feita essa transição, assim como uma nova aproximação SPH da equação do calor.

Mais adiante, na Seção 6.3, apresentaremos uma forma geométrica de impor as condições de fronteira do fluido. Em seguida, na Seção 6.4, discutiremos alguns aspectos numéricos a fim de garantir a estabilidade do sistema e apresentaremos uma nova variante do SPH conhecida como XSPH. E finalmente nas Seções 6.5 e 6.6, mostraremos a implementação e os resultados do nosso método, respectivamente.

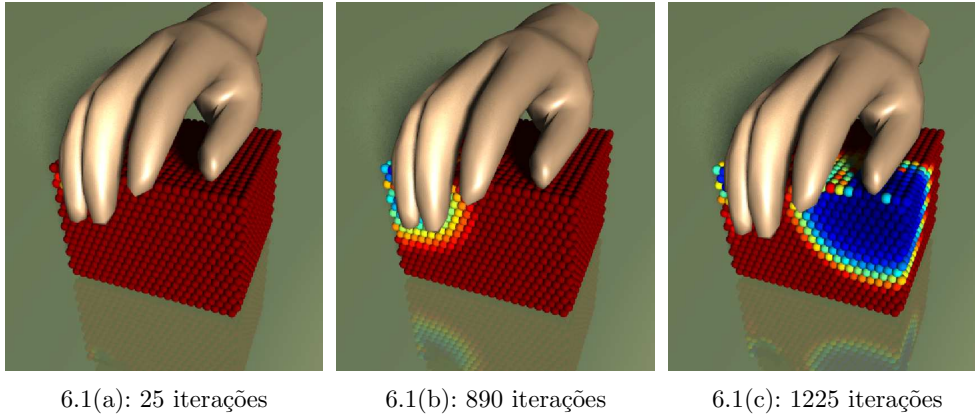


Figura 6.1: Modelo de Fluido Newtoniano Generalizado: o mapa de cores representa a viscosidade de cada partícula, variando de baixa (azul) para alta viscosidade (vermelho). Note o salto da viscosidade criado pela força aplicada pela mão, quanto maior a força menor será a viscosidade do fluido.

### 6.1 Fluido Newtoniano Generalizado

No modelo físico de Fluido Newtoniano Generalizado, para cada partícula  $i$  do sistema, o tensor extra-tensão  $\mathbf{S}_i$  é representado matematicamente da seguinte forma:

$$\mathbf{S}_i = \eta(D_i) \mathbf{D}_i, \quad \text{com} \quad D_i = \sqrt{\frac{1}{2} \text{traço}(\mathbf{D}_i)^2} \quad \text{e}$$

$$\eta(D_i) = (1 - \exp[-(J+1)D_i]) \left( D_i^{n-1} + \frac{1}{D_i} \right), \quad (6-1)$$

onde o tensor taxa de deformação  $\mathbf{D}_i$  é dado pela equação (4-45).

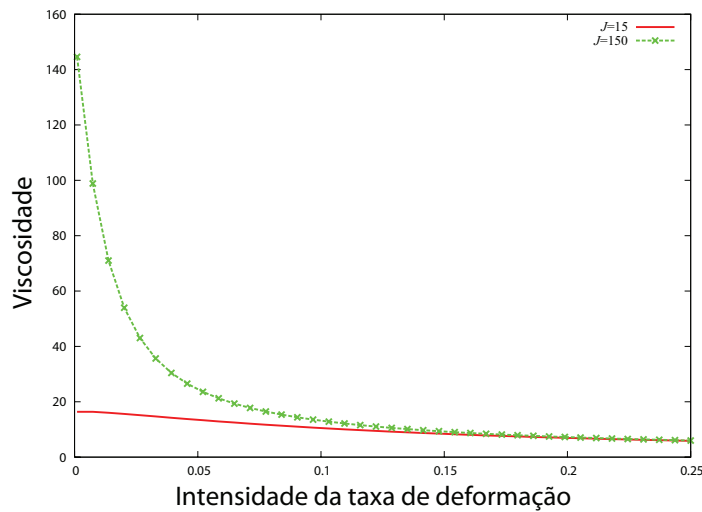


Figura 6.2: Comportamento viscoplástico do fluido.

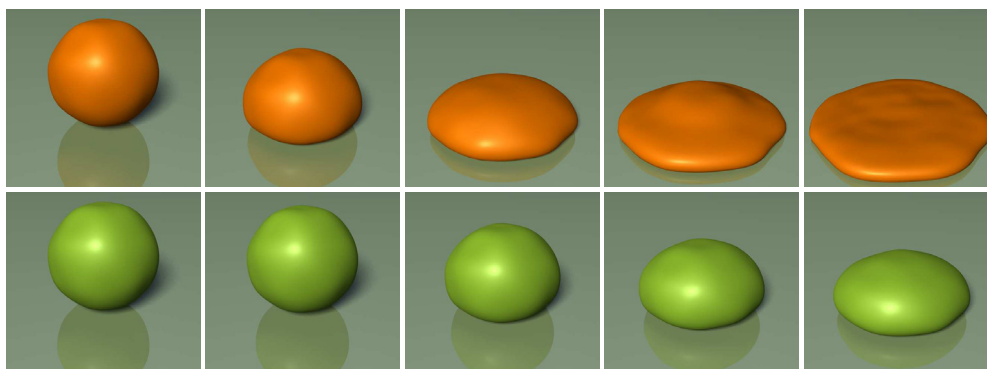


Figura 6.3: Variações da viscosidade de um fluido viscoplástico inicializado numa esfera com 1200 partículas. Acima, reduzimos a viscosidade do fluido usando  $J = 15$  e abaixo aumentamos a viscosidade com  $J = 150$ .

Essa formulação modela a viscosidade  $\eta$  como uma função não-linear em termos da intensidade da taxa de deformação  $D_i$  (veja Figura 6.2), sendo inversamente proporcional à tensão de deformação aplicada ao material (veja Figura 6.1). A função de viscosidade  $\eta$  depende apenas de dois parâmetros reológicos: o índice de comportamento do escoamento  $n$  e o termo  $J$ , conhecido como *jump number*.

Para fluidos viscoplásticos, Mendes representou de forma concisa vários parâmetros reológicos, tais como a tensão limite do escoamento e o índice de consistência, através de  $J$ . A viscosidade do fluido é controlada diretamente pela constante  $J$ , ou seja, quanto maior for o valor de  $J$  maior será a viscosidade do fluido (veja Figura 6.3). Note que, através da Figura 6.2 quando  $J = 15$  o fluido possui um comportamento newtoniano. Para simular o comportamento viscoplástico de um fluido o valor de  $n$  deve ser entre 0 e 1, logo em nossas simulações fixamos  $n = \frac{1}{2}$ .

## 6.2

### Transição de fase

A simulação de objetos que derretem e solidificam é uma tarefa delicada, pois é necessária a variação da viscosidade durante a transição de fase, de acordo com as propriedades do material. Em particular, nesse capítulo concentraremos na transição induzida pela temperatura entre as fases sólida e líquida. Modelamos essa transição variando a viscosidade de acordo com a temperatura de cada partícula de fluido. A variação da temperatura em relação ao tempo é determinada pela *equação do calor*. Essa equação governa a difusão térmica de um material, transferindo energia térmica de uma região de alta temperatura para uma região de baixa temperatura. A equação do calor é descrita através da temperatura  $T$  e do coeficiente de difusão térmica  $k$ :

