Animação Dinâmica de Corpos Elásticos

Rogério Luís Walter Liesenfeld

Animação Dinâmica de Corpos Elásticos

Este exemplar corresponde à redação final da dissertação devidamente corrigida e defendida pelo Sr. Rogério Luís Walter Liesenfeld e aprovada pela Comissão Julgadora.

Campinas, 14 de abril de 1997.

Prof. Dr. Jorge Stolfi
Orientador

Dissertação apresentada ao Instituto de Computação, UNICAMP, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

Animação Dinâmica de Corpos Elásticos[1]

Rogério Luís Walter Liesenfeld[2]

Instituto de Computação

IC — UNICAMP

Banca Examinadora:

- Wu Shin Ting[4]
- Cândido Ferreira Xavier de Mendonça Neto[3]
- João Carlos Setubal (Suplente)[3]
- Jorge Stolfi (Orientador)[3]

[1]Dissertação apresentada ao Instituto de Computação da UNICAMP, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

[2]O autor é Bacharel em Informática pela Universidade Federal do Paraná.

[4]Professora do DCA - Faculdade de Engenharia Elétrica e Computação - UNICAMP.

[3]Professor do Instituto de Computação - UNICAMP.

Agradecimentos

Ao CNPq e à FAPESP, pelo apoio financeiro recebido.

Aos meus pais e ao meu irmão, pelo apoio e confiança em todos os momentos.

Ao meu orientador Jorge Stolfi, pelo inestimável auxílio que prestou ao longo do desenvolvimento deste trabalho. Suas inúmeras críticas, sugestões e comentários em muito contribuíram para a qualidade desta dissertação. Agradeço-lhe também pela amizade, e pelo grande exemplo de excelência profissional e acadêmica que me foi dado a conhecer nestes três anos.

A todos os professores, colegas e amigos que tive o prazer de conhecer em Campinas.

Aos funcionários do Intituto de Computação e da UNICAMP.

"Be not too tame neither, but let your own discretion be your tutor: suit the action to the word, the word to the action; with this special observance, that you o'erstep not the modesty of nature: for anything so overdone is from the purpose of playing, whose end, both at the first and now, was and is, to hold, as 'twere, the mirror up to nature; to show virtue her own feature, scorn her own image, and the very age and body of the time his form and pressure."

William Shakespeare
"Hamlet, Prince of Denmark"

"Thou, nature, art my goddess; to thy laws
My services are bound."

William Shakespeare
"King Lear"

Resumo

Esta dissertação descreve o projeto e a implementação de um sistema de animação para corpos elásticos. O movimento dos corpos é simulado através da integração numérica de um sistema de equações diferenciais, que obedece as leis da mecânica newtoniana. Nós empregamos a equação de Lagrange para formular essas equações em termos de funções escalares de energia, sobre coordenadas generalizadas.

Cada corpo é modelado como uma região tridimensional de um meio material contínuo. A deformação de um fragmento do material dá origem a forças internas de elasticidade, que resistem à deformação. Calculamos estas forças através de uma função que mede a energia elástica acumulada no corpo, especialmente projetada para impedir compressões a volume zero. O atrito entre partículas vizinhas no interior do corpo, quando elas estão em movimento relativo, dá origem a forças internas de viscosidade. Estas forças são calculadas através de uma função que mede a potência dissipada pelas mesmas.

Os corpos elásticos são aproximados por malhas irregulares de fragmentos tetraédricos colados entre si. Cada tetraedro possui massa constante e propriedades materiais uniformes, com deformações restritas a transformações afins de sua forma de repouso. Assim, as coordenadas cartesianas dos vértices dos tetraedros determinam completamente a forma e posição de cada corpo.

Restrições genéricas, escritas como equações algébricas sobre as coordenadas dos vértices e o tempo, podem ser impostas ao movimento dos objetos. O sistema calcula automaticamente as forças de reação necessárias para manter essas restrições satisfeitas ao longo do tempo.

Pode-se também impor certas *condições* sobre as coordenadas (como impenetrabilidade) que, em contraste com as *restrições*, não geram forças quando satisfeitas. Quando a evolução dinâmica do sistema alcança um estado onde uma das condições está para ser violada, a integração é temporariamente suspensa, até que as equações de movimento sejam ajustadas. Feito isto, a integração recomeça, com todas as condições satisfeitas.

Descrevemos técnicas gerais para detetar a ocorrência de tais eventos, com ênfase na deteção eficiente de colisões. Quando uma colisão ocorre, uma mola temporária é fixada aos pontos de contato, aplicando uma força de separação que se opõe à interpenetração.

Abstract

This dissertation describes the design and implementation of an animation system for elastic bodies. The bodies' motion is simulated through the numerical integration of a differential equation system, according to the laws of Newtonian mechanics. We employ the Lagrange's equation to formulate these equations in terms of scalar energy functions, over generalized coordinates.

Each body is modelled as a tridimensional region of a continuous material medium. The deformation of a fragment of the material gives rise to internal elasticity forces, which resist the deformation. We compute these forces from a function that measures the elastic energy stored in the body, specially devised to prevent compressions to volume zero. The friction between neighbouring particles in the body's interior, when they are in relative motion, gives rise to internal viscosity forces. These forces are computed from a function that measures the power dissipated by them.

Elastic bodies are approximated by irregular lattices of tetrahedral fragments glued among themselves. Each tetrahedron possesses constant mass and uniform material properties, with deformations restricted to affine transformations of its rest shape. So, the Cartesian coordinates of its vertices completely determine the shape and position of each body.

Generic constraints, written as algebraic equations over the vertex coordinates and time, can be imposed to the motion of the objects. The system automatically computes the reaction forces needed to maintain these constraints satisfied over time.

It's also possible to impose certain *conditions* over the coordinates (like impenetrability), which, unlike the *constraints*, don't generate forces when satisfied. When the dynamic system's evolution reaches a state where one of the conditions is to became violated, the integration is temporarilly suspended, until the equations of motion are fixed. This done, integration is resumed, with all the conditions satisfied.

We describe general techniques to detect the occurence of such events, with emphasis on the efficient detection of collisions. When a collision happens, a spring is attached to the contact points, applying a separation force that opposes interpenetration.

Conteúdo

Lista de Tabelas

Lista de Figuras

Capítulo 1

Introdução

1.1 Animação por computador

A animação por computador visa automatizar parcial ou totalmente as atividades envolvidas na produção de animações gráficas, para fins educacionais, de entretenimento, científicos, de engenharia, etc. Um problema central nesta área é o controle do movimento dos objetos e personagens.

1.1.1 Animação cinemática

Uma técnica primitiva de animação por computador é a *animação procedural*, que consiste na criação e uso de procedimentos que especificam a evolução no tempo dos objetos e suas características. Tais procedimentos geram o movimento a partir de fórmulas ou algoritmos arbitrários escolhidos pelo animador.

Outra técnica, atualmente usada em praticamente todos os sistemas comerciais de animação, é a *interpolação de quadros-chave*, na qual o animador especifica a posição dos objetos em alguns quadros da animação, e o sistema gera os quadros intermediários a partir daqueles por meio de uma regra de interpolação.

Estas duas técnicas de *animação cinemática* oferecem grande flexibilidade no controle do movimento, porém requerem do animador muita habilidade e experiência para produzir movimentos realistas de objetos ou personagens mais complexos.

1.1.2 Animação dinâmica

Em resposta a esse problema, a técnica de *animação dinâmica*, baseada em leis físicas, surgiu como uma alternativa para a produção de animações complexas com alto grau de realismo. Nesta abordagem, o movimento dos objetos é determinado pela simulação das leis da *dinâmica* — que é a seção da mecânica newtoniana que descreve o movimento de um sistema de partículas a partir das forças aplicadas às mesmas.

Na animação dinâmica, os objetos da cena possuem certas propriedades físicas tais como massa, elasticidade e coeficientes de atrito, e o seu movimento é determinado pelas forças que agem sobre os mesmos. Algumas dessas forças surgem da própria natureza dos objetos, como por exemplo as forças internas de elasticidade e viscosidade. Outras são forças externas, como a gravidade, forças aplicadas por motores, músculos, etc. Além disso, quando há contato ou colisão entre dois objetos, surgem forças de reação que impedem que eles se interpenetrem.

Para propósitos de animação, é importante que o movimento simulado dos objetos possa ser controlado, de modo a atingir os efeitos desejados pelo animador. Em um sistema dinâmico, tal controle pode ser obtido de maneira fisicamente consistente através de forças, que podem ser calculadas diretamente ou por meio de restrições formais impostas ao movimento dos objetos. Em particular, é possível controlar cinematicamente o movimento. Por exemplo, suponha que o animador especifica completamente a trajetória desejada para um objeto em um certo intervalo de tempo: em geral é possível calcular, sob certas condições, forças externas que atuam sobre esse objeto ao longo daquele intervalo, obrigando-o a seguir a trajetória especificada.

Muitos resultados computacionais estimulantes foram obtidos na área de animação dinâmica. Apesar disso, os sistemas desse tipo são ainda utilizados em caráter experimental, ou para efeitos específicos em certas cenas. Esta situação deve-se em parte ao alto custo da animação dinâmica, tanto do ponto de vista do seu projeto e implementação quanto do custo computacional envolvido em seu uso. Na prática, tais sistemas não podem ser usados interativamente, pelo menos para modelos razoavelmente complexos.

1.2 Descrição do trabalho

Nesta dissertação descrevemos o projeto e a implementação de um sistema de animação dinâmica, especializado para objetos deformáveis elásticos e visco-elásticos. O trabalho enfatiza a modelagem física dos materiais elásticos, e a deteção eficiente de colisões entre objetos complexos e deformáveis.

1.2.1 Determinação do movimento

Na mecânica newtoniana, os objetos materiais são constituídos de partículas de tamanho desprezível e massa finita, e seu movimento é completamente especificado através da descrição do movimento dessas partículas. Neste trabalho, vamos supor que a posição, a velocidade e a aceleração de cada partícula, bem como as forças atuando sobre as partículas, podem ser descritas por coordenadas cartesianas em um sistema inercial de referência.

É inviável modelar o movimento de cada partícula num objeto macroscópico, mas felizmente, em objetos sólidos as posições de todas as partículas podem ser determinadas com precisão suficiente a partir de um conjunto pequeno de parâmetros — as *coordenadas generalizadas* do sistema. Por exemplo, a posição e a orientação de um corpo rígido ideal podem ser determinadas por apenas seis parâmetros.

Dado tal conjunto de coordenadas, pode-se montar um sistema de equações diferenciais ordinárias de segunda ordem, que descreve a evolução das coordenadas ao longo do tempo. Em cada instante, os valores das coordenadas e das velocidades generalizadas (derivadas primeiras das coordenadas), juntamente com os valores das forças externas que agem sobre as partículas, permitem calcular as acelerações generalizadas (derivadas segundas das coordenadas) nesse instante. Conhecidas as acelerações em uma seqüência discreta de instantes, pode-se calcular as posições dos objetos nesses instantes através de um método numérico para integração de equações diferenciais.

A formulação de Lagrange da mecânica de Newton fornece uma receita conveniente para produzir as equações diferenciais de movimento, desde que a energia cinética dos objetos possa ser escrita como uma função das coordenadas e velocidades generalizadas, e que todas as forças presentes no sistema possam ser escritas como funções das coordenadas e velocidades generalizadas e do tempo. A partir destas funções, a equação de Lagrange permite-nos escrever as equações de movimento.

Forças que dependem apenas das posições correntes das partículas podem, em geral, ser calculadas através de uma função de energia potencial, escrita em termos das coordenadas generalizadas apenas. Algumas forças dissipativas (que dependem das posições e velocidades correntes das partículas) podem ser calculadas através de uma função que dá a cada instante a taxa de dissipação da energia cinética (ou a *potência dissipada*) do sistema, escrita em termos das coordenadas e velocidades generalizadas. Outras forças (tais como gravidade, forças de contato, etc.) podem geralmente ser escritas como funções arbitrárias das coordenadas e velocidades generalizadas e do tempo.

1.2.2 Restrições de movimento

Em um sistema dinâmico, os objetos muitas vezes têm seu movimento restrito, ou seja, nem todas as posições ou velocidades que as partículas poderiam eventualmente atingir são permitidas. Por exemplo, uma partícula de um corpo pode ser fixada a uma dada posição no espaço, ou ser restrita a se mover sobre uma determinada superfície; dois corpos podem ser conectados por uma articulação, etc. Cada um destes exemplos pode ser expresso matematicamente através de uma ou mais equações algébricas sobre as coordenadas cartesianas das partículas, e portanto como uma ou mais equações sobre as coordenadas generalizadas do sistema. Por exemplo, se uma partícula tem coordenadas cartesianas x, y e z, podemos expressar a restrição de que ela está presa a uma esfera de raio r centrada na origem através da equação da esfera $x^2 + y^2 + z^2 = r^2$.

Essa formulação pode ser facilmente generalizada para permitir equações que envolvem explicitamente o tempo. Tais restrições permitem incorporar o controle cinemático do movimento na animação dinâmica, através de equações que especificam as trajetórias das partículas como funções do tempo.

A equação de Lagrange usualmente pressupõe que as coordenadas generalizadas são independentes. Entretanto, na presença de restrições de movimento expressas explicitamente através de equações adicionais, as coordenadas generalizadas não são mais independentes. Em alguns casos, é possível encontrar uma descrição das posições das partículas do sistema em termos de um conjunto menor de coordenadas generalizadas independentes, na qual as restrições são automaticamente satisfeitas. Entretanto, para muitos sistemas de interesse prático pode ser difícil ou mesmo impossível encontrar esse conjunto mínimo de coordenadas. Portanto, um sistema prático de animação dinâmica deve ser capaz de trabalhar com coordenadas generalizadas não independentes. Para tanto, ele precisa utilizar forças especialmente calculadas para impedir (ou minimizar) a violação das restrições, ao longo de toda a evolução do sistema. O sistema de animação descrito neste trabalho é capaz de calcular tais forças, para uma ampla classe de restrições de movimento expressas por equações sobre as coordenadas generalizadas e o tempo.

Uma abordagem para calcular essas forças consiste em introduzir uma "mola virtual" para cada equação de restrição, que penaliza a sua violação em qualquer instante. Outra consiste em calcular forças que produzem as acelerações generalizadas exatamente necessárias para impedir que restrições satisfeitas sejam violadas. Esta, que emprega um método de multiplicadores de Lagrange, é mais precisa e eficiente que aquela, embora seja mais difícil de implementar. Em nosso sistema de animação, implementamos a segunda abordagem.

1.2.3 Física de corpos elásticos

Nos corpos sólidos, em geral as partículas mantêm suas relações de vizinhança ao longo do tempo. Isto nos permite idealizar um corpo sólido, constituído de um número finito de partículas, como uma região de um meio material contínuo. A cada ponto (ou partícula) do interior dessa região, nós associamos grandezas físicas tais como densidade de massa e coeficientes de elasticidade e viscosidade.

Um corpo elástico é caracterizado pela propriedade de que cada fragmento infinitesimal do material que o constitui (e em muitos casos o corpo inteiro) possui uma forma natural de repouso que se manifesta quando não há forças externas agindo sobre o fragmento. Quando o fragmento não está na forma de repouso, surgem forças internas de elasticidade que tentam reconduzí-lo a essa forma. Não consideramos neste trabalho materiais que não retornam à forma de repouso quando as forças externas são removidas, tais como os corpos plásticos ou os que podem sofrer fraturas.

Num corpo viscoelástico, além das forças internas de elasticidade, o material possui uma viscosidade interna. Enquanto se deforma tal corpo, o movimento relativo entre as suas partículas dá origem a forças de atrito, que resistem a esse movimento. Na ausência de forças externas, essas forças dissipativas internas amortecem as vibrações do corpo, transformando parte da energia cinética das partículas em calor.

Dentre os vários tipos de materiais elásticos, restringimo-nos aos que são isotrópicos. Informalmente, um pedaço de material elástico é isotrópico se as forças elásticas, que surgem em resposta a sua deformação, não dependem da orientação do material em relação à direção da deformação.

Para corpos elásticos isotrópicos sujeitos a pequenas deformações, a lei de Hooke nos diz que podemos modelar com boa precisão as propriedades elásticas do material através de dois coeficientes para cada partícula. Nesse caso, a densidade de energia potencial de deformação em um ponto do material pode ser escrita como a soma de um termo volumétrico e um termo de forma, cada um multiplicado pelo respectivo coeficiente de elasticidade. No modelo de Hooke, tais termos são funções quadráticas nas variações de volume e forma, respectivamente, em relação ao volume e forma de repouso na vizinhança de cada ponto. Deste modo, para pequenas deformações as forças elásticas dependem linearmente das medidas de deformação.

Para deformações grandes, entretanto, a lei de Hooke não vale e a relação entre deformação e força elástica pode ser infinitamente complexa. No nosso sistema escolhemos arbitrariamente um modelo não linear para essas forças, no qual a densidade de energia potencial de deformação é escrita como a soma de dois termos similares aos utilizados no modelo de Hooke. Além disso, multiplicamos esses termos por coeficientes tais que a densidade de energia coincide com o modelo de Hooke, quando as deformações são pequenas. Uma característica importante do nosso modelo é que a densidade de energia em um fragmento do material tende para infinito quando o volume do fragmento se aproxima de zero. Isto impede que uma força externa finita seja capaz de comprimir o fragmento a um volume nulo ou negativo.

Considerações análogas valem para as forças de viscosidade, que dependem da deformação e do escoamento (mudança da deformação ao longo do tempo) do material. Para deformações pequenas e escoamento lento, as forças de viscosidade para a maioria dos materiais são determinadas em termos de dois parâmetros: um relacionado com variação de volume, e outro com escoamento a volume constante. No nosso sistema, utilizamos um modelo não linear simples para estas forças, que dá resultados realistas quando o escoamento é lento.

1.2.4 Discretização dos corpos

Neste trabalho utilizamos uma representação dos objetos por elementos finitos. Cada corpo é particionado num número finito de regiões (elementos) de forma tetraédrica.

Neste modelo, a posição de uma determinada partícula de um tetraedro é sempre uma combinação linear fixa das posições dos quatro vértices desse tetraedro. Assim, as coordenadas cartesianas dos vértices de um tetraedro são suficientes para descrever as coordenadas de todas as suas partículas. Como conseqüência, a velocidade e a aceleração de uma partícula qualquer de um tetraedro é uma combinação linear das velocidades e acelerações, respectivamente, dos vértices do tetraedro. Os coeficientes da combinação linear associada a uma partícula do tetraedro são iguais às suas coordenadas baricêntricas em relação a esse tetraedro, as quais não variam no tempo e são utilizadas como “rótulos” para identificar a partícula.

Note que a regra de interpolação acima implica que a deformação do tetraedro deve ser descrita por uma transformação afim dos seus quatro vértices, ou seja, a função que mapeia um tetraedro para outro depende linearmente das posições dos quatro vértices de cada um dos dois tetraedros.

A densidade de massa e os coeficientes de elasticidade e viscosidade são constantes no interior de cada tetraedro, em nosso modelo. Em conseqüência, a densidade de energia potencial de deformação e a potência dissipada pela forças internas de viscosidade são também constantes no interior de cada tetraedro.

Utilizamos as coordenadas cartesianas das partículas que estão nos vértices dos tetraedros como coordenadas generalizadas na equação de Lagrange. Note que um vértice pode ser partilhado por vários tetraedros. Assim, numa discretização que tem n_v vértices distintos, teremos $3n_v$ coordenadas generalizadas e $3n_v$ velocidades generalizadas (que são as velocidades lineares dos vértices distintos).

1.2.5 Deteção e tratamento de colisões

No decorrer da evolução dinâmica do sistema podem ocorrer certos eventos que requerem uma mudança nas equações de movimento e/ou uma mudança instantânea nos valores das velocidades generalizadas. O principal tipo de evento que pode ocorrer são as colisões entre corpos. Outros podem ser, por exemplo, a quebra de contato entre dois corpos, e a remoção ou adição de uma restrição de movimento.

