

Animação Dinâmica de Corpos Elásticos

ROGÉRIO L. W. LIESENFELD¹
JORGE STOLFI²

Departamento de Ciência da Computação (DCC-IMECC)
Universidade Estadual de Campinas (UNICAMP)
Caixa Postal 6065 – 13081-970 Campinas, SP
{rluis, stolfi}@dcc.unicamp.br

Abstract. We describe here a dynamic animation system for elastic bodies, based on a simplicial finite element model with linear deformations but non-linear elastic forces.

1 Introdução

Descrevemos aqui um sistema para animação dinâmica de corpos deformáveis elásticos [1], cujos aspectos originais incluem o uso de forças elásticas não-lineares, e a derivação das equações de movimento a partir da fórmula de Lagrange.

O sistema permite modelar objetos poliédricos, de topologia e geometria arbitrárias, por meio de elementos finitos simpliciais de primeiro grau. Ou seja, cada objeto é modelado por um conjunto de tetraedros colados entre si; e a deformação de cada tetraedro é restrita a uma transformação afim do \mathbf{R}^3 .

As propriedades mecânicas de cada tetraedro T são determinadas pela sua *forma de repouso* T_0 (um tetraedro no \mathbf{R}^3), e pelos coeficientes de elasticidade e viscosidade do material.

2 Equação do movimento

As equações de movimento do sistema são obtidas da *fórmula de Lagrange* [2]

$$\frac{d}{dt} \left(\frac{\partial K}{\partial \dot{q}} \right) + \frac{\partial W}{\partial \dot{q}} + \frac{\partial P}{\partial q} = V + G \quad (1)$$

onde q é um vetor que descreve a forma e posição dos objetos; \dot{q} é sua derivada temporal; K é a energia cinética do sistema; P é a energia interna do sistema, devida à deformação dos tetraedros; W é a potência dissipada pela viscosidade interna do material; V são as forças de reação nos vínculos; e G são as forças externas (incluindo gravidade, atrito, resistência do meio, e forças arbitrárias especificadas pelo usuário).

Estado do sistema: Uma vez que cada tetraedro T admite apenas deformações lineares, a posição (ou velocidade) corrente de qualquer partícula de T pode ser calculada a partir de sua posição no tetraedro de repouso T_0 e das posições (ou velocidades) correntes dos vértices de T . Portanto, se os objetos têm n

vértices distintos, os vetores q e \dot{q} que descrevem o estado corrente do sistema têm $3n$ componentes cada, que são as posições e velocidades desses vértices.

Energia cinética: A energia cinética de um tetraedro T de massa m_T é determinada pelas velocidades v_0, \dots, v_3 dos seus vértices, segundo a fórmula

$$K_T = \frac{m_T}{20} \left(\sum_{i=0}^3 |v_i|^2 + \sum_{i=0}^3 \sum_{j=0}^{i-1} |v_i| |v_j| \right) \quad (2)$$

Energia potencial: A densidade de energia elástica por unidade de volume é constante no interior de cada tetraedro, e função apenas da transformação linear C que mapeia sua forma de repouso T_0 na sua forma atual T . No nosso simulador, adotamos a seguinte fórmula para a densidade de energia:

$$e_T = \alpha \left(\Delta + \frac{1}{\Delta} - 2 \right) + \beta (\Gamma^2 - 3\Sigma) \quad (3)$$

onde α e β são os módulos de elasticidade do material, e Δ , Σ , e Γ são invariantes de rotação dados pelos coeficientes do polinômio característico $\chi(\lambda) = |C^T C - \lambda I|$ da matriz $C^T C$ [3]. A energia potencial elástica P_T do tetraedro é dada por $P_T = V_0 e_T$, onde V_0 é o volume do tetraedro de repouso T_0 .

O primeiro termo da equação (3) mede apenas a energia devida à expansão e à redução do volume do tetraedro, ignorando cisalhamentos. O segundo termo mede apenas a mudança de forma do tetraedro, ignorando mudanças de volume. Uma propriedade importante da fórmula (3) é que ela tende para infinito quando o volume corrente do tetraedro tende a zero, o que automaticamente impede que o tetraedro “vire do avesso” durante a simulação do sistema.

No decorrer da simulação é necessário calcular as derivadas parciais da fórmula (3) em relação às posições dos vértices do tetraedro. Para isso usamos o algoritmo eficiente de Baur-Strassen [4], baseado na aplicação sistemática da regra da cadeia.

¹Bolsa de mestrado FAPESP 94/4132-6.

²Bolsa de auxílio à pesquisa CNPq 301016/92-5.

Forças dissipativas internas: O termo $\partial W/\partial \dot{q}$ em (1) representa as forças devidas à viscosidade do material, que tendem a se opor ao movimento relativo de partículas vizinhas. A potência dissipada (W) por essas forças num tetraedro T , por unidade de volume, pode ser calculada a partir da matriz $D = \dot{C}C^{-1}$, que define a taxa de deformação de T , de maneira semelhante ao cálculo da energia elástica.