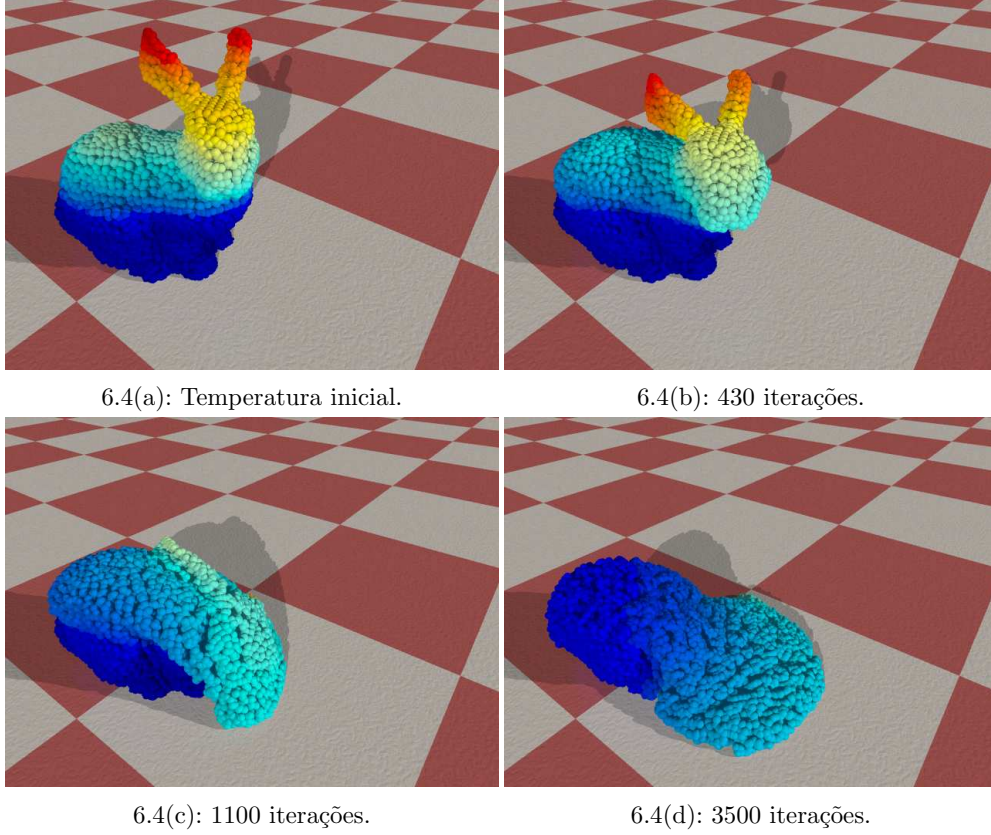


Figura 6.4: Temperatura das 10188 partículas do modelo Stanford Bunny: as regiões em azul escuro estão abaixo do ponto de fusão, e assim permanecem sólidas.

$$\frac{dT}{dt} = k \nabla^2 T. \quad (6-2)$$

No derretimento, a temperatura de algumas partes do objeto aumenta até alcançar o seu ponto de fusão no qual o objeto se torna líquido (veja Figura 6.4). Nos fluxos de lava, a temperatura pode decrescer abaixo do seu ponto de fusão fazendo com que a lava se solidifique e altere a topologia do terreno inicial (Figura 6.5). Em ambos os casos, o jump number  $J$  decresce com a temperatura. Aproximamos essa dependência através de uma combinação linear em termos da temperatura:

$$J(T) = (1 - u)J_{max} + uJ_{min}, \quad \text{com} \quad u = \frac{T - T_{min}}{T_{max} - T_{min}}. \quad (6-3)$$

Perceba que a função de viscosidade (6-1) decresce quando a temperatura aumenta e vice-versa. Em nossas simulações, assumimos que os objetos são homogêneos, isto é, possuem um coeficiente de difusão térmica constante.

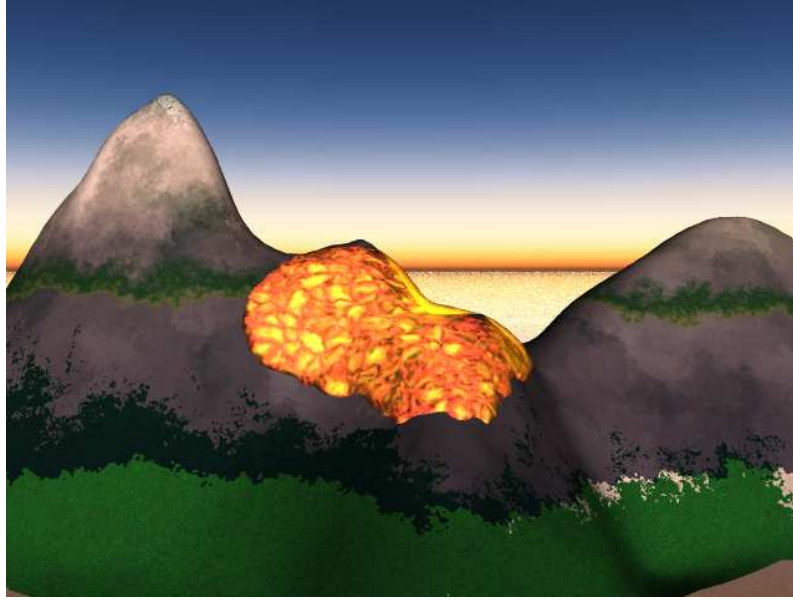


Figura 6.5: Escoamento de lava no Pão de Açúcar. A simulação da lava é feita utilizando 10137 partículas.

### 6.2.1

#### Aproximação SPH do operador laplaciano

A equação do calor (6-2), que governa a transição entre as fases sólida e líquida, requer uma aproximação por partículas do laplaciano da temperatura  $\nabla^2 T_i$ . As derivadas de segunda ordem podem ser aproximadas utilizando a convolução SPH usual através do laplaciano do núcleo para cada partícula  $i$ :

$$\nabla^2 T_i = \sum_{j \in N(\mathbf{x}_i)} \frac{m_j}{\rho_j} T_j \nabla_i^2 W_{ij}. \quad (6-4)$$

Porém, a equação (6-4) possui algumas desvantagens. Primeiro, essa aproximação é muito sensível à desordem de partículas. Segundo, a transferência de calor de uma partícula  $i$  para uma partícula  $j$  pode ser positiva ou negativa devido à mudança de sinal da derivada de segunda ordem do núcleo (veja Figura 4.1). Por outro lado, a física nos diz que uma partícula quente deve transferir calor para uma partícula fria sem se importar de que maneira é feita a separação delas. Outra desvantagem é que essa expressão não resulta em conservação de energia térmica num processo adiabático.

Para evitar esses problemas, Müller *et al.* [51] adicionaram em seu modelo um núcleo com derivada de segunda ordem positiva exclusivo para o operador laplaciano. Entretanto, além desse modelo ser errado do ponto de vista conceitual ele também introduz mais avaliações no sistema. Por essas razões, usamos uma aproximação do operador laplaciano proposta por Cleary & Monaghan [13] que envolve apenas derivadas de primeira ordem. Essa aproximação

é deduzida a partir da integral

$$I = \int (T(\mathbf{x}) - T(\mathbf{x}')) F(\mathbf{x} - \mathbf{x}') d\mathbf{x}', \quad (6-5)$$

tal que a função  $F$  é definida por

$$\mathbf{x}F(\mathbf{x}, h) = \nabla W(\mathbf{x}, h). \quad (6-6)$$

Expandindo o termo  $T(\mathbf{x}')$  no integrando de (6-5) em série de Taylor em torno de  $\mathbf{x}$ , até os termos de segunda ordem, temos

$$I = \nabla^2 T(\mathbf{x}) + r(h^2). \quad (6-7)$$

Portanto, tomando a forma SPH da integral (6-5) para uma partícula  $i$  e escolhendo

$$F(\mathbf{x}, h) = \frac{\mathbf{x} \cdot \nabla W(\mathbf{x}, h)}{\|\mathbf{x}\|^2}, \quad (6-8)$$

finalmente temos

$$\nabla^2 T_i = \sum_{j \in N(\mathbf{x}_i)} \frac{m_j}{\rho_j} (T_i - T_j) \frac{\mathbf{x}_{ij} \cdot \nabla_i W_{ij}}{|\mathbf{x}_{ij}|^2}. \quad (6-9)$$

Devido ao fato de  $\mathbf{x}_{ij} \cdot \nabla_i W_{ij} \leq 0$ , a expressão (6-9) tem a propriedade de que se  $T_i > T_j$ , então o fluxo de calor será realizado da partícula  $i$  para  $j$  e vice-versa.