A deteção de colisões exige cuidados especiais, devido ao grande número de combinações de elementos de superfície. Especificamente, se n é o número de elementos de superfície (vértices, faces e arestas), há $O(n^2)$ possíveis pares desses elementos que podem entrar em colisão. Assim, a fim de reduzir o número de pares a testar, utilizamos um algoritmo que seleciona os pares que têm alguma chance de colisão. Esse algoritmo se vale da coerência temporal inerente a um sistema dinâmico, que é devida ao movimento contínuo e, na maior parte do tempo, suave que os corpos apresentam. O resultado prático do uso desse algoritmo é que a complexidade da deteção de colisões se torna quase linear em n, em vez de quadrática.

Em nossa implementação, quando detetamos uma colisão entre dois objetos, introduzimos uma mola virtual entre os pontos em contato. A força dessas molas tende a reduzir a distância entre aqueles pontos quando há interpenetração. A mola é retirada tão logo os corpos comecem a se afastar.

1.3 Trabalho prévio na área

A literatura técnica na área de simulação dinâmica para computação gráfica, especialmente para corpos deformáveis, é relativamente pequena. Por outro lado, a maioria dos artigos não é de fácil digestão, devido ao elevado nível matemático que pressupõem.

Os trabalhos pioneiros nessa área [WB85, AG85] limitaram-se ao movimento de corpos rígidos articulados, sem abordar os problemas de controle de movimento e de impedir a interpenetração entre os corpos. Um dos primeiros sistemas completos de animação de corpos rígidos, capaz de lidar com articulações, restrições de movimento e colisões, é descrito por Cremer e Stewart [CS89]. Os trabalhos produzidos no Brasil a que tivemos acesso foram os de Nedel [Ned93], Rodrigues e Magalhães [Ra93], Musse e Laschuk [ML94], e Horta e Wu [HW95].

1.3.1 Modelagem de corpos deformáveis

Corpos deformáveis, em especial elásticos, foram abordados por Terzopoulos e outros [TPBF87, TW88, TF88], Platt e Barr [PB88], Pentland e Williams [PW89], Witkin e Welch [WW90], e Metaxas e Terzopoulos [MT92].

Uma primeira abordagem para a modelagem de corpos deformáveis consiste em selecionar um certo número de pontos no interior de um corpo, concentrando em cada um a massa do material em sua vizinhança, e ligar pontos vizinhos por dispositivos como molas e amortecedores, que modelam as forças internas de deformação. Embora tal abordagem seja capaz de produzir resultados razoáveis a um custo computacional relativamente baixo, ela não é satisfatória como um modelo definitivo. Em particular, é difícil modelar as forças internas que surgem em resposta a um cisalhamento do material, e ainda mais difícil impedir que um vértice situado no "interior" de um corpo seja deslocado (por exemplo, pela ação de uma força externa) até atravessar a "superfície" do mesmo.

Uma abordagem diferente consiste em calcular as forças elásticas no interior de um corpo deformado a partir de uma função de energia potencial de deformação, que mede a quantidade desse tipo de energia que um corpo armazena quando o mesmo está deformado. O trabalho de Terzopoulos e outros [TPBF87] exemplifica esta abordagem. Inicialmente escolhe-se uma grade regular de partículas na forma de repouso do corpo. A deformação na vizinhança de cada partícula é calculada a partir das posições das partículas vizinhas, pelo método de diferenças finitas. As forças elásticas são obtidas calculando-se as derivadas de uma função de energia elástica que é quadrática nas distâncias entre os vértices vizinhos.

Outra maneira de implementar a abordagem de energias de deformação é através do método dos elementos finitos, que pode ser visto como uma generalização das diferenças finitas para grades irregulares de partículas. Este método foi empregado por Platt e Barr [PB88], que — assim como nós fazemos nesta dissertação — modelaram corpos elásticos isotrópicos utilizando elementos finitos sujeitos a transformações afins. A energia potencial de deformação é calculada a partir de um tensor métrico que descreve a deformação de um fragmento do material em relação a sua forma de repouso. Eles calculam também forças de viscosidade interna que resistem a mudanças no tensor métrico. Entretanto, no modelo de Platt e Barr as forças elásticas são lineares nos componentes do tensor métrico, permitindo que um fragmento possa ser comprimido a volume zero ou negativo por uma força finita. Além disso, eles não separam a medida da deformação de cada fragmento nas duas componentes de mudança uniforme de volume e mudança de forma a volume constante, como nós fazemos, e portanto não conseguem modelar a mesma variedade de materiais.

Para modelar deformações com precisão aceitável, a abordagem baseada em elementos finitos pode exigir um número muito grande de elementos, o que eleva bastante o custo da simulação. Para evitar este problema, Pentland e Williams [PW89] e Witkin e Welch [WW90] propuseram modelos alternativos de deformações com número reduzido de graus de liberdade. No modelo de Witkin e Welch, por exemplo, a forma de um corpo deve ser descrita por uma única transformação polinomial de $\mathbb{R}^3$ para $\mathbb{R}^3$, aplicada à sua forma natural. As abordagens que restringem o número de graus de liberdade dos corpos têm a vantagem de serem mais fáceis de resolver por métodos numéricos, mas dificultam consideravelmente a modelagem e simulação de formas geométricas arbitrárias.

1.3.2 Restrições de movimento

O uso de restrições para controle de movimento e montagem de estruturas de objetos rígidos aparece em vários artigos [IC87, WK88, BB88, PB88, WGW90].

Metaxas e Terzopoulos [MT92] descrevem restrições gerais sobre a posição e forma de corpos deformáveis, descrevendo em particular um método de estabilização de vínculos (que nós também utilizamos) para dirigir o sistema a estados que satisfazem restrições inicialmente não satisfeitas.

Em um artigo recente, Baraff [Bar96] descreve uma técnica para otimizar o cálculo de forças de reação associadas a restrições.

1.3.3 Contatos entre corpos

Uma das primeiras abordagens dos problemas de deteção e tratamento de colisões entre corpos rígidos e/ou deformáveis foi a de Moore e Wilhelms [MW88]. Métodos recentes para a deteção eficiente de colisões entre corpos rígidos foram descritos por Lin e outros [LM94, PML95]. Métodos para o tratamento de colisões e contatos entre corpos, incluindo atrito, foram desenvolvidos por Hahn [Hah88], Baraff [Bar89, Bar91, BW92, Bar93, Bar94] e Mirtich e Canny [MC94].

No tratamento de contatos entre corpos, Baraff segue uma abordagem exata, utilizando uma combinação de equações e inequações sobre as acelerações dos objetos e as forças de contato para calcular as forças (em contatos de repouso, deslizamento, rolamento e colisão) exatamente necessárias para impedir qualquer interpenetração.

A abordagem de Mirtich e Canny é menos precisa, modelando todos os tipos de contato por séries de impulsos instantâneos, separados por períodos finitos de tempo, e aplicados a um par de pontos de contato por vez. Segundo os autores, esses impulsos são capazes de modelar corretamente as forças de atrito estático e dinâmico, sem que o número de impulsos necessários seja excessivamente elevado.

1.4 Organização da dissertação

O capítulo 2 desenvolve as equações de movimento para um sistema dinâmico geral, derivadas a partir da equação de Lagrange.

O capítulo 3 descreve o tratamento de restrições de movimento, e mostra como calcular as forças de reação correspondentes pelo método dos multiplicadores de Lagrange.

No capítulo 4 descrevemos corpos elásticos como meios contínuos, derivando fórmulas para a energia cinética, a energia potencial elástica e a potência dissipada em um corpo elástico ou viscoelástico. O significado físico dos coeficientes de elasticidade e de viscosidade, para pequenas deformações, é explicado neste capítulo.

No capítulo 5 descrevemos a modelagem por elementos finitos adotada para representar os corpos elásticos, com elementos tetraédricos sujeitos a deformações lineares, e as fórmulas para cálculo dos tensores de deformação e energia cinética dos objetos.

No capítulo 6, descrevemos os detalhes algorítmicos do cálculo das acelerações, das forças internas e de reação, e da integração numérica das equações de movimento.

O capítulo 7 trata da deteção e tratamento de eventos discretos, em particular a deteção eficiente de colisões. Nele descrevemos uma abordagem original para a deteção geral de eventos discretos. Para impedir interpenetração, descrevemos o uso de molas nos pontos de contato.

O capítulo 8 descreve sucintamente a estrutura geral do sistema de animação implementado, a funcionalidade das bibliotecas e programas codificados, e o formato dos vários arquivos de modelo.

No capítulo 9 apresentamos várias animações, exibindo quadros selecionados de cada uma e os respectivos gráficos de energia, junto a dados estatísticos de tempo computacional despendido, número de elementos finitos utilizados, etc.

O capítulo 10 apresenta nossas conclusões e indica alguns possíveis tópicos de pesquisa para a extensão dos resultados.

1.5 Notação

Todo ponto $\mathbf{u}$ do $\mathbb{R}^n$, para $n \geq 2$, será interpretado como um vetor coluna em fórmulas matriciais, exceto quando utilizarmos a notação $\mathbf{u}^\top$. Denotamos por $[u_1, u_2, \ldots, u_n]$ o vetor $\mathbf{u}$, cujos elementos (ou componentes) escalares são $u_1, u_2, \ldots, u_n$.

Sejam $\mathbf{u}$ e $\mathbf{v}$ vetores. Vamos denotar o módulo de $\mathbf{u}$ por $|\mathbf{u}|$ e o produto escalar de $\mathbf{u}$ e $\mathbf{v}$ por $\mathbf{u} \cdot \mathbf{v}$. O produto $\mathbf{u} \cdot \mathbf{u}$ será também indicado por $\mathbf{u}^2$. Se $\mathbf{A}$ é uma matriz quadrada de componentes reais, indicaremos o elemento de $\mathbf{A}$ na linha i e na coluna j por A_{ij}, e o determinante de $\mathbf{A}$ por $|\mathbf{A}|$.

Seja $f : D \to \mathbb{R}^n$ uma função tal que $D \subseteq \mathbb{R}^m$ e $m, n \geq 1$. A imagem de f será indicada por $f(D)$. A composição de duas funções $g : A \to B$ e $f : C \to D$ tal que $g(A) \subseteq C$ é a função $f \circ g : A \to D$, tal que $(f \circ g)(x) = f(g(x))$ para todo $x \in A$.

No contexto da complexidade de algoritmos, a notação $O(f(n))$ se refere ao comportamento da quantidade em estudo quando o tamanho n do problema tende a infinito. No contexto de análise diferencial, porém, a notação $O(f(x))$ descreve o comportamento quando a variável x tende a zero.

Capítulo 2

Física do movimento

"A mecânica de um sistema é determinada quando sua energia potencial é dada como função de sua configuração. Ora, se as forças atuantes forem de tal tipo que possam garantir a manutenção de certas características de ordem da configuração do sistema, torna-se possível descrever a configuração com suficiente precisão com um número relativamente pequeno de variáveis de configuração q_r; a energia potencial é considerada apenas na medida em que é dependente dessas *variáveis."*
Albert Einstein

2.1 Sistemas dinâmicos

Na mecânica de Newton, um *sistema dinâmico* consiste em um conjunto finito de partículas materiais de diâmetro desprezível e massa constante, com posição e velocidade definidas no espaço $\mathbb{R}^3$. O movimento das partículas ao longo do tempo é determinado pelas forças internas, que as próprias partículas exercem umas sobre as outras, e pelas forças externas, exercidas sobre as partículas por agentes que não pertencem ao sistema.

2.1.1 Grandezas

Uma *grandeza*, em um dado sistema dinâmico, é uma função que associa a cada instante o valor de alguma propriedade numérica desse sistema. Usaremos letras itálicas (x) para grandezas escalares, e negrito ($\mathbf{x}$) para grandezas vetoriais. Note que cada componente de uma grandeza vetorial $\mathbf{x} = [x_1, \ldots, x_m]$, para $m > 1$, é uma grandeza escalar. Exemplos de grandezas são a posição e a velocidade de uma partícula, a densidade de massa em cada ponto de um meio material contínuo, a força de atrito entre dois corpos, a energia cinética do sistema, etc.

Denotaremos as derivadas primeira e segunda (em relação ao tempo) de uma grandeza escalar x por $\dot{x}$ e $\ddot{x}$, respectivamente. No caso de grandezas vetoriais $\mathbf{x}$, a operação de derivação se aplica a cada uma de suas componentes. Por exemplo, se $\mathbf{x} \in \mathbb{R}^3$ é a posição de uma partícula, $\dot{\mathbf{x}}$ é a sua velocidade e $\ddot{\mathbf{x}}$ a sua aceleração.

2.1.2 Configuração e estado do sistema

A *configuração* de um sistema dinâmico em um dado instante é o conjunto das posições de todas as suas partículas nesse instante. O *estado* do sistema em um dado instante, por sua vez, é o conjunto das posições e das velocidades das partículas nesse mesmo instante.

Considere um sistema dinâmico constituído de p partículas. Um *grau de liberdade* desse sistema é uma maneira, única e independente das demais, pela qual se pode alterar a sua configuração. Para um sistema dinâmico irrestrito, isto é, um no qual é possível alterar a posição de uma partícula qualquer independentemente das demais, há $3p$ graus de liberdade.

Na prática, seria caro demais descrever diretamente o movimento de cada partícula, pois o seu número é excessivamente alto nos sistemas de interesse. Felizmente, em sistemas nos quais os objetos materiais são corpos sólidos, a grande maioria dos graus de liberdade tem pouquíssima ou nenhuma influência nos movimentos observados dos corpos. Por exemplo, um corpo altamente rígido — como uma barra de aço — movimenta-se sem mudar perceptivelmente sua forma e tamanho. Na verdade, as partículas da barra possuem sempre um pequeno movimento relativo, mas que é imperceptível sem uma observação muito cuidadosa. Portanto, podemos idealizar a barra de aço como um corpo absolutamente rígido, isto é, um corpo no qual a distância entre qualquer par de partículas é fixa. Em outras palavras, em tais corpos os movimentos relativos entre as partículas são negligíveis.

Desse modo, em muitas situações práticas podemos considerar apenas um número reduzido de graus de liberdade significativos, que são suficientes para descrever com precisão aceitável a configuração do sistema em qualquer instante. No caso da barra de aço, todas as configurações macroscópicas possíveis podem ser atingidas por meio de seis movimentos independentes no sistema inercial de coordenadas: uma rotação em torno de cada um dos três eixos, seguida de uma translação ao longo de cada um dos mesmos.

Assim, dado um sistema dinâmico com um número reduzido de graus de liberdade significativos, vamos trabalhar sempre com uma aproximação da sua configuração e do seu estado em qualquer instante. Contudo, para facilitar a linguagem, diremos a partir daqui que essas aproximações são a própria configuração e o próprio estado do sistema.

2.1.3 Coordenadas generalizadas

Uma vez determinados os graus de liberdade considerados relevantes para um dado sistema dinâmico, precisamos estabelecer um conjunto não redundante de parâmetros independentes que permitam expressar matematicamente a configuração do sistema. Idealmente, a quantidade de tais parâmetros deve ser igual ao número de graus de liberdade relevantes. No exemplo da barra de aço descrito acima, os seis graus de liberdade relevantes podem ser parametrizados, por exemplo, através das três coordenadas cartesianas do centro de gravidade da barra, e dos três ângulos de Euler [Sym82] que descrevem a orientação de um sistema de coordenadas fixado à barra em relação ao sistema inercial.

Suponha que a configuração do sistema dinâmico pode ser completamente especificada por um vetor $\mathbf{q} = [q_1, \ldots, q_n]$ de grandezas independentes e não redundantes (ou seja, para toda configuração possível do sistema, existe um único vetor de configuração $\mathbf{q}$, e o conjunto das configurações possíveis não pode ser descrito, sem perda de precisão, por um número menor de coordenadas generalizadas). Nesse caso, as velocidades das partículas podem ser calculadas a partir de $\mathbf{q}$ e do vetor $\dot{\mathbf{q}} = [\dot{q}_1, \ldots, \dot{q}_n]$ das respectivas derivadas primeiras dos elementos de $\mathbf{q}$. Os n parâmetros q_i são as *coordenadas generalizadas* do sistema, e suas respectivas derivadas primeiras $\dot{q}_i$ são as *velocidades generalizadas* do sistema. As derivadas segundas $\ddot{q}_i$ dessas grandezas, por sua vez, são as *acelerações generalizadas* do sistema.

A fim de simplificar a linguagem, vamos confundir a configuração do sistema com o próprio vetor $\mathbf{q}$, e o estado do sistema com o próprio par ordenado $(\mathbf{q}, \dot{\mathbf{q}})$. No restante desta dissertação, sempre que considerarmos um sistema dinâmico, n indicará o número de coordenadas generalizadas do mesmo.

2.2 Força e energia

2.2.1 Forças generalizadas

Seja p o número de partículas em um sistema dinâmico e $l = 3p$. Por definição, as coordenadas cartesianas $x_1, x_2, \ldots, x_l$ das partículas podem ser calculadas em todo instante t a partir das coordenadas generalizadas $q_1, q_2, \ldots, q_n$ através de funções $\chi_1, \chi_2, \ldots, \chi_l$ tais que $x_j(t) = \chi_j(q_1(t), \ldots, q_n(t))$, para $j = 1, \ldots, l$.

Seja f_j uma força agindo sobre a coordenada cartesiana x_j, para $j = 1, \ldots, l$. O trabalho total realizado pelas forças f_j em um deslocamento δ_j de cada coordenada x_j, supondo forças constantes durante esse deslocamento, é

$$T = \sum_{j=1}^{l} f_j\, \delta_j$$

Em termos de coordenadas generalizadas, o deslocamento δ_j é dado por

$$\delta_j = \sum_{i=1}^{n} \frac{\partial \chi_j}{\partial q_i} \bar{\delta}_i$$

onde $\bar{\delta}_i$ é um deslocamento da coordenada generalizada q_i. Podemos então reescrever o trabalho realizado como

$$T = \sum_{i=1}^{n} \bar{f}_i \, \bar{\delta}_i$$

onde cada coeficiente $\bar{f}_i$ é dado por

$$\bar{f}_i = \sum_{j=1}^{l} f_j \frac{\partial \chi_j}{\partial q_i}$$

A grandeza $\bar{f}_i$ é a *força generalizada* que age sobre a coordenada q_i. Note que o significado de uma força generalizada $\bar{f}_i$ depende do significado das coordenadas generalizadas q_i, que é em princípio arbitrário.

2.2.2 Energia cinética

A energia cinética do sistema é a soma das energias cinéticas de todas as suas partículas, e portanto pode ser descrita como uma função K de $\mathbf{q}$ e $\dot{\mathbf{q}}$.

2.2.3 Energia potencial e forças conservativas

Muitos sistemas dinâmicos apresentam um processo físico que armazena uma certa quantidade de energia potencial e_p, resultante de interações entre as partículas que dependem apenas das suas posições relativas em um dado instante. Essa energia pode, em princípio, ser convertida em energia cinética e influir no movimento. Uma vez que a grandeza e_p depende apenas das posições das partículas, o seu valor em qualquer instante pode ser determinado através de uma função P das coordenadas generalizadas $\mathbf{q}$, tal que $e_p(t) = P(\mathbf{q}(t))$ em cada instante t.

A energia potencial e_p induz forças generalizadas f_i^c (as *forças conservativas internas*) que, na ausência de outras forças, conservam a soma das energias cinética e potencial do sistema $(K + P)$. Estas forças são calculadas como

$$f_i^c(\mathbf{q}) = -\frac{\partial P}{\partial q_i}(\mathbf{q})$$

para $i = 1, \ldots, n$.

2.2.4 Potência dissipada e forças dissipativas

Em geral os sistemas dinâmicos apresentam também um processo físico que dissipa parte de sua energia cinética, devido ao atrito entre as partículas. A energia dissipada é na verdade convertida em energia térmica, isto é, movimentos microscópicos e desordenados das partículas; entretanto, a energia cinética desses movimentos não pode ser normalmente reconvertida em energia cinética macroscópica, e portanto podemos considerá-la perdida.