Forças de vínculos: Vínculos são restrições impostas sobre o movimento do sistema, devidas a articulações entre objetos, contatos entre corpos, etc., expressas por equações da forma $h(q, t) = 0$.

O termo V em (1) representa as forças internas necessárias para satisfazer estas restrições; por exemplo, as forças que dois corpos em contato aplicam um sobre o outro. As forças produzidas por um vínculo $h_i(q, t) = 0$ tem a forma $\lambda_i \partial h_i / \partial q$, onde λ_i é um escalar (*multiplicador de Lagrange*) a ser determinado. Para manter o vínculo satisfeito as forças devem ser tais que $\ddot{h}_i(q, t) = 0$ para todo vínculo. Os multiplicadores λ_i , e portanto as forças de vínculo V , podem ser calculados a partir destas equações.

Formulação matricial: Aplicando-se a equação de Lagrange a todos os tetraedros do modelo, obtemos um sistema de $3n$ equações diferenciais ordinárias acopladas de 2ª ordem. Dadas as posições e velocidades de todos os vértices do modelo, e as forças externas e internas agindo no sistema, estas equações permitem calcular as acelerações de cada vértice.

Este sistema de equações pode ser escrito na forma $M\ddot{q} = F$, onde M é uma matriz constante de dimensão $3n \times 3n$ (a *matriz de massa* do sistema), \ddot{q} é o vetor das acelerações dos vértices, e $F = V + G - \partial P / \partial q - \partial W / \partial \dot{q}$ é o vetor das forças que agem sobre os vértices. A matriz M é positivo-definida e esparsa (com $O(n)$ elementos não nulos). Este fato permite a solução eficiente do sistema $M\ddot{q} = F$ [3, 5].

Integração numérica: Para integração numérica das equações do movimento, utilizamos o método adaptativo de 4ª ordem de Runge-Kutta-Fehlberg [5].

3 Colisões

Animações realistas exigem a detecção e simulação de colisões entre os objetos. No nosso caso, basta considerar colisões entre um vértice e uma face, ou entre duas arestas. Em ambos os casos, a colisão é definida por uma equação de 4º grau, e algumas inequações auxiliares, relacionando as coordenadas e velocidades dos quatro vértices envolvidos.

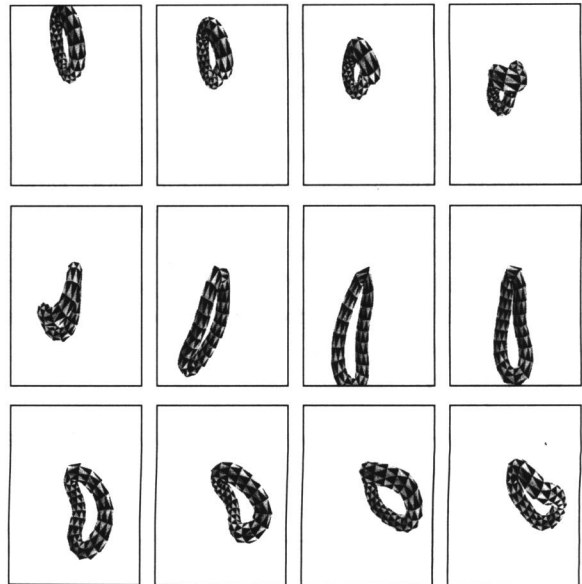
Como os tetraedros admitem apenas deformações afins, é necessário tratar cada colisão como um evento instantâneo. Nesse caso, uma colisão equivale a um *impulso* (uma força de magnitude infinita e duração infinitesimal) aplicado aos objetos no ponto

de contato, que pode ser calculado resolvendo-se uma versão modificada do sistema $M\ddot{q} = F$.

4 Implementação e resultados

O simulador tem cerca de 3000 linhas de código em Modula-3. Estamos atualmente implementando a detecção de colisões e o tratamento de vínculos. O sistema inclui também uma biblioteca de rotinas em Modula-3 para a criação, colagem, e posicionamento de tetraedros. As animações produzidas pelo simulador são exibidas posteriormente em tempo real numa estação Silicon Graphics Indigo² por um programa visualizador escrito em C++.

Abaixo vemos a animação de 1.8 segundos de um toro elástico (480 tetraedros) com um vértice preso no ponto (0, 0, 0) (no centro da imagem), e que cai sob a força da gravidade. A fatoração inicial da matriz M (540×540) levou 8.9 minutos, e a integração levou 9.1 minutos, numa Sparc 1000.



Referências

- [1] D. Terzopoulos, J. Platt, A. Barr, e K. Fleischer. *Elastically deformable models*. Proceedings ACM SIGGRAPH'87, 205–214.
- [2] K. R. Symon, *Mecânica* (5ª edição). Editora Campus (1982).
- [3] G. H. Golub e C. F. van Loan, *Matrix Computations*. Johns Hopkins University Press (1983).
- [4] W. Baur e V. Strassen, *The complexity of partial derivatives*. Theoretical Computer Science 22 (1983), 317–330.
- [5] R. L. Burden e J. D. Faires, *Numerical Analysis* (4th edition). PWS-Kent (1989).