### 6.3

#### Condição de fronteira

Em várias aplicações do método SPH, as fronteiras rígidas (terrenos ou paredes) são modeladas através do uso de partículas de fantasmas (Seção 5.3), as quais adicionam uma nova força interação nas partículas de fluido. Apesar de elegante, esse método aumenta consideravelmente o número de partículas do sistema, é computacionalmente caro, requer alto consumo de memória e a modelagem de geometrias complexas se torna difícil. Por essas razões,

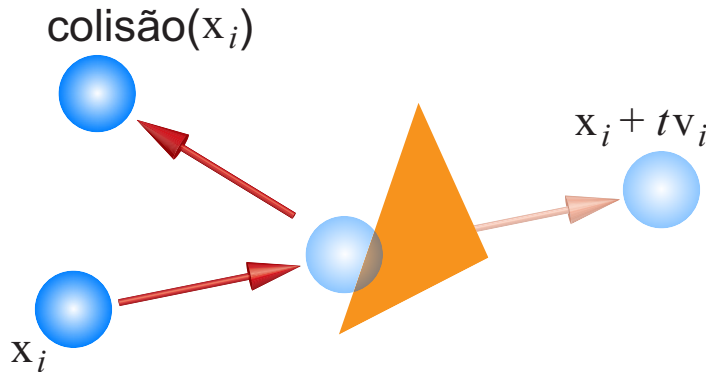
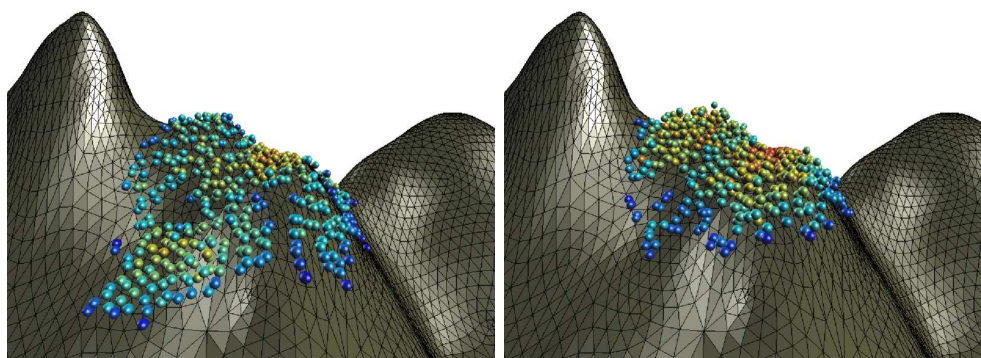


Figura 6.6: Resposta da colisão partícula  $\times$  triângulo.





6.7(a): Colisão suave.

6.7(b): Colisão não-eskorregadia.

Figura 6.7: Simulação de um escoamento de lava após 1910 iterações com colisão suave (esquerda) e com colisão não-eskorregadia (direita), com os mesmo parâmetros: 545 partículas e 9566 triângulos na fronteira. O mapa de cores representa a densidade de cada partícula. A colisão suave faz a lava deslizar mais rapidamente.

representamos explicitamente a fronteira do domínio utilizando uma malha triangular tal que as condições de fronteira são impostas através de um critério puramente geométrico, dado por um teste de colisão. O teste de colisão entre as partículas (esferas) e os triângulos orientados da fronteira é feito em duas etapas: detecção da colisão e a atualização do novo estado da partícula (resposta).

### 6.3.1

#### Resposta da colisão

A resposta do teste de colisão consiste no cálculo de uma nova posição e velocidade da partícula. Podemos inserir dissipação na velocidade de resposta simplesmente modificando a componente normal e tangencial da velocidade inspirada numa lei de Snell (Figura 6.6). Numa colisão perfeitamente elástica é invertida apenas a componente normal. Para colisões suaves, a componente tangencial e normal podem ser escaladas por coeficientes constantes entre 0 e 1, o coeficiente que dissipa a velocidade tangencial é chamado de *coeficiente de fricção* enquanto o que dissipa a velocidade normal é conhecido como *coeficiente de restituição*. Nos casos em que ambos os coeficientes são 0 a colisão é dita *não-eskorregadia* (Figura 6.7).

### 6.3.2

#### Detecção da colisão

Com objetivo de acelerar a detecção de uma eventual colisão entre as partículas do sistema com os triângulos da fronteira, armazenamos os triângulos no mesmo grid que utilizamos para fazer a busca de partículas

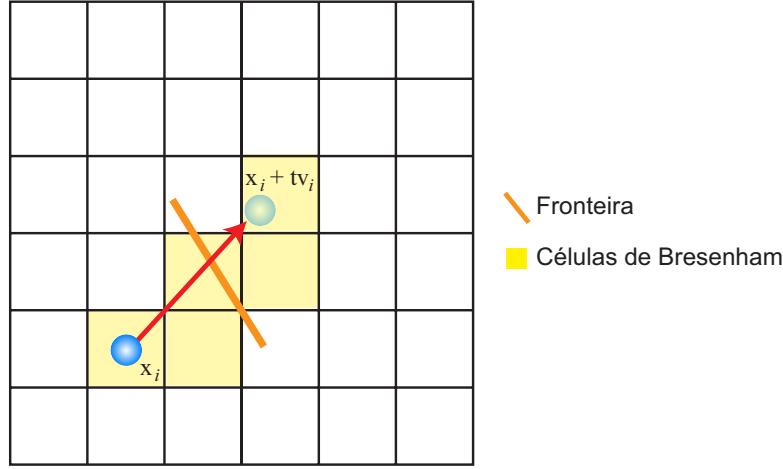


Figura 6.8: O teste de colisão realiza a busca de triângulos nas células do grid através do algoritmo de linhas de Bresenham.

vizinhas (Seção 5.1). Cada célula do grid de busca além de possuir uma lista de partículas vizinhas também possui uma lista de triângulos  $\mathcal{T}$ , para cada triângulo da fronteira fazemos o teste de intersecção com as células (cubos) do grid utilizando o algoritmo de Akenine–Möller [2]. Se o triângulo intersectar a célula logo ele é inserido na lista  $\mathcal{T}$  dessa célula. Apesar da possibilidade de um triângulo aparecer em várias células, esse armazenamento garante um tempo constante de busca para cada posição. Após a atualizar cada lista de triângulos  $\mathcal{T}$  nas células do grid, o teste de colisão é realizado de forma semelhante ao algoritmo de linhas de Bresenham [21] num grid 3D. O algoritmo de Bresenham seleciona incrementalmente algumas células do grid na tentativa de reproduzir a trajetória da partícula  $i$  através da linha que conecta os pontos  $\mathbf{x}_i$  e  $\mathbf{x}_i + \Delta t \mathbf{v}_i$ , o teste de colisão é feito somente nos triângulos das células selecionadas pelo algoritmo (Figura 6.8).