O atrito provém de interações entre as partículas que dependem das suas velocidades relativas, além de dependerem das posições relativas entre as mesmas. Assim, a quantidade de energia dissipada por unidade de tempo (ou *potência dissipada*) pode ser medida por uma função W de $\mathbf{q}$ e $\dot{\mathbf{q}}$. Essa perda de energia induz forças generalizadas f_i^{d} (as *forças dissipativas internas*), tais que

$$W(\mathbf{q},\dot{\mathbf{q}}) = -\mathbf{f}^{\mathrm{d}}(\mathbf{q},\dot{\mathbf{q}}) \cdot \dot{\mathbf{q}}$$

onde $\mathbf{f}^{\mathrm{d}}(\mathbf{q},\dot{\mathbf{q}}) = [f_1^{\mathrm{d}}(\mathbf{q},\dot{\mathbf{q}}),\ldots,f_n^{\mathrm{d}}(\mathbf{q},\dot{\mathbf{q}})]$.

Neste trabalho, restringimos W a funções que são quadráticas nas velocidades generalizadas $\dot{\mathbf{q}}$. Ou seja, W é da forma

$$W(\mathbf{q},\dot{\mathbf{q}}) = \frac{\dot{\mathbf{q}}^{\top}\mathbf{A}(\mathbf{q})\dot{\mathbf{q}}}{2}$$

onde $\mathbf{A}$ é uma matriz simétrica que depende apenas das coordenadas generalizadas $\mathbf{q}$. Portanto, as forças dissipativas internas são calculadas como

$$f_i^{\mathrm{d}}(\mathbf{q},\dot{\mathbf{q}}) = -\frac{1}{2}\frac{\partial W}{\partial \dot{q}_i}(\mathbf{q},\dot{\mathbf{q}})$$

para $i = 1,\ldots,n$. Note que estas forças, embora lineares em $\dot{\mathbf{q}}$, podem ser não lineares em $\mathbf{q}$.

2.2.5 Conservação da energia

Por definição, a energia mecânica do sistema em um dado instante é a soma de sua energia cinética e sua energia potencial nesse instante, ou seja, $K + P$.

Na ausência de forças externas, a energia mecânica do sistema permanece constante enquanto a potência dissipada W for nula. Se W é não nula durante um certo intervalo de tempo, a quantidade de energia é continuamente reduzida ao longo desse intervalo. A energia do sistema ao final de um dado intervalo de tempo é a soma da sua energia no início desse intervalo com o trabalho realizado durante o mesmo por quaisquer forças externas atuantes no sistema, menos a integral de W sobre esse intervalo.

2.3 As equações de movimento

2.3.1 A equação de Lagrange

Formalmente, o movimento das partículas de um sistema dinâmico, a partir de um dado estado inicial, pode ser obtido como a solução de um sistema de equações diferenciais, expressas nas coordenadas e velocidades generalizadas do sistema.

Essas equações diferenciais podem ser obtidas da *equação de Lagrange* [Sym82, Gol80]:

$$\frac{d}{dt}\left(\frac{\partial K}{\partial \dot{q}_i}(\mathbf{q},\dot{\mathbf{q}})\right) - \frac{\partial K}{\partial q_i}(\mathbf{q},\dot{\mathbf{q}}) + \frac{1}{2}\frac{\partial W}{\partial \dot{q}_i}(\mathbf{q},\dot{\mathbf{q}}) + \frac{\partial P}{\partial q_i}(\mathbf{q}) = f_i^{\mathrm{e}}(\mathbf{q},\dot{\mathbf{q}},t) \qquad i = 1,\ldots,n \quad (2.1)$$

onde f_i^{e} é uma força generalizada externa que atua sobre a coordenada q_i, dada por uma função arbitrária do estado do sistema e do tempo. Esta é basicamente a lei fundamental da dinâmica, "a força total sobre uma partícula é igual ao produto da sua massa pela sua aceleração instantânea", reformulada em termos de coordenadas e forças generalizadas.

Por conveniência, vamos considerar como forças internas do sistema apenas as forças f_i^{c} e f_i^{d}, para as quais temos as funções escalares P e W. Todas as demais categorias de forças serão consideradas "forças externas", seja porque elas se originam de processos físicos externos ao próprio sistema, como a força de gravidade resultante de um campo gravitacional dado, seja porque é mais conveniente calculá-las como termos das funções f_i^{e}, como no caso das forças de contato entre corpos. (A força de gravidade, em particular, poderia ser considerada como interna, desde que fosse calculada através de uma função de energia potencial gravitacional adicionada a P.)

O primeiro termo do lado esquerdo de (2.1) pode ser expandido na forma

$$\frac{d}{dt}\left(\frac{\partial K}{\partial \dot{q}_i}(\mathbf{q},\dot{\mathbf{q}})\right) = \sum_{j=1}^{n}\frac{\partial^2 K}{\partial \dot{q}_i \partial \dot{q}_j}(\mathbf{q},\dot{\mathbf{q}})\,\ddot{q}_j + \sum_{j=1}^{n}\frac{\partial^2 K}{\partial \dot{q}_i \partial q_j}(\mathbf{q},\dot{\mathbf{q}})\,\dot{q}_j$$

na qual surgem as acelerações generalizadas $\ddot{q}_j$.

Em geral, se as coordenadas cartesianas das partículas podem ser descritas como funções lineares nas coordenadas generalizadas q_i, o segundo termo de (2.1) é nulo, porque então as velocidades cartesianas das partículas (e portanto também a energia cinética) dependem apenas das velocidades generalizadas $\dot{q}_i$.

As forças ditas "fictícias", como a força centrífuga e a força de Coriolis [Sym82, Gol80], surgem quando as coordenadas cartesianas das partículas são funções não-lineares das coordenadas generalizadas do sistema.

O terceiro termo da equação de Lagrange dá os valores das forças dissipativas internas f_i^{d}, induzidas por uma função W de potência dissipada, ao passo que o quarto termo dá os valores das forças conservativas internas f_i^{c}, induzidas por uma função P de energia potencial.

2.3.2 Formulação matricial

Podemos reformular o sistema de equações diferenciais (2.1) em termos matriciais, na forma de um sistema linear nas acelerações generalizadas $\ddot{\mathbf{q}} = [\ddot{q}_1, \ddot{q}_2, \ldots, \ddot{q}_n]$.

Para tanto, definimos a *matriz de massa generalizada* $\mathbf{M}$, dada por

$$M_{ij} = \frac{\partial^2 K}{\partial \dot{q}_i \partial \dot{q}_j}(\mathbf{q}, \dot{\mathbf{q}})$$

para $i, j = 1, \ldots, n$.

Os outros termos na equação (2.1) são forças. Para eles, definimos os vetores $\mathbf{f}^{\mathrm{f}}$, $\mathbf{f}^{\mathrm{d}}$, $\mathbf{f}^{\mathrm{c}}$ e $\mathbf{f}^{\mathrm{e}}$ de forças generalizadas fictícias, dissipativas, conservativas e externas, respectivamente:

$$\begin{aligned} f_i^{\mathrm{f}} &= \frac{\partial K}{\partial q_i}(\mathbf{q}, \dot{\mathbf{q}}) - \sum_{j=1}^{n} \frac{\partial^2 K}{\partial \dot{q}_i \partial q_j}(\mathbf{q}, \dot{\mathbf{q}})\, \dot{q}_j \\ f_i^{\mathrm{d}} &= -\frac{1}{2}\frac{\partial W}{\partial \dot{q}_i}(\mathbf{q}, \dot{\mathbf{q}}) \\ f_i^{\mathrm{c}} &= -\frac{\partial P}{\partial q_i}(\mathbf{q}) \end{aligned}$$

para $i = 1, \ldots, n$. Definimos também o vetor de *forças diretas* $\mathbf{f}$ como a soma das demais forças, ou seja:

$$\mathbf{f} = \mathbf{f}^{\mathrm{f}} + \mathbf{f}^{\mathrm{d}} + \mathbf{f}^{\mathrm{c}} + \mathbf{f}^{\mathrm{e}}$$

Podemos então escrever

$$\mathbf{M}\ddot{\mathbf{q}} = \mathbf{f} \tag{2.2}$$

Note que esta equação representa um sistema de equações diferenciais ordinárias acopladas, de segunda ordem. Dados os valores de $\mathbf{q}$ e $\dot{\mathbf{q}}$ em um instante t, e conhecidas as forças externas $\mathbf{f}^{\mathrm{e}}$ nesse instante, essa equação nos permite calcular os valores das acelerações generalizadas $\ddot{\mathbf{q}}$ em t. Essa equação pode também ser integrada numericamente para determinar a evolução do estado $(\mathbf{q}, \dot{\mathbf{q}})$ ao longo do tempo.

Capítulo 3

Restrições de movimento

No capítulo anterior, supomos que as coordenadas generalizadas são sempre independentes e não redundantes. Esta suposição é válida quando as configurações que podem ser descritas através do vetor $\mathbf{q}$ são atingíveis pelas partículas do sistema, dado tempo e energia suficientes. Por exemplo, para um sistema com uma única partícula u movendo-se sem restrições no espaço, as coordenadas cartesianas $[x, y, z]$ de u representam um conjunto possível de coordenadas generalizadas independentes e não redundantes.

Em contraste, se nem todas as configurações representadas por vetores $\mathbf{q}$ podem ser atingidas pelo sistema, então não podemos, em princípio, dizer que as configurações que *podem* ser atingidas pelo sistema não podem ser descritas por um número menor de coordenadas generalizadas. Por exemplo, se a partícula u está restrita a um plano π, então as coordenadas $[x, y, z]$ não são independentes, existindo porém um conjunto de *duas* coordenadas generalizadas, relativas a um sistema cartesiano fixo no plano π, que descreve completamente as posições da partícula u sobre aquele plano. Por outro lado, seria impossível descrever a posição da partícula u com uma só coordenada generalizada (a menos que houvesse outra restrição sobre o movimento da mesma). Portanto, neste sistema há exatamente dois graus de liberdade observáveis.

Como demonstra o exemplo acima, às vezes é possível modelar restrições sobre o movimento do sistema simplesmente escolhendo-se um conjunto apropriado de coordenadas generalizadas independentes cujo número é igual ao número de graus de liberdade do sistema. Entretanto, em sistemas com restrições mais complexas pode ser impraticável ou mesmo impossível encontrar um conjunto independente de coordenadas. Por exemplo, considere um sistema com uma única partícula restrita à superfície de uma esfera. Embora o conjunto de posições dessa partícula em $\mathbb{R}^3$ seja bidimensional, não é possível especificar a sua posição sobre a esfera com apenas duas coordenadas, de maneira única e contínua.

Portanto, em geral temos de permitir um conjunto *dependente* de coordenadas generalizadas, no sentido de que nem todo vetor $\mathbf{q}$ corresponde a uma configuração atingível pelo sistema.

3.1 Modelagem de coordenadas dependentes

Neste capítulo, nos restringimos a sistemas onde as dependências entre as coordenadas generalizadas podem ser descritas por equações algébricas sobre $\mathbf{q}$, $\dot{\mathbf{q}}$, $\ddot{\mathbf{q}}$ e t, como a equação do plano ou da esfera nos exemplos acima. Especificamente, uma restrição de movimento é descrita por uma ou mais equações da forma $H(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, t) = 0$, onde H é uma função algébrica escalar. Cada equação deste tipo é dita um *vínculo* do sistema.[1]

No contexto de animação dinâmica, tais restrições podem ser utilizadas para muitos fins. Por exemplo, pode-se usá-las para fixar uma partícula a uma posição, curva ou superfície no espaço, estabelecer articulações entre dois corpos, e assim por diante.

3.1.1 Vínculos holonômicos

Na prática, as restrições de movimento mais comuns podem ser definidas através de vínculos que dependem de $\mathbf{q}$ e t apenas. Tais vínculos são ditos *holonômicos*. Diremos que uma restrição é holonômica se seus vínculos forem holonômicos.

Em alguns casos particulares, um vínculo que depende de $\dot{\mathbf{q}}$ ou $\ddot{\mathbf{q}}$ pode ser reduzido a um vínculo holonômico por integração analítica. Por exemplo, para manter uma velocidade generalizada $\dot{q}_i$ constante, usaríamos o vínculo

$$\dot{q}_i - v = 0$$

Esta equação pode ser integrada em t, produzindo o vínculo holonômico equivalente

$$q_i - c - v(t - t_0) = 0$$

onde c é o valor de q_i no instante t_0.

No caso geral, entretanto, isto não é possível. Por exemplo, considere a restrição que obriga a velocidade $\mathbf{v} = [v_x, v_y, v_z]$ de uma partícula u a ter módulo constante. O vínculo não holonômico que descreve esta restrição é $v_x^2 + v_y^2 + v_z^2 - c^2 = 0$, onde c é o módulo da velocidade inicial de u. Note que esta partícula poderá percorrer qualquer trajetória em $\mathbb{R}^3$ sem violar a restrição, e portanto este vínculo não pode ser reescrito como uma restrição envolvendo $\mathbf{q}$ e t apenas.

3.1.2 Espaço de configurações

O *espaço de configurações* de um sistema dinâmico é o conjunto de todos os possíveis vetores $\mathbf{q}$ de configuração do sistema, que satisfazem ou não as restrições impostas ao

[1] Em textos de Física, o termo "vínculo" usualmente refere-se a uma condição expressa por uma equação ou uma inequação; vamos, entretanto, utilizar este termo para nos referir tão somente a condições expressas por equações.

mesmo; ou seja, é o próprio espaço $\mathbb{R}^n$. Uma configuração é *válida* quando ela satisfaz a todos os vínculos. A evolução do sistema ao longo do tempo é uma trajetória nesse espaço, que inclui apenas configurações válidas.

Um vínculo holonômico $H(\mathbf{q}, t) = 0$ determina, para cada instante t, uma superfície $(n-1)$-dimensional no espaço das configurações $\mathbf{q}$. De modo geral, o vínculo elimina um grau de liberdade do sistema, obrigando a configuração $\mathbf{q}$ a permanecer sobre aquela superfície em cada instante.

No caso do vínculo não depender do tempo, essa *superfície de vínculo* contém obrigatoriamente a trajetória do sistema. A figura (3.1) ilustra a trajetória do sistema sobre a superfície de um vínculo $H(\mathbf{q}) = 0$ no espaço de configurações para $n = 3$.

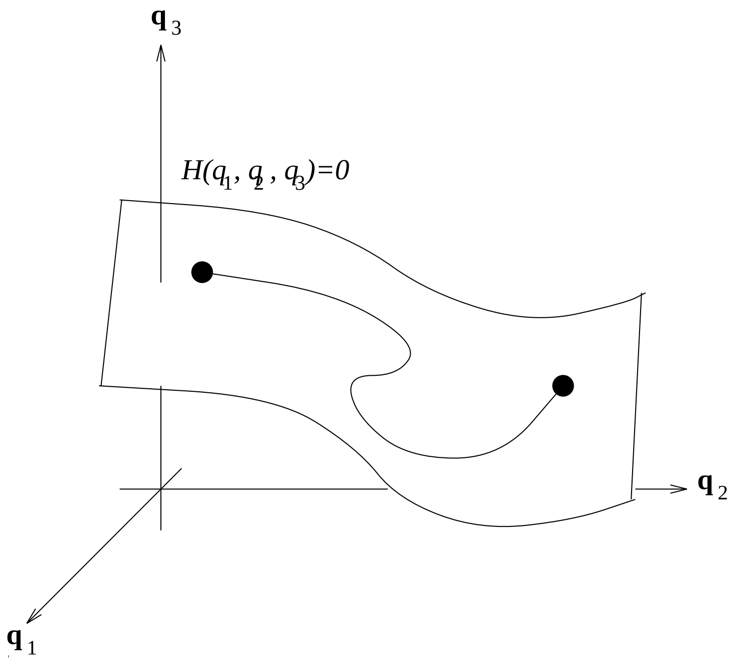


Figura 3.1: Trajetória sobre uma superfície de vínculo no $\mathbb{R}^3$

No caso de um sistema com dois ou mais vínculos, a trajetória permanece sobre a superfície de cada vínculo. Portanto, o espaço de configurações válidas é a interseção de todas as superfícies de vínculo.

3.2 Modelagem das restrições

Num sistema com restrições, além das forças internas e externas que descrevemos no capítulo anterior (as forças diretas), deve haver forças adicionais cuja única função é manter as restrições. Para que um vínculo seja satisfeito, o processo físico que dá origem ao mesmo deve aplicar uma força generalizada no sistema. Por exemplo, no caso de duas partículas de posições $\mathbf{u}$ e $\mathbf{v}$ ligadas por uma barra rígida de comprimento L, a manutenção do vínculo $|\mathbf{u} - \mathbf{v}| = L$ exige que a barra aplique forças centrípetas nas partículas. A força sobre um vínculo que é responsável por mantê-lo satisfeito é dita a sua *força de vínculo*.

Um exemplo de força de vínculo são as forças que agem sobre um corpo fixado a uma superfície rígida, impedindo a separação nos pontos de contato. Outro exemplo são as forças que, num corpo articulado por junções mecânicas, mantêm as diferentes partes conectadas. Ainda outro exemplo são as tensões internas em um corpo rígido, que mantêm fixas as distâncias entre partículas vizinhas.

3.2.1 Método das forças de penalidade

Uma maneira simples de obrigar a trajetória calculada do sistema a respeitar aproximadamente as restrições é substituir cada vínculo $H(\mathbf{q}, t) = 0$ por uma *força de penalidade* que, quando o vínculo não está satisfeito, tenta reduzir a violação do mesmo. Esta abordagem é simples, mas tem o defeito inerente de permitir pequenas violações das restrições.

A força de penalidade $\mathbf{f}^{\mathrm{p}}$ que substitui um vínculo $H(\mathbf{q}, t) = 0$ pode ser definida como a força de uma mola imaginária que liga a configuração corrente do sistema ao ponto mais próximo na superfície do vínculo — veja a figura (3.2).

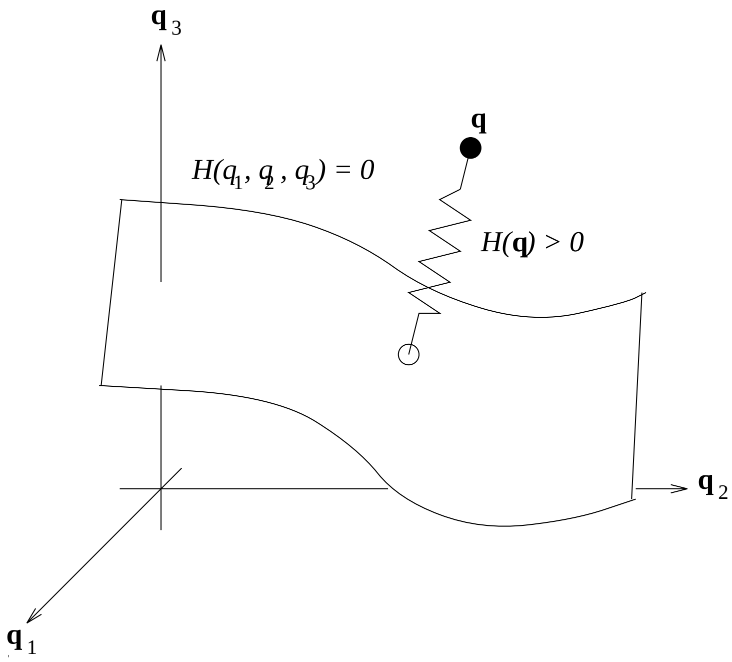


Figura 3.2: Uma “mola” para um vínculo não satisfeito

A energia potencial de deformação dessa mola pode ser calculada como $K\,H(\mathbf{q}(t),t)^2/2$, onde K é uma constante positiva. Assim, a força de penalidade sobre cada coordenada q_i é calculada como

$$f_i^{\mathrm{p}}(\mathbf{q}(t),t) = -K\,H(\mathbf{q}(t),t)\,\frac{\partial H}{\partial q_i}(\mathbf{q}(t),t)$$

para $i = 1,\ldots,n$. Para configurações suficientemente próximas da superfície de vínculo, esta força é proporcional à distância da superfície, e dirigida perpendicularmente em direção à mesma.

Usualmente, as forças de penalidade são intuitivas e fáceis de calcular. Porém, em certas situações em que há forças externas ou internas muito intensas ou velocidades muito altas, a constante K deve ser também muito alta, para que a força de penalidade sobre um dado vínculo consiga manter a violação do mesmo (o valor de H) abaixo de um valor máximo aceitável. Nesse caso, a intensidade das forças de penalidade tende a variar muito rapidamente com variações pequenas de $\mathbf{q}$, tornando as equações de movimento do sistema "rígidas" (*stiff*), o que causa grandes dificuldades em sua integração numérica. Na prática, a integração de tais equações diferenciais pode se tornar instável, impossibilitando completamente a simulação correta do movimento.