O teste de colisão entre partícula e triângulo é feito através de uma versão modificada de [29] e adaptada à nossa simulação. Para cada partícula  $i$  do sistema, utilizamos a informação da direção da velocidade  $\mathbf{v}_i$  para criar um eficiente teste de rejeição de triângulos através de uma simples verificação de sinal. Para cada triângulo  $T$  da lista  $\mathcal{T}$ , calculamos a sua normal  $\mathbf{N}_T$  através do produto vetorial de suas arestas e em seguida fazemos o teste  $\mathbf{v}_i \cdot \mathbf{N}_T$ . Se caso  $\mathbf{v}_i \cdot \mathbf{N}_T \geq 0$ , o triângulo  $T$  é descartado do teste de colisão. Caso contrário, a próxima etapa do teste de colisão consiste no cálculo do tempo de colisão  $t_I$  da partícula  $i$  com o plano determinado pelo triângulo  $T$  (Figura 6.9(a)). O tempo  $t_I$  é calculado utilizando a projeção:

$$t_I = \frac{\|(\mathbf{x}_i - \mathbf{V}_T) \cdot \mathbf{N}_T\| - r_i}{\|\mathbf{v}_i \cdot \mathbf{N}_T\|}, \quad (6-10)$$



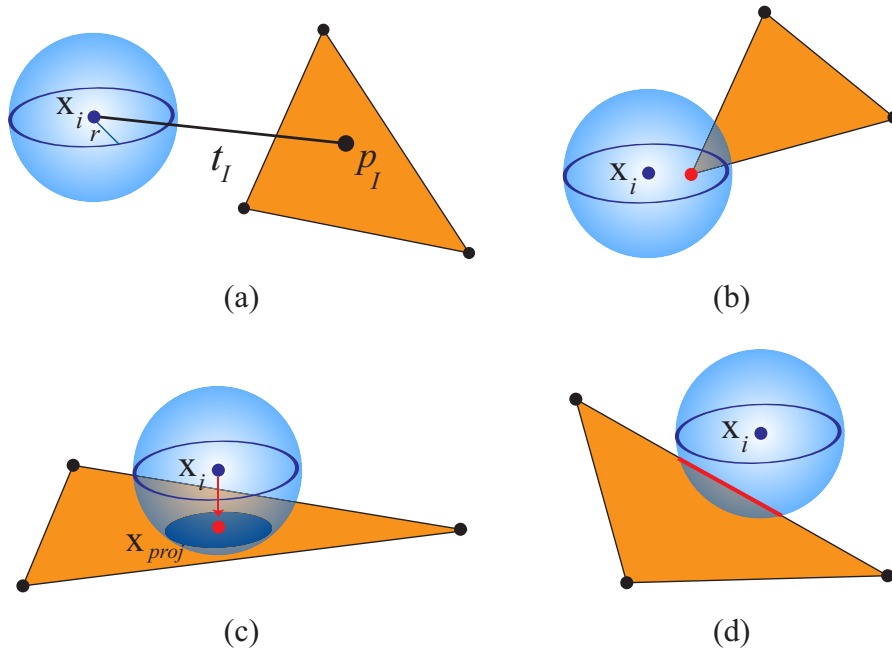


Figura 6.9: Intersecção esfera  $\times$  triângulo.

onde  $r_i$  é o raio da partícula e  $\mathbf{V}_T$  é um vértice de  $T$ . Após calcular o tempo de colisão  $t_I$ , posicionamos a partícula  $i$  no provável ponto de colisão  $\mathbf{x}'_i$  utilizando um passo do método de Euler

$$\mathbf{x}'_i = \mathbf{x}_i + t_I \mathbf{v}_i. \quad (6-11)$$

**Intersecção esfera  $\times$  triângulo.** Finalmente é feito o teste de intersecção entre a esfera associada à partícula  $i$  e o triângulo  $T$ . O teste de intersecção é realizado em três passos:

1. Teste se algum vértice de  $T$  está no interior da esfera. Se sim, a partícula  $i$  intersecta  $T$  (Figura 6.9(b)). Caso contrário, passe para o passo 2.
2. Projete o centro da esfera  $\mathbf{x}_i$  no plano determinado por  $T$ :

$$\mathbf{x}_{proj} = \mathbf{x}_i - [(\mathbf{x}_i - \mathbf{V}_T) \cdot \mathbf{N}_T] \mathbf{N}_T. \quad (6-12)$$

Verifique se  $\mathbf{x}_{proj}$  está no interior de  $T$ . Se sim, a partícula  $i$  intersecta  $T$  (Figura 6.9(c)). Caso contrário, passe para o passo 3.

3. Teste se a esfera intersecta alguma aresta de  $T$ . Se sim, a partícula  $i$  intersecta  $T$  (Figura 6.9(d)). Caso contrário, não há intersecção.

**Intersecção esfera  $\times$  segmento de reta.** No teste apresentado acima, precisamos analisar o teste de intersecção entre a partícula  $i$  e uma aresta

de  $T$  (Figura 6.9(d)). Seja  $\overline{\mathbf{V}_T \mathbf{W}_T}$  uma aresta de  $T$  formada pelos vértices  $\mathbf{V}_T$  e  $\mathbf{W}_T$ , queremos determinar se esse segmento intersecta a esfera de centro  $\mathbf{x}_i$  e raio  $r_i$ . Se a reta que passa por  $\mathbf{x}_i$  é perpendicular ao segmento  $\overline{\mathbf{V}_T \mathbf{W}_T}$  com ponto de intersecção  $\mathbf{r}$ , o teste de intersecção entre a partícula  $i$  e o segmento  $\overline{\mathbf{V}_T \mathbf{W}_T}$  se resume a determinar se o ponto  $\mathbf{r}$  está entre os pontos  $\mathbf{V}_T$  e  $\mathbf{W}_T$  (Figura 6.10). Podemos determinar o ponto  $\mathbf{r}$  através da equação paramétrica do segmento de reta

$$\mathbf{r} = \mathbf{V}_T + u\mathbf{d}, \quad \text{com} \quad \mathbf{d} = \mathbf{W}_T - \mathbf{V}_T \quad \text{e} \quad u = \frac{\mathbf{d} \cdot \mathbf{x}_i - \mathbf{d} \cdot \mathbf{V}_T}{\|\mathbf{d}\|^2}. \quad (6-13)$$

Seja  $d^2 = \|\mathbf{r} - \mathbf{x}_i\|^2$ , se caso  $d^2 > r_i^2$  a reta que passa por  $\overline{\mathbf{V}_T \mathbf{W}_T}$  não intersecta a esfera. Caso  $d^2 \leq r_i^2$  e  $u \in [0, 1]$  logo o segmento  $\overline{\mathbf{V}_T \mathbf{W}_T}$  intersecta a esfera e o ponto  $\mathbf{r}$  está entre os pontos  $\mathbf{V}_T$  e  $\mathbf{W}_T$ . O ciclo completo do teste de colisão é dado pelo Algoritmo 2.

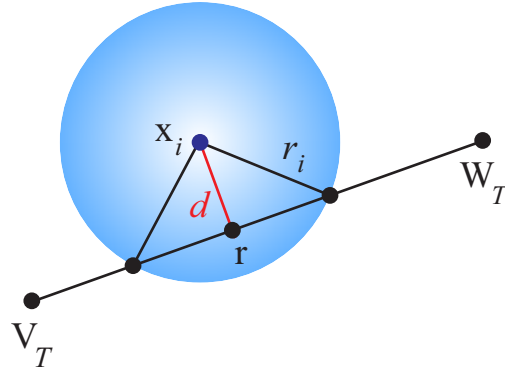


Figura 6.10: Intersecção esfera  $\times$  segmento de reta.

---

**Algoritmo 2** Teste de colisão

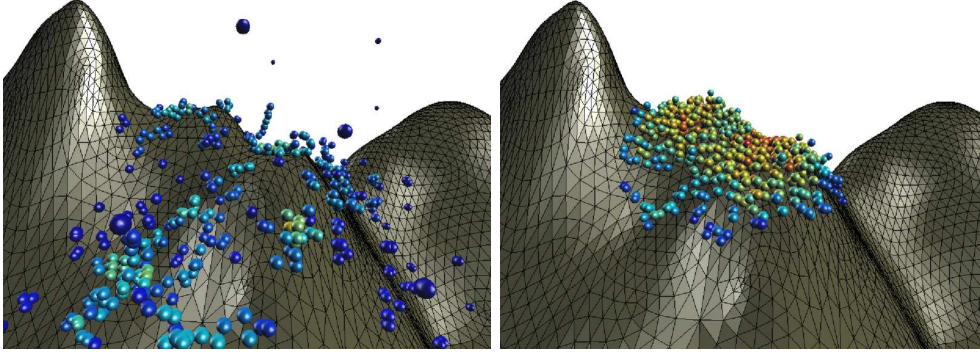
---

```

1: for cada partícula  $i$  do
2:   for cada triângulo  $T$  na lista de triângulos  $\mathcal{T}$  do
3:     if  $\mathbf{v}_i \cdot \mathbf{N} < 0$  then
4:       Calcule o tempo de colisão  $t_I$  (equação (6-10))
5:       if  $t_I < \Delta t$  then
6:         Calcule  $\mathbf{x}'_i = \mathbf{x}_i + t_I \mathbf{v}_i$ 
7:         if há intersecção entre a partícula na posição  $\mathbf{x}'_i$  e  $T$  then
8:           Calcule a velocidade de resposta  $\mathbf{v}'_i$  (Seção 6.3.1)
9:           Atualize:  $\mathbf{x}_i \leftarrow \mathbf{x}'_i + (\Delta t - t_I) \mathbf{v}'_i$  (Figura 6.7)
10:        end if
11:      end if
12:    end if
13:  end for
14: end for

```

---



6.11(a): Sem XSPH.

6.11(b): Com XSPH.

Figura 6.11: Simulação de um escoamento de lava após 1910 iterações, sem XSPH (esquerda) e com a correção XSPH (direita), com a mesma configuração da Figura 6.7. A simulação explode sem XSPH, devido à distância arbitrariamente pequena entre as partículas.

## 6.4

### Aspectos numéricos

Nessa seção, com o objetivo de melhorar a estabilidade numérica das simulações, introduziremos algumas ferramentas numéricas tais como viscosidade artificial e uma variante do método SPH conhecida como XSPH.

#### 6.4.1

##### Viscosidade artificial

Para evitar instabilidades numéricas devidas às oscilações nos campos vetoriais da velocidade e pressão, utilizamos uma técnica muito comum em elementos finitos que consiste em adicionar um termo de viscosidade artificial na equação de momento (4-38) a fim de dissipar essas oscilações indesejadas. Isso é feito da seguinte forma:

$$\frac{d\mathbf{v}_i}{dt} \leftarrow \frac{d\mathbf{v}_i}{dt} - \sum_{j \in N(\mathbf{x}_i)} \frac{m_j}{\rho_i} \Pi_{ij} \nabla_i W_{ij}. \quad (6-14)$$

O efeito da viscosidade artificial em sistemas SPH se deve ao seguinte termo:

$$\Pi_{ij} = \begin{cases} \frac{-\alpha \mu_{ij} c}{0.5(\rho_i + \rho_j)}, & \mathbf{v}_{ij} \cdot \mathbf{x}_{ij} < 0 \\ 0, & \mathbf{v}_{ij} \cdot \mathbf{x}_{ij} \geq 0 \end{cases} \quad \text{com} \quad \mu_{ij} = \frac{h(\mathbf{v}_{ij} \cdot \mathbf{x}_{ij})}{\|\mathbf{x}_{ij}\|^2 + 0.01h^2}, \quad (6-15)$$

com  $\mathbf{x}_{ij} = \mathbf{x}_i - \mathbf{x}_j$  e  $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$ . O termo linear da equação (6-15) corresponde à viscosidade volumétrica (*bulk viscosity*) [37]. Geralmente, o valor da constante  $\alpha$  é tomado próximo de 1.

### 6.4.2

#### Correção de velocidade XSPH

Com o objetivo de prevenir a interpenetração de partículas, a qual pode gerar aglomerados instáveis de partículas, Monaghan [46] introduziu a técnica XSPH (X de desconhecido) que consiste em calcular uma média das velocidades das partículas vizinhas. Essa técnica permite que as partículas se movimentem de uma forma mais ordenada num escoamento incompressível, reduzindo o problema de desordem de partícula nas simulações SPH (Figura 6.11).

Na técnica XSPH, a velocidade de cada partícula  $i$  é corrigida através da média das velocidades de suas partículas vizinhas ponderada por um parâmetro global constante  $\varepsilon \in [0, 1]$  da seguinte forma:

$$\mathbf{v}_i \leftarrow \mathbf{v}_i + \varepsilon \sum_{j \in N(\mathbf{x}_i)} \frac{2m_j}{(\rho_i + \rho_j)} (\mathbf{v}_j - \mathbf{v}_i) W_{ij}. \quad (6-16)$$

### 6.4.3

#### Integração numérica

Na simulação SPH de fluidos viscoplásticos, também integramos as equações de Navier–Stokes utilizando o integrador leap–frog (Seção 5.2.2), que dentre os integradores de precisão de segunda ordem é o mais eficiente, pois realiza uma única avaliação da aceleração por passo de tempo e utiliza pouco consumo de memória de armazenamento por avaliação. Nessa simulação, a condição de estabilidade CFL leva em conta os efeitos viscosos, resultando num passo de tempo adaptativo determinado pela expressão [50]:

$$\Delta t = 0.1 \min_i \left\{ \frac{h}{\|\mathbf{v}_i\| + c}, \frac{h^2 \rho_i}{8 \eta_i} \right\}. \quad (6-17)$$

## 6.5

### Implementação

A implementação do sistema SPH para fluidos viscoplásticos é feita de maneira muito semelhante a que foi mostrada no Capítulo 5. A principal diferença nessa implementação consiste no cálculo da viscosidade a partir do tensor de deformação, o ciclo de simulação SPH para fluidos viscoplásticos segue o esquema da Figura 6.12.

A inicialização das partículas no interior da malha triangular dos objetos simulados é feita utilizando um algoritmo de detecção de ponto no interior de um poliedro e pode ser encontrado no livro de O'Rourke [55], assim como um algoritmo para calcular o volume desses objetos (poliedros). Nessa implementação, existem os atributos de sistema e os atributos de partícula. Os atri-

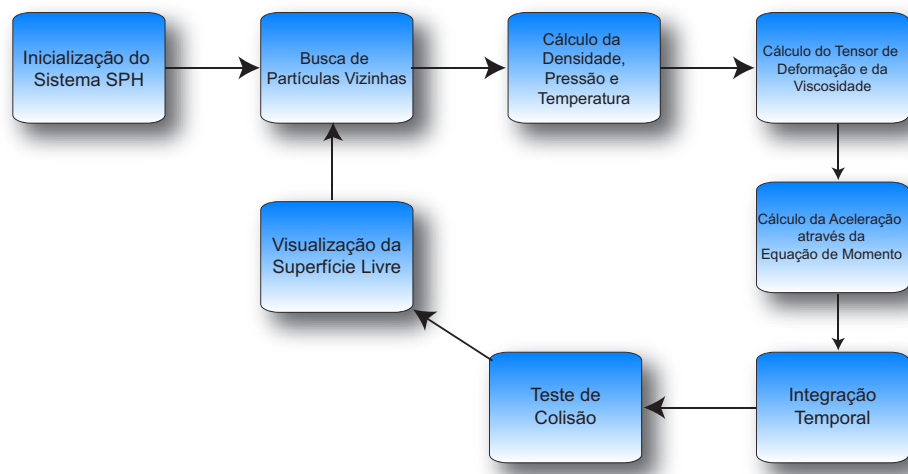


Figura 6.12: Ciclo de simulação de um sistema SPH para fluidos não-newtonianos.

butos de sistema como massa, velocidade do som e o comprimento suave  $h$  são constantes globais e não se alteram com o tempo. Os atributos de partícula variam em relação ao tempo e são armazenados em cada partícula. Esses atributos são dados pela Tabela 6.1 e são atualizados na sequência do Algoritmo 3.

Atributo	Descrição
$\mathbf{x}$	posição
$\mathbf{v}$	velocidade
$\mathbf{a}$	aceleração
$\mathbf{D}$	tensor de deformação
$\rho$	densidade
$\eta$	viscosidade
$T$	temperatura

Tabela 6.1: Atributos da partícula.