3.2.2 Forma diferencial dos vínculos

Em nosso simulador, implementamos os vínculos de maneira mais precisa incorporando as restrições ao próprio sistema de equações diferenciais de movimento. Ou seja, a cada passo da integração, calcula-se as acelerações $\ddot{\mathbf{q}}$ de modo a não violar nenhum dos vínculos nos instantes seguintes.

Para esta análise convém definir a grandeza $h(t) = H(\mathbf{q}(t),\dot{\mathbf{q}}(t),\ddot{\mathbf{q}}(t),t)$, para todo instante t. A função h é dita o *desvio* da trajetória do sistema em relação ao vínculo $H(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}},t) = 0$. Portanto, o vínculo $H(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}},t) = 0$ está satisfeito ao longo de um intervalo de tempo $I = [t_a,t_b]$ se e somente se $h(t) = 0$ para todo $t \in I$.

No caso de vínculos $H(\mathbf{q},\dot{\mathbf{q}},\ddot{\mathbf{q}},t) = 0$ que dependem explicitamente de $\ddot{\mathbf{q}}$, basta adicionar a equação de cada vínculo ao sistema de equações (2.1), desde que a função H seja estritamente linear nas acelerações generalizadas $\ddot{q}_i$. Nesse caso, pode-se reescrever a equação de vínculo na forma

$$\sum_{i=1}^{n} \frac{\partial H}{\partial \ddot{q}_i}(\mathbf{q},\dot{\mathbf{q}},t)\,\ddot{q}_i = H^*(\mathbf{q},\dot{\mathbf{q}},t)$$

onde H^* coleta os termos de H que não dependem de $\ddot{\mathbf{q}}$.

No caso de um vínculo $H(\mathbf{q},\dot{\mathbf{q}},t) = 0$ que depende explicitamente de $\dot{\mathbf{q}}$ mas não de $\ddot{\mathbf{q}}$, para obter uma equação sobre $\ddot{\mathbf{q}}$ temos que derivar o desvio h correspondente em relação ao tempo. Se $h(t_a) = 0$ no instante inicial t_a do intervalo I, é necessário e suficiente manter

$\dot{h}(t) = 0$ para todo $t \in I$, para que o vínculo seja satisfeito ao longo de I. Uma vez que

$$\dot{h}(t) = \sum_{i=1}^{n} \frac{\partial H}{\partial q_i}(\mathbf{q}, \dot{\mathbf{q}}, t)\, \dot{q}_i + \sum_{i=1}^{n} \frac{\partial H}{\partial \dot{q}_i}(\mathbf{q}, \dot{\mathbf{q}}, t)\, \ddot{q}_i + \frac{\partial H}{\partial t}(\mathbf{q}, \dot{\mathbf{q}}, t)$$

basta acrescentar ao sistema de equações de movimento a equação

$$\sum_{i=1}^{n} \frac{\partial H}{\partial \dot{q}_i}(\mathbf{q}, \dot{\mathbf{q}}, t)\, \ddot{q}_i = -\sum_{i=1}^{n} \frac{\partial H}{\partial q_i}(\mathbf{q}, \dot{\mathbf{q}}, t)\, \dot{q}_i - \frac{\partial H}{\partial t}(\mathbf{q}, \dot{\mathbf{q}}, t)$$

No caso mais comum de um vínculo holonômico $H(\mathbf{q}, t) = 0$, precisamos derivar a condição $h(t) = 0$ duas vezes para obter uma equação envolvendo $\ddot{\mathbf{q}}$. Suponha que $h(t_a) = 0$ e $\dot{h}(t_a) = 0$ no instante inicial t_a. Para que os vínculos continuem satisfeitos até o instante final t_b, é necessário e suficiente que a derivada segunda $\ddot{h}$ de cada função de desvio h seja nula ao longo de I.

A derivada primeira do desvio h é dada por

$$\dot{h}(t) = \sum_{i=1}^{n} \frac{\partial H}{\partial q_i}(\mathbf{q}, t)\, \dot{q}_i + \frac{\partial H}{\partial t}(\mathbf{q}, t)$$

e a sua derivada segunda por

$$\ddot{h}(t) = \sum_{i=1}^{n} \left[\left(\sum_{j=1}^{n} \frac{\partial^2 H}{\partial q_i \partial q_j}(\mathbf{q}, t)\, \dot{q}_j + 2\frac{\partial^2 H}{\partial q_i \partial t}(\mathbf{q}, t) \right) \dot{q}_i + \frac{\partial H}{\partial q_i}(\mathbf{q}, t)\, \ddot{q}_i \right] + \frac{\partial^2 H}{\partial t^2}(\mathbf{q}, t) \tag{3.1}$$

Portanto, no cálculo das acelerações $\ddot{\mathbf{q}}$ através das equações de movimento (2.1), temos que levar em conta a equação

$$\sum_{i=1}^{n} \frac{\partial H}{\partial q_i}(\mathbf{q}, t)\, \ddot{q}_i = -\sum_{i=1}^{n} \left(\sum_{j=1}^{n} \frac{\partial^2 H}{\partial q_i \partial q_j}(\mathbf{q}, t)\, \dot{q}_j + 2\frac{\partial^2 H}{\partial q_i \partial t}(\mathbf{q}, t) \right) \dot{q}_i - \frac{\partial^2 H}{\partial t^2}(\mathbf{q}, t) \tag{3.2}$$

Nas derivações acima, estamos supondo implicitamente que as funções H têm todas as derivadas necessárias. Um tratamento mais rigoroso fugiria ao escopo desta dissertação.

3.2.3 Estabilização de vínculos

A análise de vínculos holonômicos na seção (3.2.2) seria precisa se o estado inicial do sistema fosse válido (isto é, se $h(t) = 0$ e $\dot{h}(t) = 0$ para todo vínculo no instante inicial), e se a integração das equações de movimento fosse exata. Na prática, entretanto, não basta tentar garantir $\ddot{h}(t) = 0$. Se o vínculo está violado num instante t, isto é, $\dot{h}(t) \neq 0$ ou $h(t) \neq 0$, a condição $\ddot{h}(t) = 0$ vai na verdade preservar ou aumentar essa violação.

Podemos, contudo, obrigar essas violações a diminuir com o tempo, substituindo a equação diferencial $\ddot{h}(t) = 0$ pela equação

$$\ddot{h}(t) + 2\,\xi\, \dot{h}(t) + \xi^2\, h(t) = 0 \tag{3.3}$$

onde ξ é uma constante positiva.

Para justificar esta fórmula, imaginemos que $h(t)$ é o deslocamento de uma "mola" em $\mathbb{R}^n$ que liga a configuração corrente do sistema a uma configuração sobre a superfície do vínculo $H(\mathbf{q}, t) = 0$ (como ilustra a figura (3.2)). A equação diferencial (3.3) possui a solução

$$h(t) = (A + B\,(t - t_0))\, e^{-\xi(t-t_0)}$$

com $A = h(t_0)$ e $B = \dot{h}(t_0) + \xi\, h(t_0)$, que é única [BD94]. A constante ξ é a freqüência natural de oscilação da mola. O termo em $\dot{h}(t)$ acrescenta um "amortecedor" a essa mola, que leva o valor de h a cair o mais rapidamente possível em direção ao zero, sem jamais ultrapassá-lo, evitando assim que o sistema passe a oscilar indefinidamente em torno da superfície de vínculo.

Expandindo a equação (3.3), obtemos novas restrições sobre as acelerações $\ddot{q}_i$, em substituição às equações (3.2):

$$\begin{aligned}\sum_{i=1}^{n} \frac{\partial H}{\partial q_i}(\mathbf{q}, t)\, \ddot{q}_i \;=\; & -\sum_{i=1}^{n} \left(\sum_{j=1}^{n} \frac{\partial^2 H}{\partial q_i \partial q_j}(\mathbf{q}, t)\, \dot{q}_j + 2 \frac{\partial^2 H}{\partial q_i \partial t}(\mathbf{q}, t) + 2\,\xi\, \frac{\partial H}{\partial q_i}(\mathbf{q}, t) \right) \dot{q}_i \\ & -\xi^2\, H(\mathbf{q}, t) - 2\,\xi\, \frac{\partial H}{\partial t}(\mathbf{q}, t) - \frac{\partial^2 H}{\partial t^2}(\mathbf{q}, t)\end{aligned} \tag{3.4}$$

De maneira análoga, vínculos que dependem de $\dot{\mathbf{q}}$ mas não de $\ddot{\mathbf{q}}$ podem ser estabilizados obrigando-se o desvio h a satisfazer $\dot{h}(t) + \xi h(t) = 0$, cuja solução é uma exponencial decrescente $h(t) = h(t_0)\, e^{-\xi(t-t_0)}$.

3.3 A direção das forças de vínculo

A diferença entre a abordagem diferencial e a modelagem de vínculos por forças de penalidade (seção (3.2.1)), é que no primeiro caso a fórmula (3.4) impõe uma restrição direta às acelerações $\ddot{\mathbf{q}}$ de modo a satisfazer o vínculo; enquanto que, na outra solução, o desvio gera apenas mais uma força sobre o sistema, que pode não ser suficiente para manter o vínculo satisfeito.

Na abordagem exata, a magnitude da força de vínculo não é conhecida *a priori*. Ela também não pode ser calculada isoladamente, pois depende das forças nos outros vínculos do sistema. Entretanto, para muitos tipos de vínculos $H(\mathbf{q}, t) = 0$, a força associada tem direção conhecida $\mathbf{d}$. Nesse caso, a força de vínculo $\mathbf{f}^{\mathrm{v}}$ correspondente tem a forma $\lambda\, \mathbf{d}$, onde λ é um parâmetro escalar a ser determinado.

Nas duas abordagens, as acelerações $\ddot{\mathbf{q}}$ são calculadas a partir das forças totais. No método aproximado, entretanto, a força de penalidade é calculada apenas do estado $(\mathbf{q}, \dot{\mathbf{q}})$, enquanto que no método exato (com ou sem estabilização), a força de vínculo é calculada levando em conta as equações que restringem as acelerações $\ddot{\mathbf{q}}$, de modo que o valor resultante de $\ddot{\mathbf{q}}$, calculado a partir das forças, satisfaça exatamente essas equações.

No que segue, vamos denotar por k o número de vínculos do sistema. Os parâmetros $\lambda_1, \ldots, \lambda_k$, que dão a intensidade de cada força de vínculo $\mathbf{f}_i^{\mathrm{v}} = \lambda_i\, \mathbf{d}_i$, são conhecidos como *multiplicadores de Lagrange*.

Em princípio, a força de vínculo associada a uma equação $H(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}, t) = 0$ depende do processo físico envolvido e pode ter qualquer direção que não seja tangente à superfície desse vínculo. Muitas vezes, a direção $\mathbf{d}$ da força de vínculo $\mathbf{f}^{\mathrm{v}}$ pode ser deduzida considerando-se a taxa de trabalho realizado pela mesma, que é sempre $\mathbf{f}^{\mathrm{v}} \cdot \dot{\mathbf{q}} = \lambda(\mathbf{d} \cdot \dot{\mathbf{q}})$. Em particular, se o processo físico é conservativo (ou seja, não acrescenta nem remove energia útil do sistema), pode-se concluir que a direção $\mathbf{d}$ é ortogonal à velocidade $\dot{\mathbf{q}}$. Nas seções seguintes, vamos determinar a direção $\mathbf{d}$ para alguns dos casos mais comuns na prática.

3.3.1 Vínculos sem atrito

Um caso simples e muito comum é o de um vínculo holonômico $H(\mathbf{q}, t) = 0$ cujo processo físico é insensível a movimentos do sistema nas direções tangentes à superfície do vínculo. Neste caso, a direção $\mathbf{d}$ da força é normal àquela superfície. Isto é,

$$\mathbf{d} = \left[\frac{\partial H}{\partial q_1}(\mathbf{q}, t), \frac{\partial H}{\partial q_2}(\mathbf{q}, t), \ldots, \frac{\partial H}{\partial q_n}(\mathbf{q}, t)\right]$$

Note que, numa trajetória onde o vínculo é respeitado, devemos ter $\dot{h}(t) = 0$ para todo t. Uma vez que

$$\dot{h}(t) = \sum_{i=1}^{n} \frac{\partial H}{\partial q_i}(\mathbf{q}, t)\, \dot{q}_i + \frac{\partial H}{\partial t}(\mathbf{q}, t) = \mathbf{d} \cdot \dot{\mathbf{q}} + \frac{\partial H}{\partial t}(\mathbf{q}, t)$$

segue-se que

$$\mathbf{d} \cdot \dot{\mathbf{q}} = -\frac{\partial H}{\partial t}(\mathbf{q}, t)$$

Portanto, a potência W realizada pela força de vínculo num instante t é

$$W = -\lambda\, \frac{\partial H}{\partial t}(\mathbf{q}, t)$$

O exemplo clássico é o de uma partícula de posição $\mathbf{q} = [x, y, z]$ que desliza sem atrito numa superfície (possivelmente móvel) $H(\mathbf{q}, t) = 0$. A força de contato $\mathbf{f}$, que mantém a partícula sobre a superfície, é sempre perpendicular à mesma, isto é,

$$\mathbf{f} = \lambda \left(\frac{\partial H}{\partial x}(\mathbf{q}, t), \frac{\partial H}{\partial y}(\mathbf{q}, t), \frac{\partial H}{\partial z}(\mathbf{q}, t)\right)$$

Veja a figura (3.3).

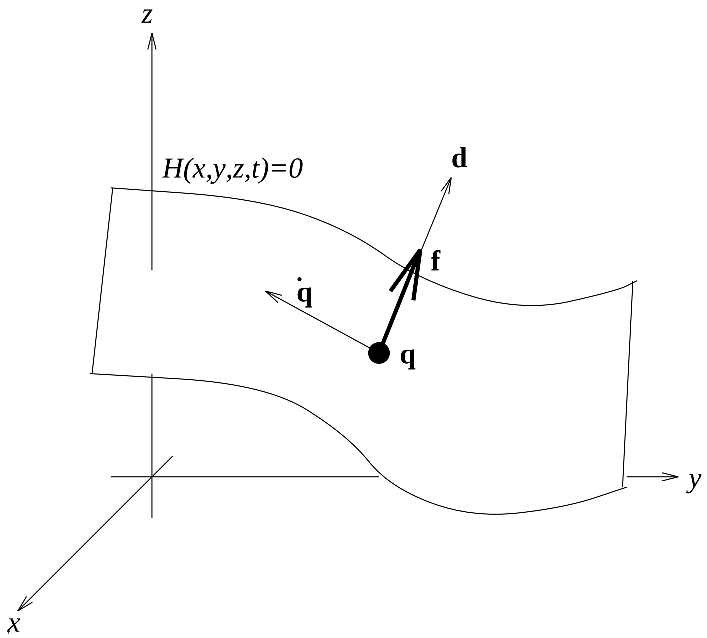


Figura 3.3: Força de vínculo associada à equação $H(\mathbf{q}, t) = 0$ no $\mathbb{R}^3$

3.3.2 Vínculos com atrito

Outro caso comum é o de vínculos com atrito, onde, além da força de vínculo normal à superfície do vínculo, existe uma componente tangencial que depende da velocidade $\dot{\mathbf{q}}$, e se opõe à mesma (de modo que o trabalho realizado pela força é negativo).

Por exemplo, considere uma partícula u de posição $\mathbf{p}$ que está restrita a uma superfície fixa S. Seja v a partícula de S que está em contato com u num dado instante. Se a superfície não for completamente lisa, uma força de atrito dinâmico $\mathbf{f}^{\mathrm{t}}$ que reduz a energia cinética do sistema irá surgir, na direção oposta à da velocidade relativa $\mathbf{v}$ entre as partículas u e v. No modelo de Coulomb para o atrito entre superfícies secas, a força de atrito dinâmico é

$$\mathbf{f}^{\mathrm{t}} = -\mu \, |\mathbf{f}^{\mathrm{n}}| \frac{\mathbf{v}}{|\mathbf{v}|}$$

onde μ é um coeficiente de atrito dinâmico relacionado aos dois pontos de contato, e $\mathbf{f}^{\mathrm{n}}$ é a força normal de contato, que é ortogonal a S no ponto correspondente a v.

Uma vez que a força tangencial depende da força normal e pode influenciar indiretamente esta última, as duas precisam ser calculadas simultaneamente.

3.3.3 Vínculos sobre as velocidades

Para um vínculo não holonômico $H(\mathbf{q}, \dot{\mathbf{q}}, t) = 0$ que depende de $\dot{\mathbf{q}}$ (mas não de $\ddot{\mathbf{q}}$), a força de vínculo $\mathbf{f}^{\mathrm{v}}$ pode ser definida como

$$f_i^{\mathrm{v}} = -\lambda \frac{\partial H}{\partial \dot{q}_i}(\mathbf{q}, \dot{\mathbf{q}}, t)$$

para $i = 1, \ldots, n$.

3.3.4 Formulação matricial

As equações de movimento com vínculos podem ser reescritas em notação compacta, definindo-se as matrizes $k \times n$ $\mathbf{J}$ e $\mathbf{D}$ e o vetor $\boldsymbol{\psi}$ como

$$\begin{aligned}
J_{lj} &= \frac{\partial H_l}{\partial q_j}(\mathbf{q}, t) \\
\psi_l &= -\sum_{i=1}^{n}\left(\sum_{j=1}^{n} \frac{\partial^2 H_l}{\partial q_i \partial q_j}(\mathbf{q}, t)\, \dot{q}_j + 2 \sum_{i=1} \frac{\partial^2 H_l}{\partial q_i \partial t}(\mathbf{q}, t) + 2\,\xi\, \frac{\partial H}{\partial q_i}(\mathbf{q}, t)\right) \dot{q}_i \\
&\quad -\xi^2 H_l(\mathbf{q}, t) - 2\,\xi\, \frac{\partial H_l}{\partial t}(\mathbf{q}, t) - \frac{\partial^2 H_l}{\partial t^2}(\mathbf{q}, t) \\
D_{lj} &= -(\mathbf{d}_l)_j
\end{aligned}$$

para $i, j = 1, \ldots, n$ e $l = 1, \ldots, k$. As equações de movimento (2.1) e (3.4) equivalem então ao seguinte sistema de equações em $n + k$ incógnitas:

$$\begin{bmatrix} \mathbf{M} & \mathbf{D}^{\mathsf{T}} \\ \mathbf{J} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \boldsymbol{\psi} \end{bmatrix} \tag{3.5}$$

onde $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_k]$ é o vetor dos multiplicadores de Lagrange.

Capítulo 4

Física do material

"A mecânica dos chamados meios contínuos se caracteriza pela ficção de que a densidade e a velocidade da matéria dependem, de maneira contínua, das coordenadas e do tempo, e que a parte das interações não explicitamente dadas pode ser considerada como forças de superfície (forças de pressão) que mais uma vez são funções contínuas da localização."

Albert Einstein

4.1 Modelo contínuo

Num corpo sólido, as partículas vizinhas do material normalmente permanecem próximas quando o corpo se move. Se a distância entre as partículas varia com o tempo, a ponto de mudar macroscopicamente a forma ou o tamanho do corpo, dizemos que o material é *deformável*.

Seja $\tilde{U}$ o conjunto (finito) de partículas de um sistema dinâmico. No caso de corpos sólidos, onde o número de partículas é astronômico, convém aproximar o conjunto $\tilde{U}$ por uma região contínua U do $\mathbb{R}^3$ (como se tivéssemos preenchido os espaços vazios entre as partículas com uma infinidade de novas partículas). Note que esta aproximação não seria aplicável a um meio gasoso ou líquido, onde as relações de vizinhança entre as partículas podem variar consideravelmente ao longo do tempo.

Neste trabalho, vamos supor que U é compacto e tridimensional. (Um conjunto $U \subset \mathbb{R}^3$ é *tridimensional* se toda vizinhança de todo ponto $u \in U$ tem um ponto interior, isto é, um ponto cuja vizinhança é uma bola aberta; e *compacto* se ele é topologicamente fechado e tem diâmetro finito.) Note que o conjunto U pode ser desconexo; nesse caso, cada componente conexa corresponde a um objeto isolado do sistema.

4.1.1 Configuração e estado

No capítulo 2, definimos a configuração do sistema como sendo uma função injetora $\tilde{\mathbf{f}} : \tilde{U} \to \mathbb{R}^3$ que dá a posição no espaço $\tilde{\mathbf{f}}(u) = [\tilde{f}_x(u), \tilde{f}_y(u), \tilde{f}_z(u)]$ de cada partícula $u = [u_x, u_y, u_z] \in \tilde{U}$.

No modelo contínuo, por sua vez, a configuração do sistema é definida como uma função $\mathbf{f} : U \to \mathbb{R}^3$, que dá a posição de cada partícula $u \in U$. Vamos exigir que $\mathbf{f}$ seja contínua em U, e injetora no interior U^+ de U. Esta última condição (conhecida como a propriedade de *impenetrabilidade* da matéria) significa que duas partículas em U^+ nunca assumem a mesma posição no espaço ao mesmo tempo. Note que, ao excluir os pontos da fronteira de U, nesta condição, estamos permitindo que dois pontos distintos na superfície dos objetos se toquem. Observe ainda que cada bola aberta em U é mapeada por uma configuração $\mathbf{f}$ para uma bola aberta em $\mathbb{R}^3$, e portanto a imagem $\mathbf{f}(U)$ de $\mathbf{f}$ é também um conjunto compacto e tridimensional.

A *configuração de referência* do sistema é a função identidade sobre U. As coordenadas $[u_x, u_y, u_z]$ de uma partícula em $u \in U$ são ditas as *coordenadas de referência* de u.

Analogamente ao caso discreto, redefinimos o *estado* do sistema como um par de funções $\mathbf{f}$ e $\dot{\mathbf{f}}$ de U para $\mathbb{R}^3$, tais que para toda partícula $u \in U$, $\mathbf{f}(u) = [f_x(u), f_y(u), f_z(u)]$ é a posição de u e $\dot{\mathbf{f}}(u) = [\dot{f}_x(u), \dot{f}_y(u), \dot{f}_z(u)]$ é a velocidade de u.

Definimos também a *trajetória* do sistema como uma função $\mathbf{f} : U \times \mathbb{R} \to \mathbb{R}^3$ tal que para todo $u \in U$, $\mathbf{f}(u,t) = [f_x(u,t), f_y(u,t), f_z(u,t)]$ é a posição da partícula u no instante t.

4.1.2 Densidade de massa

No modelo discreto, cada partícula tem massa definida e invariável. Quando passamos para o modelo contínuo, precisamos substituir o conceito de massa de cada partícula pelo conceito de *densidade de massa* em cada ponto no interior de U.

Seja u um ponto de U^+ e R um subconjunto tridimensional de U^+ que contém u. Por definição, a densidade de massa $\rho(u)$ no ponto u é o limite da massa contida em R dividida pelo volume de R, quando o diâmetro de R tende a zero. Vamos supor que esse limite existe para *quase* todo ponto $u \in U^+$; ou seja, que o conjunto de pontos onde ele não existe tem volume zero. Mais ainda, vamos supor que a parte de U onde ρ não está definida tem massa total nula. Com essas suposições, podemos dizer que a massa m de uma região $R \subseteq U$ é dada por $m = \int_R \rho(u)\, du$.

4.1.3 Energia cinética

A energia cinética K do sistema para um estado $(\mathbf{f}, \dot{\mathbf{f}})$ é dada por

$$K = \int_U \frac{\rho(u)}{2} \dot{\mathbf{f}}(u)^2 \, du \qquad (4.1)$$

Observe que esta fórmula implica $K \geq 0$, e que $K = 0$ somente quando $\dot{\mathbf{f}}(u) = 0$ para todo $u \in U$.

4.1.4 Energia elástica

Em geral, todo fragmento infinitesimal de material deformável possui uma forma natural de repouso, na qual ele permanece indefinidamente na ausência de forças externas. Para levar o fragmento a uma forma que não é de repouso, é necessário aplicar forças às partículas, realizando trabalho positivo, ou seja, aumentando a energia do sistema. Nos materiais ditos *elásticos*, se a deformação for feita suficientemente devagar, a energia investida nessa transformação pode ser recuperada mais tarde na forma de energia mecânica. As forças internas que surgem como reação à deformação são ditas *forças de elasticidade*, e a energia acumulada é a *energia elástica* (ou *energia potencial de deformação*).

4.1.5 Forças de viscosidade

Quando a velocidade de deformação do fragmento não é desprezível, o movimento relativo entre as partículas dá origem a forças internas de atrito ou *forças de viscosidade*, que se opõem a esse movimento. O trabalho realizado por essas forças é convertido em calor, e portanto pode ser considerado perdido do ponto de vista da simulação mecânica.

No restante deste capítulo veremos como calcular as forças de elasticidade e viscosidade, a partir das propriedades do material e das variações relativas de posição e velocidade entre partículas vizinhas em um meio contínuo.

4.2 Cálculo das forças de elasticidade

4.2.1 Deformação local

Vamos supor que a função de configuração $\mathbf{f}$ é analítica (em particular, infinitamente diferenciável com derivadas parciais finitas) em quase todo ponto de U^+ (ou seja, o conjunto dos pontos de U^+ em que $\mathbf{f}$ não é analítica tem volume zero). Indicaremos o subconjunto de U^+ no qual $\mathbf{f}$ é diferenciável por $\bar{U}$ — que é também o conjunto onde a densidade ρ é definida.

Esta suposição não é correta na escala atômica, pois num corpo sólido real, cada átomo oscila rápida e incessantemente em torno de uma posição de equilíbrio. Entretanto, como a amplitude dessa oscilação é desprezível do ponto de vista mecânico, podemos identificar a posição de cada partícula com sua posição de equilíbrio.

Se $\mathbf{f}$ é analítica no ponto $u \in U^+$, a posição na configuração $\mathbf{f}$ de uma partícula $u + \delta$, para um vetor $\delta = [\delta_x, \delta_y, \delta_z] \in \mathbb{R}^3$, é dada por

$$\mathbf{f}(u + \delta) = \mathbf{f}(u) + \mathbb{J}_\mathbf{f}(u)\,\delta + O(|\delta|^2)$$

onde $\mathbb{J}_\mathbf{f}$ é a *matriz jacobiana* de $\mathbf{f}$, definida em todo ponto $u \in \bar{U}$ como

$$\mathbb{J}_\mathbf{f}(u) = \begin{bmatrix} \frac{\partial f_x}{\partial u_x}(u) & \frac{\partial f_x}{\partial u_y}(u) & \frac{\partial f_x}{\partial u_z}(u) \\ \frac{\partial f_y}{\partial u_x}(u) & \frac{\partial f_y}{\partial u_y}(u) & \frac{\partial f_y}{\partial u_z}(u) \\ \frac{\partial f_z}{\partial u_x}(u) & \frac{\partial f_z}{\partial u_y}(u) & \frac{\partial f_z}{\partial u_z}(u) \end{bmatrix}$$

Dizemos que a matriz $\mathbb{J}_\mathbf{f}(u)$ descreve em primeira ordem a configuração do material nas vizinhanças da partícula u.

Pode-se demonstrar que, se R é uma região tridimensional de U, então o determinante $|\mathbb{J}_\mathbf{f}(u)|$, para um ponto u no interior de R, é o limite da razão entre o volume de $\mathbf{f}(R)$ e o volume de R, quando o diâmetro de R tende a zero. Como materiais reais não podem ser comprimidos para volume zero, o determinante $|\mathbb{J}_\mathbf{f}(u)|$ é sempre diferente de zero, ou seja, a matriz $\mathbb{J}_\mathbf{f}$ nunca é singular.

4.2.2 Configuração de repouso

Uma *configuração de repouso para um ponto $u \in U^+$* é uma configuração $\mathbf{q}_u : U \to \mathbb{R}^3$ na qual existe um fragmento tridimensional do material que contém u e está em repouso.

Note que a configuração de repouso depende em geral do ponto u considerado, pois um corpo macroscópico pode não possuir uma única configuração que seja de repouso para todos os pontos de U. Por exemplo, considere um corpo elástico de forma cilíndrica, que inicialmente está em repouso. Se esse corpo for dobrado em forma de anel e suas extremidades forem "coladas", a configuração resultante armazenará uma certa quantidade de energia elástica, embora o corpo seja incapaz de retornar à forma cilíndrica.

No restante deste capítulo, $\mathbf{q}_u$ denota uma configuração do sistema que é de repouso para uma dada partícula $u \in U^+$, enquanto que $\mathbf{p}$ denota a configuração do sistema em um dado instante, dita a *configuração corrente*.

4.2.3 Densidade de energia elástica

Seja R um subconjunto tridimensional de U^+ que contém a partícula $u \in U^+$. Por definição, a densidade de energia elástica $\phi(u)$ no ponto u é o limite da quantidade de energia elástica contida em R dividida pelo volume de repouso de R, quando o diâmetro de R tende a zero. Vamos supor que esse limite existe para quase todo ponto $u \in U^+$, e que a energia elástica armazenada no subconjunto de U onde ϕ não está definida é nula.

Portanto, a energia elástica e_R armazenada numa região $R \subseteq U$, para uma dada configuração $\mathbf{f}$, é

$$e_R = \int_R \phi(u)\, |\mathbb{J}_{\mathbf{q}_u}(u)|\, du$$

Note o fator $|\mathbb{J}_{\mathbf{q}_u}(u)|$, necessário para converter o volume de repouso (usado na definição de ϕ) para o volume na configuração de referência (usado na integração). Para todo $u \in \bar{U}$, este determinante existe e é único, independentemente de quantas configurações de repouso $\mathbf{q}_u$ diferentes existam para a partícula u.

Comprova-se experimentalmente que, para a grande maioria dos materiais, a densidade de energia ϕ no ponto u depende apenas das diferenças de primeira ordem entre as configurações $\mathbf{q}_u$ e $\mathbf{p}$ nas vizinhanças de u. Formalmente, $\phi(u)$ depende apenas da matriz jacobiana $\mathbf{C}(u)$ da composição $(\mathbf{p} \circ \mathbf{q}_u^{-1})(u)$, que mapeia o material na vizinhança de u da configuração de repouso $\mathbf{q}_u$ para a configuração corrente $\mathbf{p}$. As derivadas parciais de ordem mais alta das funções $\mathbf{p}$ e $\mathbf{q}_u$ não têm influência perceptível sobre ϕ. A matriz $\mathbf{C}$, dita o *tensor de deformação* (de $\mathbf{p}$ em relação a $\mathbf{q}_u$), é dada por

$$\mathbf{C}(u) = \mathbb{J}_{\mathbf{p}}(u)\, \mathbb{J}_{\mathbf{q}_u}^{-1}(u)$$

para todo ponto $u \in U^+$ no qual $\mathbf{p}$ e $\mathbf{q}_u$ são analíticas. (Note que a inversa $\mathbf{q}_u^{-1}$ existe em U^+ pela hipótese da impenetrabilidade.)

4.2.4 Invariância rotacional

Note que o tensor $\mathbf{C}$, assim como a densidade de energia elástica ϕ, não é afetado por uma translação do material na configuração corrente $\mathbf{p}$. Entretanto, o tensor $\mathbf{C}$ é sensível a rotações rígidas do material, que não alteram as posições relativas entre as partículas, e que portanto não deveriam afetar a densidade ϕ.

Para eliminar o efeito de rotações rígidas, podemos usar o teorema da *decomposição dos valores singulares* [GL85]. Ele diz que qualquer matriz $\mathbf{C} \in \mathbb{R}^{3\times 3}$ pode ser decomposta num produto da forma

$$\mathbf{C} = \mathbf{RDS} \tag{4.2}$$

onde $\mathbf{R}$ e $\mathbf{S}$ são matrizes de rotação e $\mathbf{D}$ é uma matriz diagonal única, a menos de permutação dos elementos da diagonal.

Em particular, o tensor de deformação $\mathbf{C}(u) = \mathbf{R}(u)\mathbf{D}(u)\mathbf{S}(u)$ equivale a uma rotação inicial $\mathbf{S}(u)$ do material em torno da partícula u, seguida de uma compressão ou expansão $\mathbf{D}(u)$ ao longo dos eixos de coordenadas, e uma rotação final $\mathbf{R}(u)$.

Portanto, a densidade de energia elástica ϕ, para materiais elásticos em geral, deve depender do produto $\mathbf{DS}$, mas não do fator $\mathbf{R}$, que nada mais faz do que orientar o material de acordo com a configuração corrente $\mathbf{p}$. Além disso, ϕ deve ter um valor mínimo (zero) quando $\mathbf{D}$ for a identidade.

Uma maneira prática de eliminar o efeito da rotação rígida $\mathbf{R}$, que não requer o cálculo direto da decomposição (4.2), é calcular o *tensor métrico*

$$\mathbf{T} = \mathbf{C}^\top \mathbf{C} = \mathbf{S}^\top \mathbf{DDS}$$

que é uma matriz simétrica. O nome desta matriz vem da seguinte propriedade: se $\delta = \mathbf{q}_u(u) - \mathbf{q}_u(v)$ é o deslocamento entre a partícula u e uma partícula v, na configuração de repouso $\mathbf{q}_u$, então o deslocamento $\epsilon = \mathbf{p}(u) - \mathbf{p}(v)$ entre elas na configuração corrente $\mathbf{p}$ é

$$\epsilon = \mathbf{C}(u)\delta + O(|\delta|^2)$$

e o quadrado da distância entre as mesmas é

$$|\epsilon|^2 = \delta^\top \mathbf{C}^\top(u)\mathbf{C}(u)\delta + O(|\delta|^3) = \delta^\top \mathbf{T}(u)\delta + O(|\delta|^3)$$

Note que, como era de esperar, o tensor $\mathbf{T}$ é afetado por qualquer mudança no vetor δ, mas permanece constante para mudanças no vetor ϵ que não alteram a distância $|\epsilon|$.

4.2.5 Materiais isotrópicos

Neste trabalho, vamos modelar apenas materiais elásticos *isotrópicos*. cujas propriedades mecânicas são idênticas em todas as direções. Em outras palavras, a energia elástica devida a uma deformação não depende da orientação do material em relação à direção em que a deformação ocorre. Por exemplo, um pedaço de borracha comum pode ser deformado tão facilmente numa direção do material quanto em outra. Em contraste, um pedaço de madeira geralmente exibe um comportamento não isotrópico, sendo mais facilmente esticado na direção transversal às suas fibras do que na direção das mesmas.

Num material isotrópico, a densidade de energia elástica ϕ não deve ser afetada por uma rotação do material em torno da partícula u na configuração de repouso $\mathbf{q}_u$, ou seja, $\phi(u)$ deve ser independente da matriz de rotação $\mathbf{S}(u)$, que orienta o material na vizinhança de u de acordo com a configuração de repouso $\mathbf{q}_u$.

Portanto, ϕ deve ser expressa como uma função $\Phi(\sigma_1, \sigma_2, \sigma_3)$ dos elementos diagonais da matriz $\mathbf{D}$ (que são os valores singulares do tensor $\mathbf{C}$). Mais do que isso, Φ deve ser *simétrica*, isto é, independente da ordem das variáveis $\sigma_1, \sigma_2, \sigma_3$.

Porém, não vamos definir Φ diretamente a partir dos valores singulares $\sigma_1, \sigma_2, \sigma_3$, pois não queremos realizar aquela decomposição explicitamente. Vamos, em vez disso, nos utilizar do fato de que toda função simétrica em três variáveis pode ser reescrita como outra função, em três combinações simétricas independentes daquelas três variáveis.

4.2.6 O polinômio característico

O polinômio

$$\chi(\lambda) = |\mathbf{T} - \lambda \mathbf{I}|$$

onde $\mathbf{I}$ é a matriz identidade de dimensão 3×3 e λ é um escalar a ser determinado, é dito o *polinômio característico* da matriz $\mathbf{T}$. Este polinômio pode ser escrito na forma

$$\chi(\lambda) = \lambda^3 - \Gamma\lambda^2 + \Sigma\lambda - \Delta$$

onde Γ, Σ e Δ são combinações simétricas independentes das raízes $\lambda_1, \lambda_2, \lambda_3$ da equação $\chi(\lambda) = 0$. Verifica-se facilmente que

$$\begin{aligned}
\Delta &= \lambda_1\lambda_2\lambda_3 \\
\Gamma &= \lambda_1 + \lambda_2 + \lambda_3 \\
\Sigma &= \lambda_1\lambda_2 + \lambda_1\lambda_3 + \lambda_2\lambda_3
\end{aligned}$$

As raízes $\lambda_1, \lambda_2, \lambda_3$ são denominadas os *auto-valores* de $\mathbf{T}$. É fácil verificar também que os auto-valores de $\mathbf{T} = \mathbf{C}^\top\mathbf{C}$ são iguais aos quadrados dos valores singulares $\sigma_1, \sigma_2, \sigma_3$ da matriz $\mathbf{C}$.

Portanto, a função Φ, para um material isotrópico, pode ser definida a partir dos coeficientes Δ, Γ e Σ, que podem ser calculados a partir de $\mathbf{T}$ como

$$\begin{aligned}
\Delta &= |\mathbf{T}| \\
\Gamma &= T_{11} + T_{22} + T_{33} \\
\Sigma &= T^{(2)}_{11} + T^{(2)}_{22} + T^{(2)}_{33}
\end{aligned} \tag{4.3}$$

onde $\mathbf{T}^{(2)}$ denota a matriz de cofatores de $\mathbf{T}$. Pode-se também calcular estes coeficientes a partir do tensor $\mathbf{C}$, pelas fórmulas

$$\begin{aligned}
\Delta &= |\mathbf{C}|^2 \\
\Gamma &= \textstyle\sum_{ij}(C_{ij})^2 \\
\Sigma &= \textstyle\sum_{ij}(C^{(2)}_{ij})^2
\end{aligned}$$

onde $C^{(2)}_{ij}$ são os cofatores de $\mathbf{C}$.

4.2.7 Análise para pequenas deformações

Antes de escolhermos uma expressão para a função Φ que seja adequada a deformações grandes, vamos estudar o caso mais simples de deformações pequenas, para materiais isotrópicos.

Numa deformação pequena, o valor singular σ_i é próximo de 1, para $i = 1, 2, 3$. Para esta análise, vamos definir a função $\Phi^*(\epsilon_1, \epsilon_2, \epsilon_3) = \Phi(1 + \epsilon_1, 1 + \epsilon_2, 1 + \epsilon_3)$.

Note que a função Φ^*, contrariamente à função Φ, não possui termos constantes nem lineares, pois ela tem um mínimo absoluto, zero, para $\epsilon_1 = \epsilon_2 = \epsilon_3 = 0$. Portanto, a expansão em série de Taylor da função Φ^*, quando $\epsilon_1, \epsilon_2, \epsilon_3$ são próximos de zero, é dominada pelos termos de segundo grau. Ou seja, para deformações pequenas a função homogênea e simétrica Φ^* é aproximadamente quadrática em $\epsilon_1, \epsilon_2, \epsilon_3$.

Demonstra-se que o conjunto $\mathbb{H}_3^2$ das funções quadráticas homogêneas e simétricas de três variáveis tem dimensão dois. Ou seja, podemos escrever

$$\Phi^*(\epsilon_1, \epsilon_2, \epsilon_3) \approx \alpha(u)\,\mathcal{F}^*(\epsilon_1, \epsilon_2, \epsilon_3) + \beta(u)\,\mathcal{G}^*(\epsilon_1, \epsilon_2, \epsilon_3) \tag{4.4}$$

onde $\mathcal{F}^*$ e $\mathcal{G}^*$ são quaisquer duas funções linearmente independentes de $\mathbb{H}_3^2$, e $\alpha(u), \beta(u)$ são coeficientes reais não negativos, que dependem das propriedades elásticas do material nas vizinhanças da partícula u.