---

**Algoritmo 3** Sistema SPH para fluidos viscoplásticos

---

```

1: repeat
2:   Faça a busca das partículas vizinhas. (Seção 5.1.2)
3:   for cada partícula  $i$  do
4:     Atualize a pressão  $p_i$ . (equação (4-41) )
5:     Atualize a viscosidade  $\eta_i$ . (equação (6-1))
6:   end for
7:   for cada partícula  $i$  do
8:     Calcule a derivada da densidade. (equação (4-37))
9:     Calcule a aceleração. (equações (4-38) e (6-14))
10:    Calcule a derivada da temperatura. (equação 6-2)
11:  end for
12:  for cada partícula  $i$  do
13:    Atualize  $\mathbf{v}_i$ ,  $T_i$  e  $\rho_i$  com o integrador leap-frog. (Seção 5.2.2)
14:    Correção de  $\mathbf{v}_i$  com XSPH. (equação (6-16))
15:    Eventual colisão:  $(\mathbf{x}_i, \mathbf{v}_i) \leftarrow \text{colisao}(\mathbf{x}_i + \Delta t \mathbf{v}_i)$ . (Seção 6.3)
16:  end for
17:  Atualize  $\Delta t$  usando a condição CFL. (equação (6-17))
18:   $\text{tempo} = \text{tempo} + \Delta t$ 
19: until  $\text{tempo} < \text{tempo}_{total}$ 

```

---

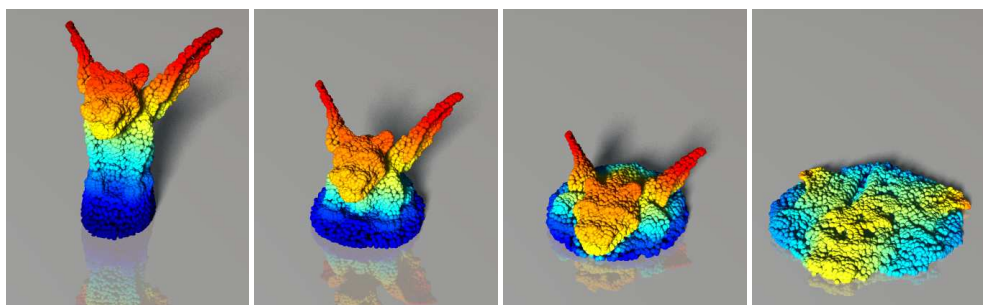
## 6.6

### Resultados

Testamos o método descrito nesse capítulo com a finalidade de simular fenômenos físicos complexos. Nosso principal objetivo está ilustrado nas Figuras 1.1 e 6.5, onde os resultados batem com a intuição dos processos de deformação plástica, derretimento de objetos sólidos e fluxos de lava.

Propomos uma nova formulação lagrangeana para fluidos não-newtonianos usando o método SPH combinado com algumas técnicas numéricas com intuito de garantir a estabilidade das simulações. A estabilidade aparece claramente na Figura 6.13, onde a cabeça do modelo Gargoyle permanece bem definida mesmo quando o modelo está quase completamente derretido. Nesse caso, todas as 6976 partículas iniciam com a temperatura acima do ponto de fusão e fluem como fluido não-newtoniano.

O controle eficiente da viscosidade através da formulação de Fluidos Newtonianos Generalizados permite resultados visualmente realísticos. No exemplo da Figura 6.14, criamos um objeto viscoplástico a partir da superfície chair com 7000 partículas e simulamos a sua interação com um objeto sólido complexo representado pelo esqueleto de uma mão com 2351 triângulos. O nosso método, além de capturar muito bem os efeitos viscosos do material, permite que o objeto sofra grandes mudanças topológicas sem controle explícito de sua superfície livre. Perceba que, mesmo após quase todo material escorrer entre



6.13(a): 20 iter.      6.13(b): 550 iter.      6.13(c): 780 iter.      6.13(d): 1320 iter.

Figura 6.13: Derretimento do modelo Gargoyle totalmente líquido usando 6976 partículas: o mapa de cores representa a temperatura de cada partícula. A estabilidade do método preserva a forma do objeto sem uma representação explícita da malha mesmo após várias iterações.

os dedos e tocar o chão, parte dele fica agarrado no esqueleto.

A flexibilidade do nosso método, além de permitir simulações de fluidos viscoplásticos, também possibilita a simulação de objetos sólidos que sofrem deformações. No exemplo da Figura 6.15, simulamos o impacto de uma esfera de metal contra uma parede plástica (verde). Note que, momentos antes da esfera perfurar a parede, a energia dissipada pelo choque da esfera produz uma grande deformação no material.

O exemplo mostrado na Figura 6.16 simula o derretimento do modelo Stanford Bunny com 10188 partículas. A simulação é inicializada com um gradiente linear de temperatura tal que a orelhas derretem enquanto o corpo permanece frio e sólido. Note que o comportamento físico é coerente, especialmente quando uma das orelhas esfria após tocar o corpo, enquanto a outra permanece quente (Figura 6.4).

A adaptatividade do passo de tempo devido à condição CFL além de garantir estabilidade numérica na integração temporal permite que as simulações sejam mais eficientes. Por exemplo, o derretimento das letras “CGF 2007” mostrado na Figura 6.17 é discretizado utilizando 10773 partículas, ou seja, mais partículas que o modelo Bunny. Entretanto, essa simulação requer um tempo médio de execução menor para cada passo de tempo (veja Tabela 6.2). Isso se deve à baixa densidade de partículas do modelo, a qual permite que os passos de tempo sejam maiores.

Nossas simulações também obtiveram sucesso em reproduzir a morfologia vista em escoamentos reais de lava tais como o espalhamento da frente de lava na ausência de solidificação e o desenvolvimento de complexas e assimétricas estruturas em forma de dedos conhecidas como *lobus* (veja Figura 6.18). A formação dessas estruturas se deve à influência da topografia do terreno na lava; essa influência é reproduzida graças ao teste de colisão introduzido na

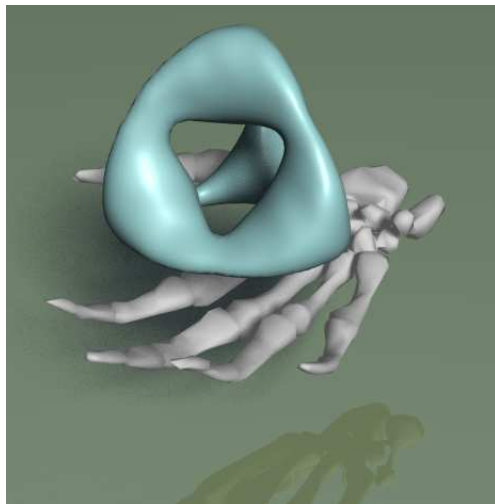


Seção 6.3.

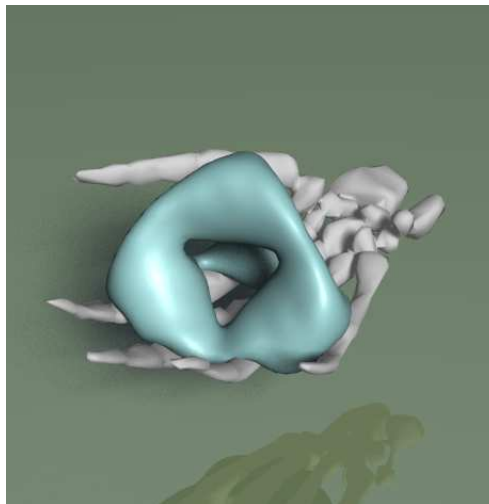
Resumindo, o modelo de Fluido Newtoniano Generalizado proposto por Mendes se adapta muito bem em nossas simulações SPH, pois além de sua formulação ser concisa, ela mantém o significado físico de todo processo.

Simulação	Número de partículas	Tempo por iteração
Mão	5000	0.13s
Parede	5000	0.09s
Chair	7000	0.15s
Bunny	10188	0.18s
Vela	7364	0.11s
Gargoyle	7007	0.16s
CGF	10773	0.13s
Terreno	4900	0.08s
Pão de Açúcar	10137	0.48s

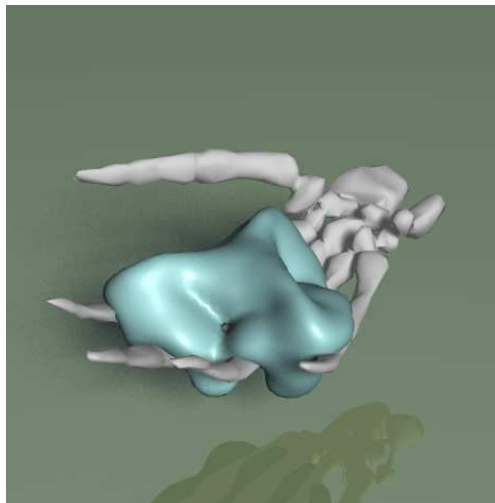
Tabela 6.2: Tempo médio de iteração em cada simulação executadas num computador com processador Pentium 4 – 2.4 GHz e com 2 Gb de RAM.



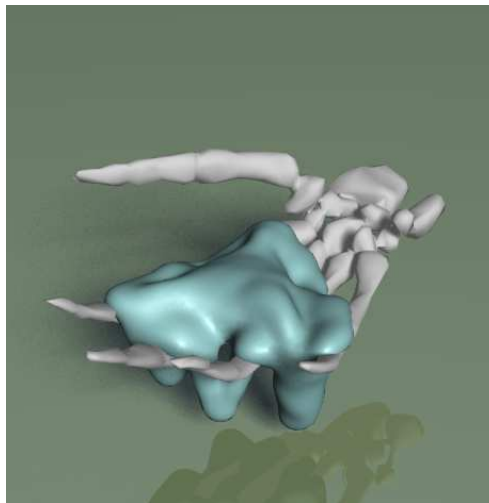
6.14(a): Objeto inicial.



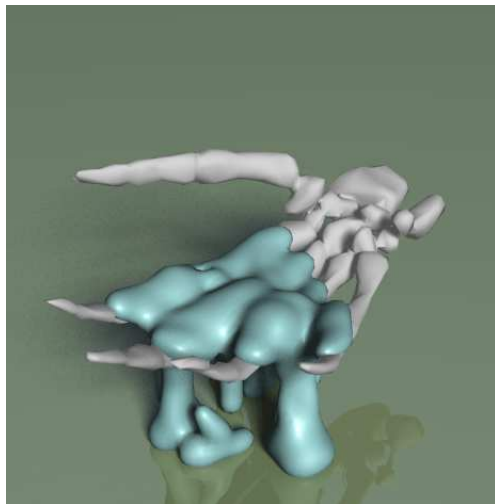
6.14(b): 600 iterações.



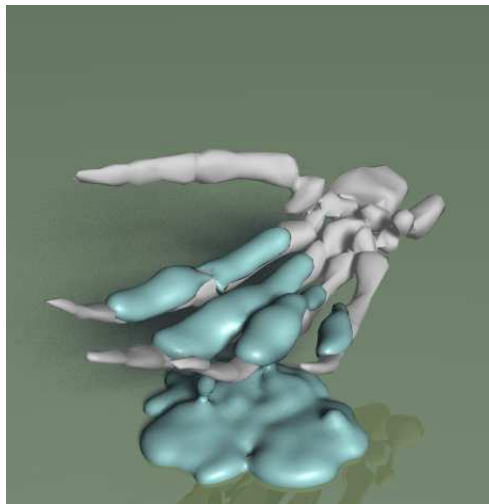
6.14(c): 1400 iterações.



6.14(d): 2800 iterações.

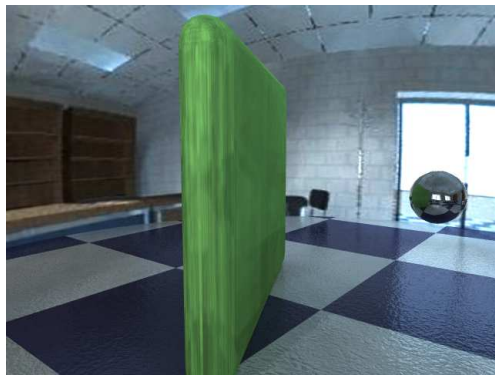


6.14(e): 6000 iterações.

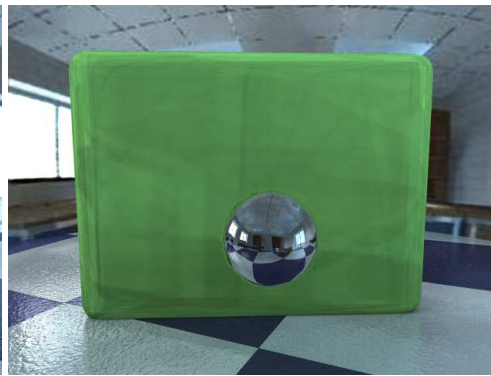


6.14(f): 19000 iterações.

Figura 6.14: Superfície chair modelada como um material viscoplástico com 7000 partículas, interagindo com um objeto sólido complexo representado pelo esqueleto de uma mão com 2351 triângulos.



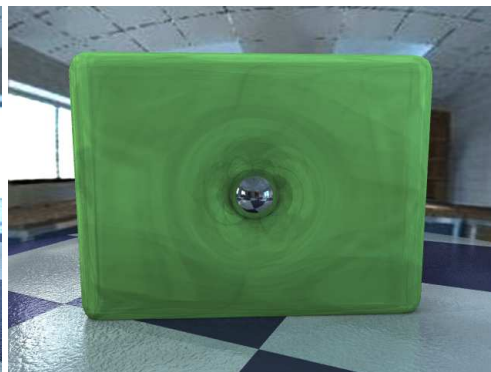
6.15(a): Objeto inicial.



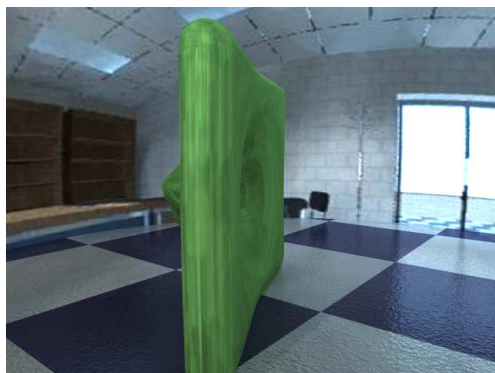
6.15(b): Objeto inicial.



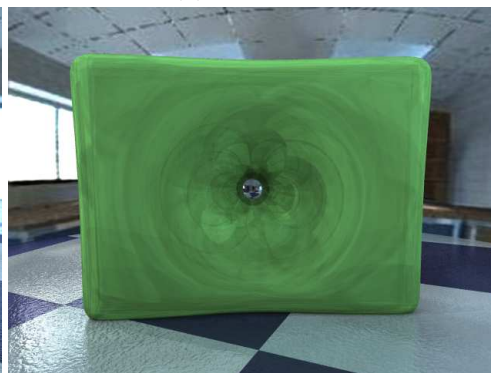
6.15(c): 1300 iterações.



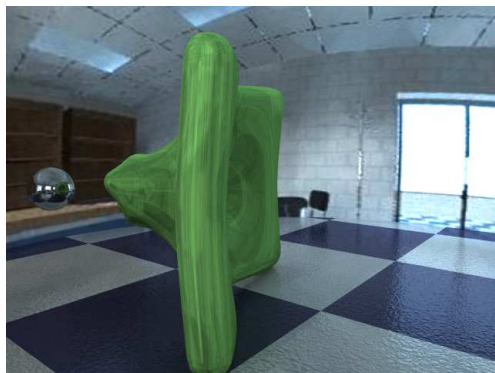
6.15(d): 1300 iterações.