4.2.8 Módulos de elasticidade

Do ponto de vista físico, é conveniente tomar como base de $\mathbb{H}_3^2$ as funções

$$\begin{aligned}
\mathcal{F}^* &= \frac{(\epsilon_1 + \epsilon_2 + \epsilon_3)^2}{2} \\
\mathcal{G}^* &= \frac{2}{3}\frac{\epsilon_1^2 + \epsilon_2^2 + \epsilon_3^2 - \epsilon_1\epsilon_2 - \epsilon_1\epsilon_3 - \epsilon_2\epsilon_3}{2}
\end{aligned}$$

Os coeficientes $\alpha(u)$ e $\beta(u)$ na fórmula (4.4) são os *módulos de elasticidade* do material no ponto u. Suas unidades de medida são força por unidade de área (*tensão*), ou de modo equivalente, energia por unidade de volume (*densidade de energia*).

Para compreender o significado físico destes coeficientes, vamos considerar as seguintes deformações particulares. Seja $R \subset U$ um fragmento pequeno do material tal que as propriedades elásticas no seu interior são aproximadamente uniformes, e $\mathbf{q}$ uma configuração de repouso para R. Se aplicamos uma pressão uniforme p sobre a superfície de R, o fragmento sofre uma contração uniforme (sem variação de forma), do volume V para o volume $V - \delta V$. Veja a figura (4.1).

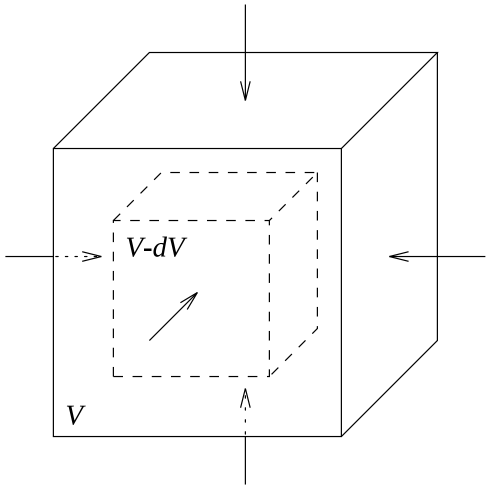


Figura 4.1: Compressão uniforme de um fragmento cúbico do material

Para deformações pequenas, a variação relativa de volume $\delta V/V$ é aproximadamente proporcional à pressão p. Nesse caso, demonstra-se que o trabalho realizado nessa deformação é aproximadamente $1/2p\,\delta V$. Logo, a densidade de energia dentro do fragmento R comprimido é

$$\phi \approx \frac{1}{2} p \frac{\delta V}{V} \tag{4.5}$$

Por outro lado, o tensor de deformação correspondente a esta contração é

$$\mathbf{C} = \begin{bmatrix} 1+\epsilon & 0 & 0 \\ 0 & 1+\epsilon & 0 \\ 0 & 0 & 1+\epsilon \end{bmatrix}$$

onde $1+\epsilon = (1-\delta V/V)^{1/3}$, e portanto

$$\epsilon = -\frac{1}{3}\frac{\delta V}{V} + O\left((\delta V/V)^2\right)$$

Para esta deformação temos

$$\mathcal{F}^* = \frac{(3\epsilon)^2}{2} = \frac{1}{2}\left(\frac{\delta V}{V}\right)^2 + O\left((\delta V/V)^3\right) \tag{4.6}$$

$$\mathcal{G}^* = \frac{2}{3}\frac{3\epsilon^2 - 3\epsilon^2}{2} = 0 \tag{4.7}$$

Portanto, a fórmula (4.4) reduz-se a

$$\Phi^*(u) \approx \frac{\alpha(u)}{2}\left(\frac{\delta V}{V}\right)^2 \tag{4.8}$$

Comparando (4.5) com (4.8), concluimos que o coeficiente $\alpha(u)$, conhecido como *módulo volumétrico*, é

$$\alpha(u) = \lim_{\delta V \to 0} \frac{p}{(\delta V/V)}$$

A equação (4.7) mostra que o termo $\mathcal{G}^*$ é praticamente nulo para pequenas expansões ou contrações uniformes, que mudam o volume sem afetar a forma.

Suponha agora que o fragmento R tem forma cúbica na configuração de repouso $\mathbf{q}$, com altura h e faces de área A. Se aplicarmos forças paralelas e opostas $\mathbf{f}$ e $-\mathbf{f}$ sobre duas faces opostas desse fragmento, elas sofrem um deslocamento relativo s — veja a figura (4.2).

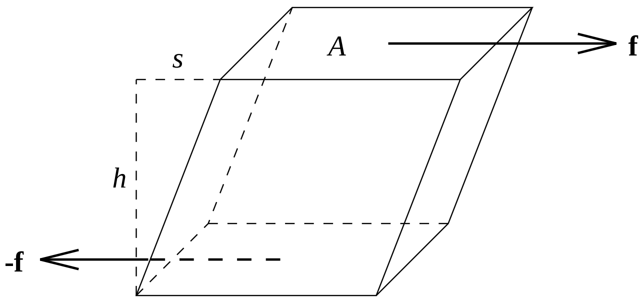


Figura 4.2: Cisalhamento de um fragmento cúbico do material

Novamente, a deformação é proporcional à força. Portanto, o trabalho realizado é aproximadamente $|\mathbf{f}|\, s/2$, do que concluimos que a densidade de energia no fragmento deformado é

$$\phi \approx \frac{|\mathbf{f}|\, s}{2A\, h} \tag{4.9}$$

Esta deformação (conhecida como *cisalhamento*) é descrita pelo tensor de deformação

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & s/h \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Verifica-se que os valores singulares desta matriz são $1+\epsilon_1, 1+\epsilon_2, 1+\epsilon_3$, onde

$$\begin{aligned} \epsilon_1 &= 0 \\ \epsilon_2 &= -\frac{s}{2h} + O\left((s/h)^2\right) \\ \epsilon_3 &= \frac{s}{2h} + O\left((s/h)^2\right) \end{aligned}$$

Para esta deformação, temos

$$\mathcal{F}^* = 0 \tag{4.10}$$

$$\mathcal{G}^* = \frac{1}{2}\left(\frac{s}{h}\right)^2 + O\left((s/h)^3\right) \tag{4.11}$$

Portanto, a fórmula (4.4) reduz-se a

$$\Phi^*(u) \approx \frac{\beta(u)}{2} \left(\frac{s}{h}\right)^2 \tag{4.12}$$

Comparando (4.9) com (4.12), concluimos que, para pequenos cisalhamentos, o coeficiente $\beta(u)$, conhecido como *módulo de rigidez*, é

$$\beta(u) = \lim_{s \to 0} \frac{|\mathbf{f}|\, h}{A\, s}$$

As equações (4.10) e (4.11) mostram que o termo $\mathcal{F}^*$, contrariamente a $\mathcal{G}^*$, é praticamente insensível a pequenos cisalhamentos do material, que mudam a forma sem afetar o volume.

Finalmente, vale notar que um terceiro tipo de deformação, um alongamento numa direção e uma contração numa direção ortogonal (que muda a forma sem alterar o volume) pode ser também obtido através de uma rotação em torno de u na configuração de repouso $\mathbf{q}_u$ seguida de um cisalhamento e uma rotação na configuração corrente $\mathbf{p}$. Portanto, qualquer deformação pequena do material pode ser decomposta nessas três deformações, básicas dentre as quais as de contração/expansão uniforme e cisalhamento são independentes.

4.2.9 Densidade de energia para grandes deformações

A fórmula (4.4) da densidade de energia para pequenas deformações permite que forças finitas comprimam o volume de um fragmento do material para zero ou negativo. Para evitar que isso ocorra em nosso modelo, vamos exigir que a função Φ, que mede a densidade de energia ϕ para deformações arbitrárias, tenda a infinito quando o volume do fragmento tende a zero.

Por simplicidade, decidimos formular a função Φ de maneira semelhante à função Φ^* para deformações pequenas, isto é, como uma soma de dois termos:

$$\Phi(\sigma_1, \sigma_2, \sigma_3) = \alpha(u)\, \mathcal{F}(\sigma_1, \sigma_2, \sigma_3) + \beta(u)\, \mathcal{G}(\sigma_1, \sigma_2, \sigma_3))$$

onde $\sigma_1, \sigma_2, \sigma_3$ são os valores singulares de $\mathbf{C}(u)$; $\mathcal{F}, \mathcal{G}$ são funções não negativas que medem respectivamente a mudança de volume e a mudança de forma do material nas vizinhanças de cada partícula u, em relação a uma configuração de repouso $\mathbf{q}_u$; e α, β são os módulos de elasticidade para pequenas deformações, descritos na seção (4.2.8).

Especificamente, escolhemos as funções

$$\begin{aligned} \mathcal{F} &= a\left(\Delta^2 + \frac{1}{\Delta^2} - 2\right) \\ \mathcal{G} &= b\left(\Gamma^2 - 3\Sigma\right) \end{aligned}$$

onde a e b são constantes. Note que $\mathcal{F}$ depende apenas da mudança de volume, para deformações de qualquer magnitude, enquanto que $\mathcal{G}$ depende apenas da mudança de forma, permanecendo constante quando somente o volume varia.

Para um fragmento do material que está em repouso, temos $\mathbf{C} = \mathbf{I}$, e portanto $\Delta = 1$, $\Gamma = 3$, $\Sigma = 3$ e $\Phi = 0$, como era esperado.

Comparando as funções $\mathcal{F}$ e $\mathcal{G}$ com as funções $\mathcal{F}^*$ e $\mathcal{G}^*$, pode-se verificar que para pequenas deformações

$$\begin{aligned} \mathcal{F} &\approx 16\left(\frac{\delta V}{V}\right)^2 \\ \mathcal{G} &\approx 3\left(\frac{s}{h}\right)^2 \end{aligned}$$

Portanto, precisamos escolher $a = 1/32$ e $b = 1/6$, para que Φ seja equivalente a Φ^*, quando as deformações forem suficientemente pequenas.

Resumindo, no nosso modelo a densidade de energia elástica ϕ é calculada através da função

$$\Phi(\Delta, \Gamma, \Sigma) = \frac{\alpha(u)}{32}\left(\Delta^2 + \frac{1}{\Delta^2} - 2\right) + \frac{\beta(u)}{6}\left(\Gamma^2 - 3\Sigma\right) \tag{4.13}$$

onde Δ, Γ e Σ são calculados a partir do tensor métrico $\mathbf{T}(u)$, pelas fórmulas (4.3).

4.2.10 Exemplos

Para esclarecer o comportamento de Φ em deformações grandes, vamos considerar três deformações diferentes num material com $\alpha = \beta = 1$.

O tensor de deformação

$$\mathbf{C}'(s) = \begin{bmatrix} s^{1/3} & 0 & 0 \\ 0 & s^{1/3} & 0 \\ 0 & 0 & s^{1/3} \end{bmatrix}$$

descreve uma mudança de volume por um fator s, sem mudança de forma. A figura (4.3) mostra a densidade de energia elástica Φ (que, no caso, contém apenas o termo $\mathcal{F}$) em função do parâmetro s:

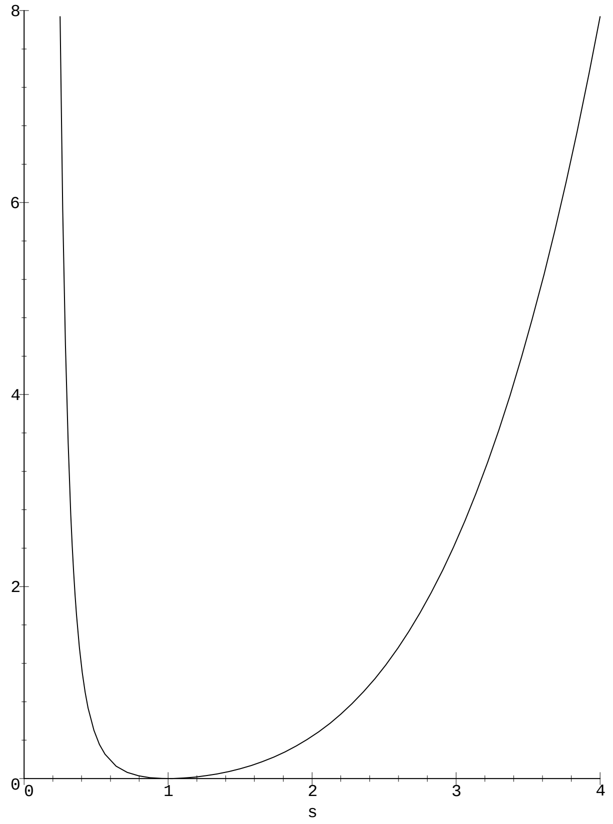


Figura 4.3: A densidade de energia Φ para deformação de expansão ou contração uniforme $\mathbf{C}'(s)$

A figura (4.3) evidencia que o primeiro termo de Φ é simétrico em relação a expansões e compressões do volume de repouso por um mesmo fator. Por exemplo, este termo dá o mesmo valor se o volume de repouso for dobrado ou reduzido à metade.

O tensor

$$\mathbf{C}''(s) = \begin{bmatrix} s & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1/s \end{bmatrix}$$

descreve uma mudança de forma que deixa o volume inalterado, ou mais especificamente, um alongamento na direção x por um fator s, e uma contração na direção z por um fator

$1/s$. A figura (4.4) mostra a densidade de energia Φ (que neste caso se reduz ao termo $\mathcal{G}$) em função de s:

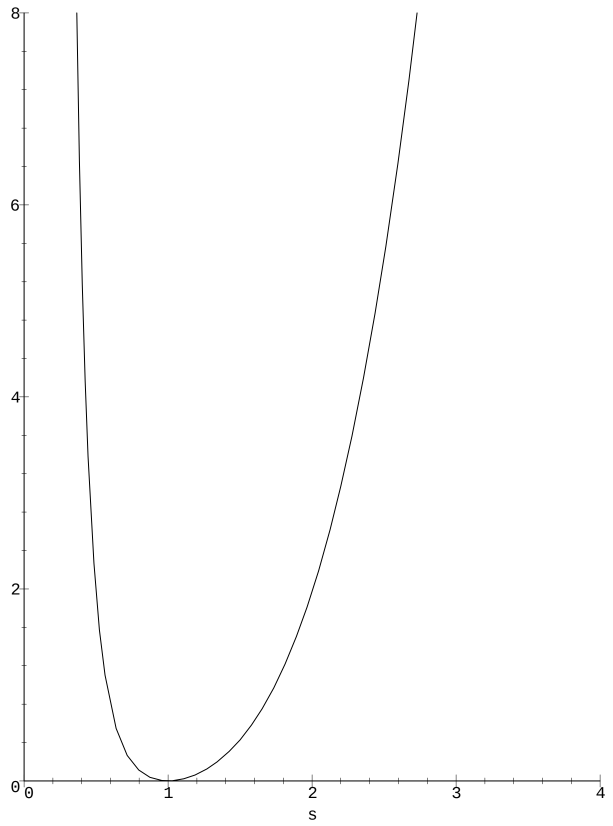


Figura 4.4: A densidade de energia Φ para deformação a volume constante $\mathbf{C}''(s)$

Outra mudança de forma a volume constante, especificamente um cisalhamento no plano xOy por um ângulo $\arctan s$, é descrita pelo tensor

$$\mathbf{C}'''(s) = \begin{bmatrix} 1 & 0 & 0 \\ s & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

A figura (4.5) mostra a densidade Φ (que neste caso também se reduz ao termo $\mathcal{G}$) em função do parâmetro s:

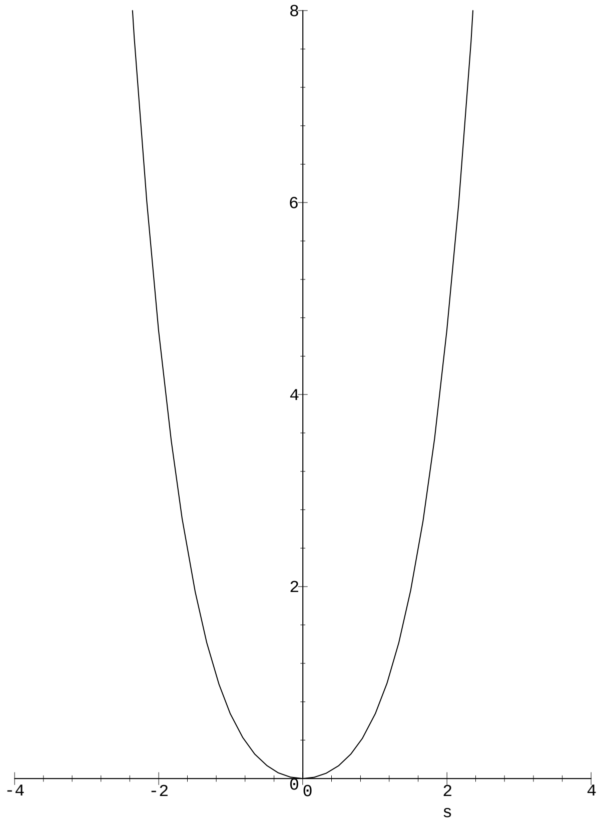


Figura 4.5: A densidade de energia Φ para cisalhamento $\mathbf{C}'''(s)$

Note que o segundo termo de Φ também exibe simetria. Por exemplo, se um fragmento do material na configuração de repouso aumenta de 2 numa direção e reduz de $1/2$ numa direção ortogonal, a energia é a mesma da situação em que ele se reduz de $1/2$ na primeira direção e aumenta de 2 na segunda – veja a figura (4.4). O mesmo ocorre para um cisalhamento de s ou de $-s$, como demonstra a figura (4.5).

4.3 Cálculo das forças de viscosidade

Como vimos no capítulo 2, as forças internas de viscosidade, que dependem das velocidades relativas entre partículas próximas, são calculadas a partir de uma função $W(\mathbf{q}, \dot{\mathbf{q}})$, que mede a potência dissipada por essas forças num dado instante.

Denotaremos por $\omega(u)$ a energia dissipada pelas forças de viscosidade nas vizinhanças da partícula u, por unidade de volume de repouso e por unidade de tempo, ou seja, w é a *densidade de potência dissipada*. Portanto, a potência $\Omega(t)$ dissipada numa região $R \subseteq U$, num instante t, é $\Omega(t) = \int_R \omega(u)|\mathrm{J}_{\mathbf{q}_u}|\,du$. Por sua vez, a energia total dissipada pelas forças de viscosidade interna num intervalo $I = [t_a, t_b]$ é $\int_{t_a}^{t_b} \Omega(t)dt$.

4.3.1 Campo de velocidades

Para uma partícula $u \in \bar{U}$ e um vetor $\delta \in \mathbb{R}^3$, a velocidade da partícula $u + \delta$ num estado $(\mathbf{f}, \dot{\mathbf{f}})$ é

$$\dot{\mathbf{f}}(u+\delta) = \dot{\mathbf{f}}(u) + \mathrm{J}_{\dot{\mathbf{f}}}(u)\,\delta + O(|\delta|^2)$$

onde $\mathrm{J}_{\dot{\mathbf{f}}}$ é a matriz jacobiana do campo de velocidades $\dot{\mathbf{f}}$, isto é,

$$\mathrm{J}_{\dot{\mathbf{f}}}(u) = \begin{bmatrix} \dfrac{\partial \dot{f}_x}{\partial u_x}(u) & \dfrac{\partial \dot{f}_x}{\partial u_y}(u) & \dfrac{\partial \dot{f}_x}{\partial u_z}(u) \\ \dfrac{\partial \dot{f}_y}{\partial u_x}(u) & \dfrac{\partial \dot{f}_y}{\partial u_y}(u) & \dfrac{\partial \dot{f}_y}{\partial u_z}(u) \\ \dfrac{\partial \dot{f}_z}{\partial u_x}(u) & \dfrac{\partial \dot{f}_z}{\partial u_y}(u) & \dfrac{\partial \dot{f}_z}{\partial u_z}(u) \end{bmatrix}$$

para todo $u \in \bar{U}$. Dizemos que a matriz $\mathrm{J}_{\dot{\mathbf{f}}}$ descreve em primeira ordem o campo de velocidades das partículas, nas vizinhanças de cada partícula u. Note que $\mathrm{J}_{\dot{\mathbf{f}}}$ também pode ser escrita como

$$\frac{d}{dt}\mathrm{J}_{\mathbf{f}} = \dot{\mathrm{J}}_{\mathbf{f}}$$

4.3.2 Tensor de escoamento

Comprova-se experimentalmente que, para a grande maioria dos materiais, as forças de viscosidade dependem apenas das diferenças de primeira ordem entre as velocidades das partículas vizinhas; ou seja, da matriz $\dot{\mathbf{C}}(u) = \mathrm{J}_{\dot{\mathbf{r}}}(u)$, onde

$$\dot{\mathbf{r}} = \frac{d}{dt}(\mathbf{p} \circ \mathbf{q}_u^{-1})$$

e portanto $\dot{\mathbf{C}}(u) = \frac{d}{dt}C(u)$. Uma vez que a configuração de repouso $\mathbf{q}_u$ não depende do tempo, a matriz $\dot{\mathbf{C}}$ pode ser escrita como

$$\dot{\mathbf{C}}(u) = \dot{\mathrm{J}}_{\mathbf{p}}(u)\,\mathrm{J}_{\mathbf{q}_u}^{-1}(\mathbf{q}_u(u))$$

Dizemos que $\dot{\mathbf{C}}$ é o *tensor de escoamento* do material.