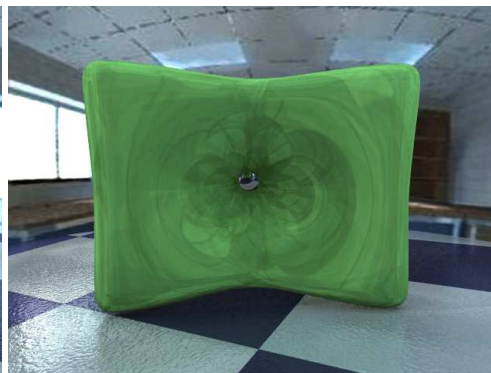
6.15(e): 1500 iterações.



6.15(f): 1500 iterações.



6.15(g): 2000 iterações.

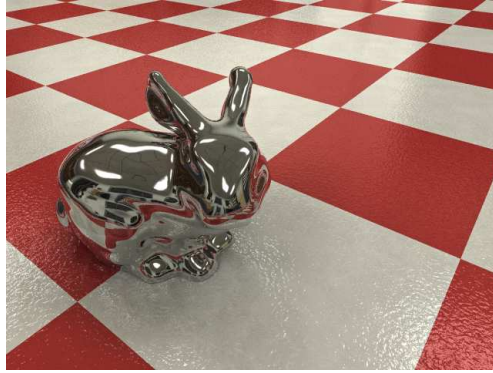


6.15(h): 2000 iterações.

Figura 6.15: Colisão de uma esfera metálica contra uma parede plástica (verde) constituída por 5000 partículas. A simulação é vista ao mesmo tempo em duas posições diferentes: vista lateral (esquerda) e vista frontal (direta).



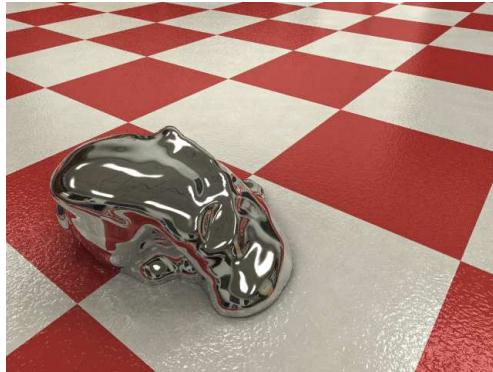
6.16(a): Objeto inicial.



6.16(b): 400 iterações.



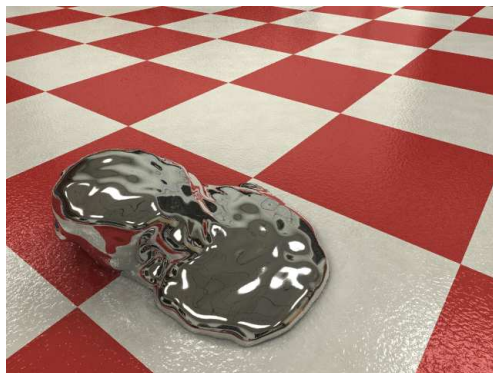
6.16(c): 800 iterações.



6.16(d): 1200 iterações.



6.16(e): 1800 iterações.



6.16(f): 4000 iterações.

Figura 6.16: Evolução da superfície livre da simulação de derretimento do modelo Stanford Bunny com 10188 partículas, iniciando frio na base e quente no topo do modelo.





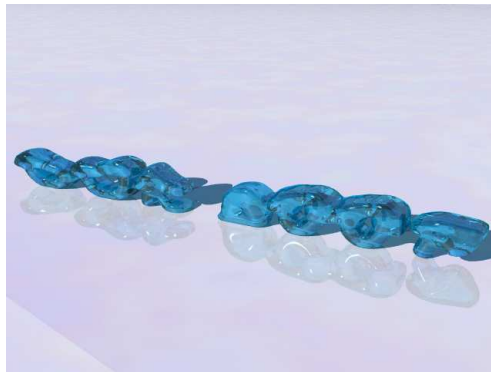
6.17(a): Objeto inicial.



6.17(b): 450 iterações.



6.17(c): 900 iterações.



6.17(d): 1500 iterações.



6.17(e): 2500 iterações.



6.17(f): 4150 iterações.

Figura 6.17: Derretimento das letras “CGF 2007” usando 10773 partículas, iniciando frio na base e quente no topo do modelo. A adaptatividade do passo de tempo permite uma simulação mais precisa com poucas iterações.

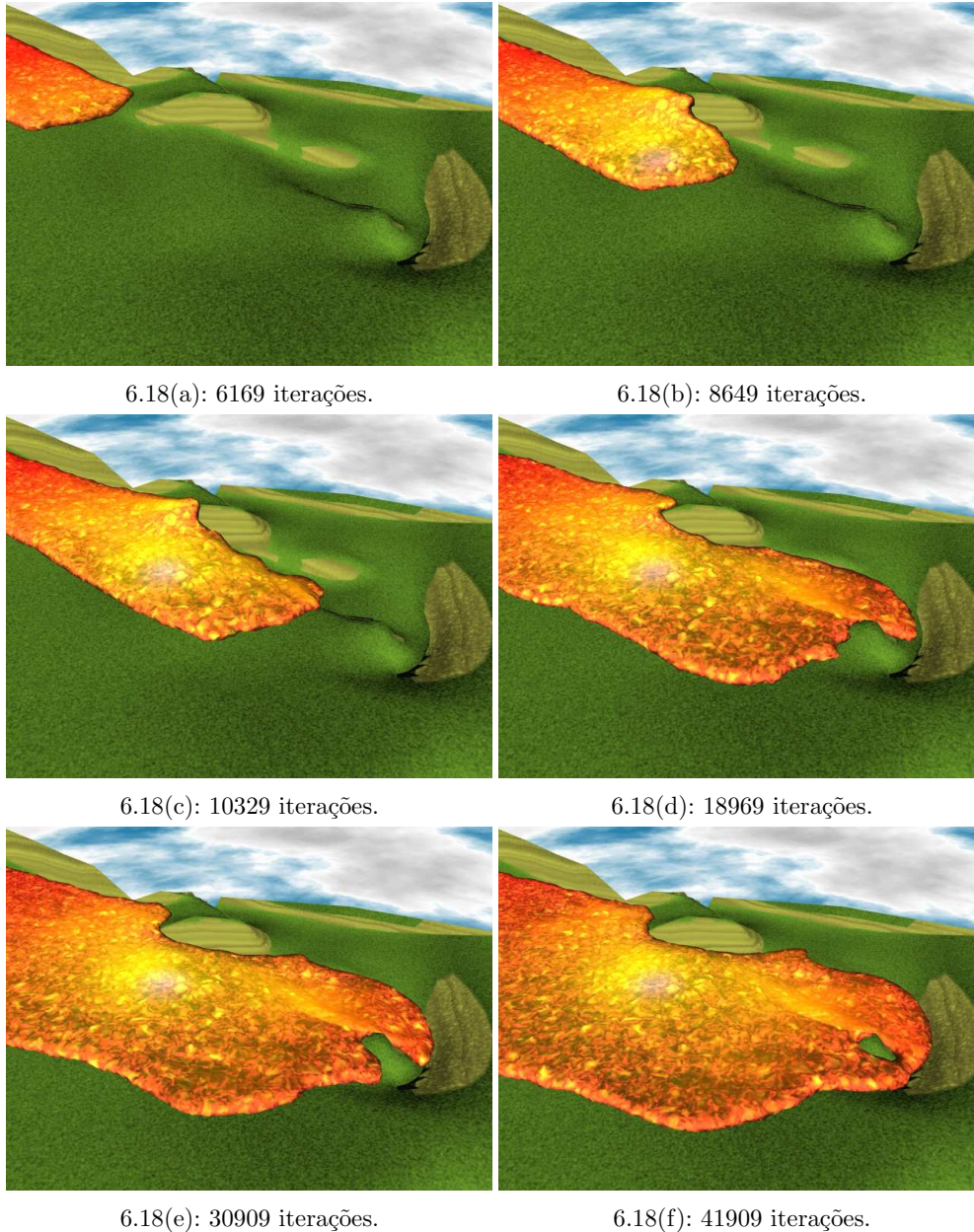


Figura 6.18: Escoamento de lava num terreno virtual de 1547 triângulos, com 4900 partículas.