4.3.3 Invariância sob rotação

As forças de viscosidade não são afetadas por movimentos rígidos do material (translações e rotações na configuração corrente). Portanto, elas dependem apenas da derivada temporal do tensor métrico $\mathbf{T} = \mathbf{C}^\top \mathbf{C}$:

$$\dot{\mathbf{T}} = \frac{d}{dt}(\mathbf{C}^\top \mathbf{C}) = \dot{\mathbf{C}}^\top \mathbf{C} + \mathbf{C}^\top \dot{\mathbf{C}}$$

Note que $\dot{\mathbf{T}}$, assim como $\mathbf{T}$, é uma matriz simétrica. Pode-se verificar que, se δ é o deslocamento entre as partículas u e v na configuração de repouso $\mathbf{q}_u$, então

$$\delta^\top \dot{\mathbf{T}} \delta = 2\dot{\epsilon}^\top \epsilon$$

onde ϵ é o deslocamento entre u e v na configuração corrente $\mathbf{p}$, e $\dot{\epsilon}$ é a velocidade relativa entre u e v.

4.3.4 Viscosidade em materiais isotrópicos

Num material isotrópico, as forças de viscosidade não são afetadas por uma mudança de coordenadas na configuração de referência, ou seja, são as mesmas para $\dot{\mathbf{T}}$ e $\mathbf{S}^{-1}\dot{\mathbf{T}}\mathbf{S}$, onde $\mathbf{S}$ é uma matriz de rotação arbitrária (que não varia com o tempo).

Portanto, a potência dissipada pelas forças de viscosidade é dada por uma função $\Omega(\xi_1, \xi_2, \xi_3)$ dos auto-valores ξ_1, ξ_2, ξ_3 da matriz $\dot{\mathbf{T}}$.

4.3.5 Análise para escoamento lento

A densidade de potência dissipada $w(u)$ deve ser mínima (zero) quando $\dot{\mathbf{T}}$ é nula, e simétrica nos três auto-valores de $\dot{\mathbf{T}}$. Portanto, para deformações graduais ($||\dot{\mathbf{T}}||$ pequeno, ou seja, escoamento lento), Ω deve ser uma função quadrática, homogênea e simétrica dos três auto-valores. Assim como no caso da densidade de energia elástica para pequenas deformações, concluímos que Ω é uma combinação de dois termos:

$$\Omega(\xi_1, \xi_2, \xi_3) = \eta_1(u)\,\mathcal{H}^*(\xi_1, \xi_2, \xi_3) + \eta_2(u)\,\mathcal{J}^*(\xi_1, \xi_2, \xi_3)$$

onde $\mathcal{H}^*$ e $\mathcal{J}^*$ são funções independentes quaisquer de $\mathbb{H}_3^2$, e $\eta_1(u), \eta_2(u)$ são coeficientes reais não negativos, que dependem da viscosidade do material nas vizinhanças da partícula u.

4.3.6 Módulos de viscosidade

Do ponto de vista físico, é conveniente tomar

$$\mathcal{H}^* \quad = \quad \frac{(\xi_1 + \xi_2 + \xi_3)^2}{2}$$

$$\mathcal{J}^* = \frac{2}{3} \frac{(\xi_1^2 + \xi_2^2 + \xi_3^2 - \xi_1\xi_2 - \xi_1\xi_3 - \xi_2\xi_3)}{2}$$

Com esta escolha, uma análise semelhante à da seção (4.2.8) mostra que η_1 mede a resistência do material a escoamentos lentos de expansão ou contração uniforme, enquanto que η_2 mede a resistência a escoamentos lentos de cisalhamento. Estes coeficientes são conhecidos como os *módulos de viscosidade* do material.

4.3.7 Fórmula para escoamentos gerais

No nosso simulador, adotamos a fórmula

$$\Omega(\Pi, \Xi) = \frac{\eta_1(u)}{2} \left(\Pi^2\right) + \frac{2\,\eta_2(u)}{3} \left(\Pi^2 - 3\Xi\right) \tag{4.14}$$

onde Π, Ξ são calculados a partir de $\dot{\mathbf{T}}(u)$ como

$$\begin{aligned} \Pi &= \dot{T}_{11} + \dot{T}_{22} + \dot{T}_{33} \\ \Xi &= \dot{T}_{11}^{(2)} + \dot{T}_{22}^{(2)} + \dot{T}_{33}^{(2)} \end{aligned}$$

Capítulo 5

Modelo geométrico

Ao aproximar o conjunto discreto de partículas $\tilde{U}$ por uma região deformável U do $\mathbb{R}^3$, obtemos um número infinito de graus de liberdade. A fim de descrever a configuração do sistema com um número finito de coordenadas generalizadas, utilizamos o *método dos elementos finitos* [Bur87], aproximando o conjunto U por uma colagem de fragmentos de geometria simples — os *elementos finitos*. A deformação de cada elemento é restrita de modo que sua configuração possa ser descrita por um número fixo e pequeno de coordenadas.

5.1 O elemento tetraédrico

Neste trabalho, utilizamos um só tipo de elemento finito, o tetraedro, e permitimos apenas deformações afins de cada tetraedro. Deste modo, cada elemento mantém sempre sua forma tetraédrica.

5.1.1 Geometria de um elemento

Um elemento genérico T consiste de um conjunto de partículas, que na sua *configuração de referência* $\mathbf{r}$ preenchem um tetraedro de vértices $u_1 = [0,0,0]$, $u_2 = [1,0,0]$, $u_3 = [0,1,0]$ e $u_4 = [0,0,1]$, com densidade uniforme. A esse elemento está associada uma configuração (fixa) de repouso $\mathbf{q}$ num sistema de coordenadas próprio, bem como uma configuração corrente $\mathbf{p}$, definida em cada instante durante a simulação.

Vamos sempre numerar os vértices de um tetraedro de maneira que os vértices u_1, u_2, u_3 são vistos em ordem anti-horária por um observador postado no vértice u_4. Veja a figura (5.1).

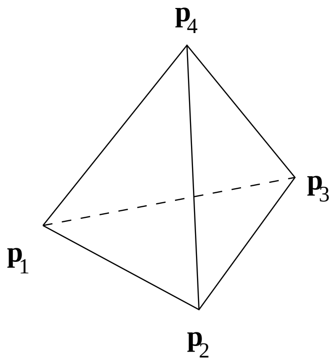


Figura 5.1: Um tetraedro com vértices em numeração positiva

5.1.2 Coordenadas baricêntricas

Cada partícula de um elemento T pode ser identificada unicamente pelas suas *coordenadas baricêntricas* $\alpha_1, \alpha_2, \alpha_3, \alpha_4$, onde $[\alpha_2, \alpha_3, \alpha_4]$ são as suas coordenadas na configuração de referência, e $\alpha_1 = 1 - \alpha_2 - \alpha_3 - \alpha_4$. Note que cada partícula u do tetraedro possui uma quádrupla única de coordenadas baricêntricas, e que essas coordenadas não variam no tempo. Note também que $0 \leq \alpha_1, \alpha_2, \alpha_3, \alpha_4 \leq 1$.

Se $\mathbf{f}$ é uma configuração do tetraedro T, e $\mathbf{f}_1, \ldots, \mathbf{f}_4$ são as posições em $\mathbf{f}$ dos vértices $u_1, \ldots, u_4$ de T, respectivamente, então a posição $\mathbf{f}(u)$ de uma partícula $u \in T$ é

$$\mathbf{f}(u) = \alpha_1\mathbf{f}_1 + \alpha_2\mathbf{f}_2 + \alpha_3\mathbf{f}_3 + \alpha_4\mathbf{f}_4 \tag{5.1}$$

Derivando-se a fórmula (5.1) obtemos fórmulas que permitem calcular a velocidade $\mathbf{v}(u)$ e a aceleração $\mathbf{a}(u)$ de cada partícula u a partir das velocidades $\mathbf{v}_1, \ldots, \mathbf{v}_4$ e acelerações $\mathbf{a}_1, \ldots, \mathbf{a}_4$ dos vértices do tetraedro que contém u:

$$\mathbf{v}(u) = \alpha_1\mathbf{v}_1 + \alpha_2\mathbf{v}_2 + \alpha_3\mathbf{v}_3 + \alpha_4\mathbf{v}_4 \tag{5.2}$$

$$\mathbf{a}(u) = \alpha_1\mathbf{a}_1 + \alpha_2\mathbf{a}_2 + \alpha_3\mathbf{a}_3 + \alpha_4\mathbf{a}_4 \tag{5.3}$$

5.1.3 Deformação de um elemento

Se $\mathbf{f}_i = [x_i, y_i, z_i]$, para $i = 1, \ldots, 4$, são as coordenadas do vértice u_i do tetraedro T na configuração $\mathbf{f}$, então a equação (5.1) pode ser escrita na forma matricial

$$\mathbf{f}(u) = \mathbf{f}_1 + \mathbf{M}u$$

onde $\mathbf{M}$ é a matriz

$$\mathbf{M} = \begin{bmatrix} x_2 - x_1 & x_3 - x_1 & x_4 - x_1 \\ y_2 - y_1 & y_3 - y_1 & y_4 - y_1 \\ z_2 - z_1 & z_3 - z_1 & z_4 - z_1 \end{bmatrix}$$

Ou seja, $\mathbf{M}$ é a parte linear da transformação afim que leva o tetraedro T da configuração de referência $\mathbf{r}$ para a configuração $\mathbf{f}$. Verifica-se que o determinante de $\mathbf{M}$ é positivo, dado que é seis vezes o volume de T na configuração $\mathbf{f}$. Portanto, a inversa de $\mathbf{M}$ existe e mapeia as partículas de T na configuração $\mathbf{f}$ de volta para a configuração de referência $\mathbf{r}$.

Vamos denotar por $\mathbf{A}$ e $\mathbf{B}$ as matrizes $\mathbf{M}$ correspondentes a levar o tetraedro T de $\mathbf{r}$ para $\mathbf{q}$, e de $\mathbf{r}$ para $\mathbf{p}$, respectivamente. A matriz que corresponde a levar T da configuração de repouso $\mathbf{q}$ para a configuração corrente $\mathbf{p}$ é portanto

$$\mathbf{C} = \mathbf{B}\mathbf{A}^{-1}$$

A figura (5.2) ilustra os mapeamentos entre as configurações de referência, de repouso e corrente de um tetraedro T.

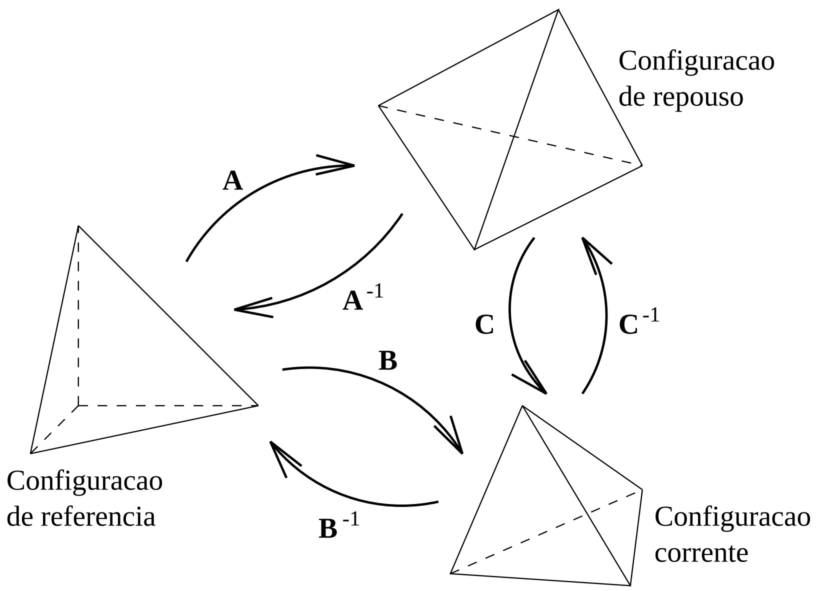


Figura 5.2: Mapeamento entre as configurações de um tetraedro

5.1.4 Propriedades materiais dos elementos

Em nosso modelo, as propriedades materiais (densidade de massa, módulos de elasticidade e de viscosidade) são uniformes no interior de cada tetraedro, mas podem mudar de um para outro.

Note que a densidade de um tetraedro varia quando seu volume muda, mas a sua massa permanece constante. Se m é a massa (fixa) de um tetraedro e V o seu volume em uma configuração qualquer, a sua densidade ρ nessa mesma configuração é dada por $\rho = m/V$.

Outros parâmetros associados a cada tetraedro são o módulo volumétrico α, o módulo de rigidez β, o primeiro módulo de viscosidade η_1 e o segundo módulo de viscosidade η_2 (veja as seções (4.2.8) e (4.3.4)). Subentende-se que estes parâmetros podem variar de um tetraedro para outro, mas são constantes dentro de cada tetraedro.

5.2 Corpos complexos

Qualquer objeto poliédrico fechado pode ser modelado por um número finito de tetraedros. Por exemplo, a figura (5.3) mostra um modelo de elementos finitos para um corpo em forma de cubo, constituído de seis tetraedros.

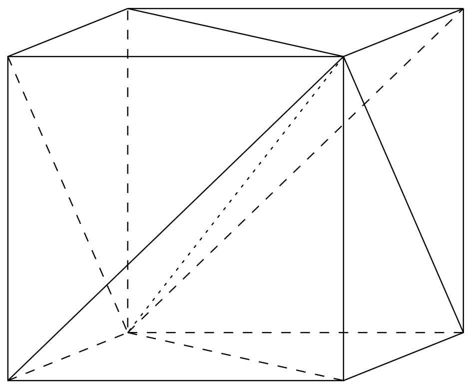


Figura 5.3: Um cubo feito de seis tetraedros

Dois tetraedros T_1 e T_2 podem estar colados por uma face, por uma aresta ou por um vértice. Em qualquer configuração do sistema, entretanto, os tetraedros T_1 e T_2 possuem interiores disjuntos. Note que cada subconjunto conexo de tetraedros do modelo constitui um corpo isolado, e que todo corpo construído por uma união arbitrária de tetraedros apresenta forma poliédrica, com superfície triangularizada. Note também que se unimos dois corpos apenas por uma aresta temos uma *dobradiça*, e se o fazemos por um vértice apenas, temos uma *articulação universal* ou *esférica*.

O conjunto de *vértices do modelo* consiste de todos os vértices de todos os elementos; porém, quando dois ou mais vértices de elementos são colados entre si, eles contam como um único vértice do modelo. Da mesma forma, definimos o conjunto de *arestas do modelo*. Por exemplo, o modelo da figura (5.3) tem 8 vértices e 19 arestas. Note que dois vértices do modelo são unidos por uma aresta se e somente se pertencerem a um mesmo elemento. No que segue, usaremos n_v, n_a e n_t para denotar os números de vértices, arestas e elementos (tetraedros) do modelo.

5.2.1 Definição das coordenadas generalizadas

A geometria de cada corpo na configuração corrente é inteiramente determinada pelas coordenadas correntes dos vértices dos seus tetraedros. Portanto, as coordenadas generalizadas do sistema são as coordenadas cartesianas dos vértices do modelo.

O estado do sistema é descrito pelo vetor $\mathbf{q} = [q_1, \ldots, q_n]$ de coordenadas cartesianas dos vértices, onde $n = 3n_v$ é o número total de coordenadas generalizadas, e pelo vetor $\dot{\mathbf{q}} = [\dot{q}_1, \ldots, \dot{q}_n]$ de velocidades lineares dos vértices. Cada tripla $[q_j, q_{j+1}, q_{j+2}]$ de coordenadas tal que $j \bmod 3 = 1$ equivale às coordenadas cartesianas de um vértice do modelo $\mathbf{r}_i = [x_i, y_i, z_i]$, onde $i = \lceil \frac{j}{3} \rceil$; analogamente, para cada vértice de velocidade linear $\mathbf{v}_i = d\mathbf{r}_i/dt$, existe uma única tripla $[\dot{q}_j, \dot{q}_{j+1}, \dot{q}_{j+2}]$ de velocidades generalizadas.

5.3 Energia e potência dissipada

A energia cinética K e a energia potencial P do sistema, assim como a potência W dissipada pelas forças internas de viscosidade, são simplesmente a soma dessas grandezas para cada elemento do modelo. Note que uma configuração $\mathbf{f} : U \to \mathbb{R}^3$ do sistema não é diferenciável nos pontos que pertencem às faces dos tetraedros. Portanto, a densidade de massa ρ, a densidade de energia elástica ϕ, e a densidade de potência dissipada ω não são definidas nesses pontos. Entretanto, o conjunto total desses pontos tem volume zero, e como dissemos no capítulo 4, tem massa e energia nulas.

5.3.1 Energia cinética de um elemento

A energia cinética K de um tetraedro T de massa m pode ser calculada como

$$K = \int_0^1 \int_0^{1-\alpha_4} \int_0^{1-\alpha_3-\alpha_4} \frac{\rho}{2} \mathbf{v}^2 \, d\alpha_2 \, d\alpha_3 \, d\alpha_4$$

onde $\mathbf{v}$ é a velocidade no estado corrente de uma partícula do tetraedro com coordenadas baricêntricas $[\alpha_2, \alpha_3, \alpha_4]$, e $\rho = 6m$ é a densidade do tetraedro na configuração de referência. Pela fórmula (5.2), temos que

$$K = 3m \int_0^1 \int_0^{1-\alpha_4} \int_0^{1-\alpha_3-\alpha_4} (\alpha_1 \mathbf{v}_1 + \alpha_2 \mathbf{v}_2 + \alpha_3 \mathbf{v}_3 + \alpha_4 \mathbf{v}_4)^2 \, d\alpha_2 \, d\alpha_3 \, d\alpha_4$$

onde $\mathbf{v}_1, \ldots, \mathbf{v}_4$ são as velocidades correntes dos vértices de T. Integrando, esta fórmula se reduz a

$$K = \frac{m}{20}(\mathbf{v}_1^2 + \mathbf{v}_2^2 + \mathbf{v}_3^2 + \mathbf{v}_4^2 + \mathbf{v}_1 \cdot \mathbf{v}_2 + \mathbf{v}_1 \cdot \mathbf{v}_3 + \mathbf{v}_1 \cdot \mathbf{v}_4 + \mathbf{v}_2 \cdot \mathbf{v}_3 + \mathbf{v}_2 \cdot \mathbf{v}_4 + \mathbf{v}_3 \cdot \mathbf{v}_4)$$

5.3.2 Energia elástica de um elemento

Sejam Δ, Γ e Σ os invariantes de rotação dentro de um tetraedro T, calculados a partir do tensor de deformação $\mathbf{C}$ (veja a seção (4.2)). Estes invariantes, e portanto a densidade de energia elástica Φ, são constantes no interior de T.

Calculamos a energia elástica P do tetraedro multiplicando seu volume de repouso V^0 pelo valor da densidade de energia elástica no seu interior, dada pela fórmula (4.13):

$$P = V^0 \left[\frac{\alpha}{32} \left(\Delta^2 + \frac{1}{\Delta^2} - 2 \right) + \frac{\beta}{6} \left(\Gamma^2 - 3\Sigma \right) \right] \tag{5.4}$$

5.3.3 Potência dissipada em um elemento

A matriz $\dot{\mathbf{T}}$, para um dado estado do sistema, é uniforme no interior de cada tetraedro T. Portanto, as quantidades Π e Ξ, bem como a densidade de potência dissipada Ω, calculada através da fórmula (4.14), são as mesmas em todos os pontos do interior de T. Calculamos a potência dissipada no interior do tetraedro T como

$$W = V \left[\frac{\eta_1}{2} \left(\Pi^2 \right) + \frac{2\,\eta_2}{3} \left(\Pi^2 - 3\Xi \right) \right] \tag{5.5}$$

onde V é o volume de T no estado corrente.

Capítulo 6

Simulação do movimento

Tipicamente, sistemas interessantes de simular apresentam algumas dezenas ou centenas de tetraedros viscoelásticos, sujeitos a gravidade, colisões e restrições holonômicas simples. Neste capítulo, descrevemos as ferramentas numéricas que adotamos para simular o movimentos de tais sistemas, exceto pelos algoritmos de deteção e tratamento de colisões e outros eventos discretos, deixados para o capítulo seguinte.

Como vimos no capítulo 3, o sistema de equações de movimento de um sistema dinâmico em sua forma mais completa pode ser formulado matricialmente como

$$\begin{bmatrix} \mathbf{M} & \mathbf{D}^\top \\ \mathbf{J} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \boldsymbol{\psi} \end{bmatrix} \tag{6.1}$$

Para calcular a evolução do sistema ao longo do tempo é necessário, em primeiro lugar, resolver o sistema linear acima em $\ddot{\mathbf{q}}$ e $\boldsymbol{\lambda}$, para o estado inicial do sistema. Em seguida, é preciso integrar a equação $\mathbf{M}\ddot{\mathbf{q}} = \mathbf{f} - \mathbf{D}^\top\boldsymbol{\lambda}$, para obtermos um novo estado num instante posterior. Se um evento discreto (por exemplo, uma colisão) é detetado entre estes dois estados, o último deles deve ser descartado, e um novo estado deve ser calculado no instante do evento. Todo evento requer um tratamento específico, que na maioria das vezes causa uma mudança nas equações de movimento. Uma vez obtido um novo estado válido para o sistema, todo o processo é repetido. Dada a complexidade do modelo, não deve surpreender o fato de que cada iteração deste processo de simulação envolve um custo computacional significativo, da ordem de milhões de instruções aritméticas elementares.

6.1 Fase de pré-processamento

Em nosso modelo de elementos finitos, a massa de cada tetraedro é invariável no tempo e a energia cinética do sistema é uma função quadrática nas velocidades $\dot{q}_i$, e independente das coordenadas q_i. Em conseqüência, a matriz de massa $\mathbf{M}$ é constante, o que nos permite calculá-la uma única vez, antes de dar início à simulação propriamente dita. Mais ainda, calculamos uma fatoração de $\mathbf{M}$, visando facilitar a resolução do sistema linear (6.1) no decorrer da simulação.

Exceto por $\mathbf{M}$, todas as matrizes e vetores em (6.1), incluindo as várias forças individuais que compõem $\mathbf{f}$, variam ao longo do tempo e portanto precisam ser calculadas repetidas vezes durante a simulação. Em particular, as matrizes $\mathbf{J}$ e $\mathbf{D}$ e o vetor $\boldsymbol{\psi}$ têm de ser calculados diretamente para cada novo estado do sistema, conforme descrito no capítulo 3.

6.1.1 Cálculo da matriz de massa

Na seção (2.3.2), vimos que os elementos de $\mathbf{M}$ são dados por

$$M_{ij} = \frac{\partial^2 K}{\partial \dot{q}_i \partial \dot{q}_j}$$

para todo $i, j = 1, \ldots, n$. A matriz $\mathbf{M}$ é simétrica, uma vez que

$$\frac{\partial^2 K}{\partial \dot{q}_i \partial \dot{q}_j} = \frac{\partial^2 K}{\partial \dot{q}_j \partial \dot{q}_i}$$

Seja K_l a energia cinética do l-ésimo tetraedro do modelo, para $l = 1, \ldots, n_t$. Partindo de uma matriz $\mathbf{M}$ inicialmente nula, nós calculamos os elementos M_{ij} tais que $i \geq j$ somando separadamente as contribuições de cada tetraedro, ou seja, para cada tetraedro T_l calculamos as doze derivadas de K_l em relação às coordenadas dos quatro vértices do mesmo, e as somamos aos devidos elementos de $\mathbf{M}$.

Pode-se verificar que $\mathbf{M}$ é uma matriz de blocos 3×3, onde cada bloco corresponde a um par ordenado (u, v) de vértices do modelo, para $1 \leq u, v \leq n_v$. Para cada par (u, v), se u é igual a v ou se há uma aresta entre os vértices distintos u e v, então o bloco correspondente a esse par é uma matriz diagonal, e os três elementos da diagonal são positivos e iguais entre si; caso contrário, se não há nenhuma aresta entre os vértices distintos u e v, o bloco é uma matriz nula. Assim, cada bloco não nulo de $\mathbf{M}$ corresponde a um vértice ou a uma aresta do modelo, e vice-versa. Um vértice u produz um bloco não nulo centrado na diagonal principal de $\mathbf{M}$, cujos elementos não nulos são M_{ii} tais que $i \in \{3u-2, 3u-1, 3u\}$. Uma aresta entre os vértices distintos u e v produz dois blocos não nulos fora da diagonal principal de $\mathbf{M}$, cujos elementos não nulos são M_{ij} e M_{ji} tais que $i = 3u - r$, $j = 3v - r$ e $r \in \{0, 1, 2\}$. Num bloco de vértice, cada elemento da diagonal é um décimo da soma das massas dos tetraedros incidentes ao vértice, enquanto que num bloco de aresta cada elemento da diagonal é um vigésimo da soma das massas dos tetraedros incidentes à aresta.

Note que o número de elementos não nulos em $\mathbf{M}$ é exatamente $3n_v + 6n_a$, para qualquer modelo de tetraedros. Em modelos práticos, o número de arestas usualmente é linear no número de vértices. Em tais casos, a matriz $\mathbf{M}$ é *esparsa*, e portanto o espaço necessário para armazená-la é linear no tamanho do modelo, se armazenarmos apenas os elementos não nulos — como fazemos em nossa implementação.

A título de exemplo, a matriz de massa para um corpo constituído de um único tetraedro de massa m é

$$\mathbf{M} = \frac{m}{20}\begin{bmatrix} 2&0&0&1&0&0&1&0&0&1&0&0\\ 0&2&0&0&1&0&0&1&0&0&1&0\\ 0&0&2&0&0&1&0&0&1&0&0&1\\ \\ 1&0&0&2&0&0&1&0&0&1&0&0\\ 0&1&0&0&2&0&0&1&0&0&1&0\\ 0&0&1&0&0&2&0&0&1&0&0&1\\ \\ 1&0&0&1&0&0&2&0&0&1&0&0\\ 0&1&0&0&1&0&0&2&0&0&1&0\\ 0&0&1&0&0&1&0&0&2&0&0&1\\ \\ 1&0&0&1&0&0&1&0&0&2&0&0\\ 0&1&0&0&1&0&0&1&0&0&2&0\\ 0&0&1&0&0&1&0&0&1&0&0&2 \end{bmatrix}$$

6.1.2 Fatoração da matriz de massa

Podemos escrever a energia cinética K como

$$K = \frac{\dot{\mathbf{q}}^\top \mathbf{M} \dot{\mathbf{q}}}{2}$$

Pela definição de energia cinética temos que $K > 0$ para todo $\dot{\mathbf{q}}$ não nulo, o que significa que a matriz $\mathbf{M}$ é *positivo-definida.* O fato de $\mathbf{M}$ ser simétrica e positivo-definida nos permite aplicar o método de Choleski [BF89] para fatorá-la na forma

$$\mathbf{M} = \mathbf{L}\mathbf{D}\mathbf{L}^\top$$

onde $\mathbf{L}$ é uma matriz triangular inferior de diagonal unitária e $\mathbf{D}$ é uma matriz diagonal cujos elementos na diagonal são todos positivos. Para resolver um sistema $\mathbf{Mx} = \mathbf{y}$ basta resolver os seguintes sistemas lineares:

$$\begin{aligned} \mathbf{Lv} &= \mathbf{y} \\ \mathbf{Du} &= \mathbf{v} \\ \mathbf{L}^\top\mathbf{x} &= \mathbf{u} \end{aligned} \tag{6.2}$$

onde $\mathbf{u}$ e $\mathbf{v}$ são vetores temporários. O primeiro sistema linear é resolvido por uma *substituição* para frente, o segundo é trivial e o terceiro é resolvido por uma substituição para trás.

Quando $\mathbf{M}$ é esparsa, verifica-se na maioria das vezes que a matriz $\mathbf{L}$ também é esparsa, embora menos do que $\mathbf{M}$. Em nossos testes, observamos que para matrizes de massa de bom tamanho (de 300 a 1000 linhas), o número de elementos não nulos em $\mathbf{M}$ não chega a 3% do seu número total de elementos. Quando essas matrizes são fatoradas, o número de elementos não nulos no fator $\mathbf{L}$ não chega a 7% de seu tamanho total.

O número de novos elementos não nulos em $\mathbf{L}$ depende das relações de vizinhança entre os tetraedros do modelo, e da ordem das linhas e colunas de $\mathbf{M}$. Existe sempre uma permutação das linhas e colunas de $\mathbf{M}$ que minimiza esse número, e há várias técnicas na literatura especializada para encontrar eficientemente uma permutação suficientemente próxima da ótima [Pis84]. Em nosso sistema não nos preocupamos em realizar essa otimização.

6.2 Cálculo das forças diretas

Como vimos no capítulo 2, o vetor de forças diretas $\mathbf{f}$ sobre os vértices do modelo é dado pela soma de quatro componentes: $\mathbf{f} = \mathbf{f}^{\mathrm{f}} + \mathbf{f}^{\mathrm{d}} + \mathbf{f}^{\mathrm{c}} + \mathbf{f}^{\mathrm{e}}$, sendo que as forças externas $\mathbf{f}^{\mathrm{e}}$ são dadas pelo usuário ou calculados pelo próprio sistema, como no caso das forças de gravidade. No nosso modelo, as forças fictícias $\mathbf{f}^{\mathrm{f}}$ são sempre nulas, porque a energia cinética K é independente de $\mathbf{q}$. As forças internas conservativas $\mathbf{f}^{\mathrm{c}}$ e dissipativas $\mathbf{f}^{\mathrm{d}}$, que em nosso sistema correspondem às forças elásticas e às forças de viscosidade, respectivamente, são calculadas pelo sistema de animação para cada estado obtido durante a simulação.

6.2.1 Cálculo das forças externas

Seja T um tetraedro cujos vértices possuem posições $\mathbf{p}_1, \ldots, \mathbf{p}_4$ na configuração corrente. Seja $\mathbf{p} = \alpha_1\mathbf{p}_1 + \alpha_2\mathbf{p}_2 + \alpha_3\mathbf{p}_3 + \alpha_4\mathbf{p}_4$ a posição de uma partícula u do tetraedro T, onde $\alpha_1, \ldots, \alpha_4$ são as coordenadas baricêntricas dessa partícula em relação aos vértices de T. Dada uma força $\mathbf{f}_u \in \mathbb{R}^3$ sobre essa partícula, calculamos as forças generalizadas equivalentes $\bar{\mathbf{f}}_i = [\bar{f}_i^x, \bar{f}_i^y, \bar{f}_i^z]$, sobre os quatro vértices de T, como $\bar{\mathbf{f}}_i = \alpha_i \mathbf{f}_u$, para $i = 1, \ldots, 4$ (veja a seção 2.2.1).

Seja $\mathbf{f}_T \in \mathbb{R}^3$ uma força externa distribuída uniformemente sobre todas as partículas do tetraedro T. Pode-se verificar que a força generalizada equivalente sobre os vértices de T é $\bar{\mathbf{f}}_i = \frac{1}{4}\mathbf{f}_T$, para $i = 1, \ldots, 4$. Por exemplo, suponha que $\mathbf{f}_T = m\,\mathbf{g}$ é a força da gravidade sobre T, onde m é a massa de T e $\mathbf{g} \in \mathbb{R}^3$ é a aceleração (constante) da gravidade em coordenadas cartesianas. Neste caso, as forças generalizadas sobre os vértices de T são

$\bar{\mathbf{f}}_i = \frac{1}{4}\, m\, \mathbf{g}$. Note que a força total da gravidade sobre um vértice é a soma das contribuições devidas a cada tetraedro incidente a esse mesmo vértice.

6.2.2 Cálculo das forças elásticas

As forças de elasticidade $\mathbf{f}^c$ são calculadas de maneira eficiente através da aplicação sistemática da regra da cadeia [BS83] às derivadas da energia elástica P_j de cada tetraedro T_j do modelo, para $j = 1, \ldots, n_t$.

Para cada coordenada generalizada q_i, a correspondente força de elasticidade f_i^c é o oposto da soma das derivadas $\frac{\partial P_j}{\partial q_i}(\mathbf{q})$, para os tetraedros T_j que apresentam a coordenada q_i num dos quatro vértices. Para obter as doze derivadas de P_j associadas ao tetraedro T_j, primeiro calculamos os tensores de deformação $\mathbf{B}$ e $\mathbf{C}$ e as medidas de deformação Δ, Γ e Σ de T_j, e então aplicamos a fórmula a seguir para cada coordenada q_i presente em um dos vértices desse tetraedro:

$$\frac{\partial P_j}{\partial q_i} = \sum_{k,l,r,s,u,v} \left\{ \left[\left(\frac{\partial P_j}{\partial \Delta}\frac{\partial \Delta}{\partial T_{kl}} + \frac{\partial P_j}{\partial \Gamma}\frac{\partial \Gamma}{\partial T_{kl}} + \frac{\partial P_j}{\partial \Sigma}\frac{\partial \Sigma}{\partial T_{kl}} \right) \frac{\partial T_{kl}}{\partial C_{rs}} \right] \frac{\partial C_{rs}}{\partial B_{uv}} \right\} \frac{\partial B_{uv}}{\partial q_i}$$

onde T_{kl} são os elementos do tensor métrico $\mathbf{T} = \mathbf{C}^\top \mathbf{C}$.

As derivadas de P_j em relação a Δ, Γ e Σ, de acordo com a fórmula (5.4), são

$$\begin{aligned} \frac{\partial P_j}{\partial \Delta} &= V_j^0 \frac{\alpha}{16}\left(\Delta - \frac{1}{\Delta^3}\right) \\ \frac{\partial P_j}{\partial \Gamma} &= V_j^0 \frac{\beta}{3}\Gamma \\ \frac{\partial P_j}{\partial \Sigma} &= -V_j^0 \frac{\beta}{2} \end{aligned}$$

e destes últimos em relação a T_{kl} são

$$\begin{aligned} \frac{\partial \Delta}{\partial T_{kl}} &= T_{kl}^{(2)} \\ \frac{\partial \Gamma}{\partial T_{kl}} &= \begin{cases} 1 & \text{se } k = l \\ 0 & \text{se } k \neq l \end{cases} \\ \frac{\partial \Sigma}{\partial T_{kl}} &= \begin{cases} T_{11} + T_{22} + T_{33} - T_{kl} & \text{se } k = l \\ -T_{lk} & \text{se } k \neq l \end{cases} \end{aligned}$$

Note que, pelo fato da matriz $\mathbf{T}$ ser simétrica, há apenas seis derivadas independentes de Δ, Γ e Σ em relação a $\mathbf{T}$. As derivadas de $\mathbf{T}$ em relação a $\mathbf{C}$ são dadas por

$$\frac{\partial T_{kl}}{\partial C_{rs}} = \begin{cases} C_{rl} & \text{se } s = k \\ 0 & \text{se } s \neq k \end{cases} + \begin{cases} C_{rk} & \text{se } s = l \\ 0 & \text{se } s \neq l \end{cases}$$

As derivadas de $\mathbf{C}$ em relação a $\mathbf{B}$ são constantes para cada tetraedro:

$$\frac{\partial C_{rs}}{\partial B_{uv}} = \begin{cases} (A^{-1})_{vs} & \text{se } r = u \\ 0 & \text{se } r \neq u \end{cases}$$

A matriz $\mathbf{B}$ é linear nas coordenadas q_i dos vértices de T_j, e portanto cada uma de suas derivadas (que vale 0, 1 ou -1) é conhecida de antemão.

Na verdade, basta calcular apenas 9 das 12 derivadas $\partial P_j / \partial q_i$, uma vez que a soma das forças internas de elasticidade em cada tetraedro é nula. A força no quarto vértice é igual e oposta à soma das forças nos outros três. O mesmo se aplica às derivadas $\partial W_j / \partial \dot{q}_i$, que dão as forças internas de viscosidade.

6.2.3 Cálculo das forças de viscosidade

A força interna de viscosidade em um vértice, assim como para a força de elasticidade, é dada pela soma das contribuições de todos os tetraedros incidentes a esse vértice. Mais precisamente, para cada velocidade generalizada $\dot{q}_i$, a correspondente força de viscosidade f_i^{d} é o oposto da metade da soma das derivadas $\frac{\partial W_j}{\partial \dot{q}_i}(\mathbf{q}, \dot{\mathbf{q}})$, para os tetraedros T_j que apresentam a coordenada q_i num dos quatro vértices.

Suponha que as forças de elasticidade já foram calculadas. Calculamos então as matrizes $\dot{\mathbf{B}}$, $\dot{\mathbf{C}}$ e $\dot{\mathbf{T}}$, e os valores de Π e Ξ, para cada tetraedro. Para cada velocidade $\dot{q}_i$ associada a um vértice do tetraedro T_j calculamos:

$$\frac{\partial W_j}{\partial \dot{q}_i} = \sum_{k,l,r,s,u,v} \left\{ \left[\left(\frac{\partial W_j}{\partial \Pi} \frac{\partial \Pi}{\partial \dot{T}_{kl}} + \frac{\partial W_j}{\partial \Xi} \frac{\partial \Xi}{\partial \dot{T}_{kl}} \right) \frac{\partial \dot{T}_{kl}}{\partial \dot{C}_{rs}} \right] \frac{\partial \dot{C}_{rs}}{\partial \dot{B}_{uv}} \right\} \frac{\partial \dot{B}_{uv}}{\partial \dot{q}_i}$$

Pela fórmula (5.5), temos que

$$\begin{aligned} \frac{\partial W_j}{\partial \Pi} &= V_j \left(\eta_1 + \frac{4}{3} \eta_2 \right) \Pi \\ \frac{\partial W_j}{\partial \Xi} &= -2 V_j \eta_2 \end{aligned}$$

e

$$\begin{aligned} \frac{\partial \Pi}{\partial \dot{T}_{kl}} &= \begin{cases} 1 & \text{se } k = l \\ 0 & \text{se } k \neq l \end{cases} \\ \frac{\partial \Xi}{\partial \dot{T}_{kl}} &= \begin{cases} \dot{T}_{11} + \dot{T}_{22} + \dot{T}_{33} - \dot{T}_{kl} & \text{se } k = l \\ -\dot{T}_{lk} & \text{se } k \neq l \end{cases} \end{aligned}$$

Pode-se verificar facilmente as seguintes igualdades:

$$\frac{\partial \dot{T}_{kl}}{\partial \dot{C}_{rs}} = \frac{\partial T_{kl}}{\partial C_{rs}}$$

$$\frac{\partial \dot{C}_{rs}}{\partial \dot{B}_{uv}} = \frac{\partial C_{rs}}{\partial B_{uv}}$$

$$\frac{\partial \dot{B}_{uv}}{\partial \dot{q}_i} = \frac{\partial B_{uv}}{\partial q_i}$$

que permitem reaproveitar boa parte das contas efetuadas no cálculo das forças de elasticidade.

6.3 Cálculo das forças de vínculo

6.3.1 Restrições simples

Vamos considerar nesta seção apenas o caso de um vértice restrito a se mover num plano paralelo a um dos planos xOy, yOz ou zOx, da maneira que foi implementado em nosso sistema de animação. A restrição que permite obter esse efeito é expressa como $q_i - c = 0$, onde q_i é uma coordenada cartesiana de um vértice, e c é o valor dessa coordenada no estado inicial. Supondo-se que $\dot{q}_i$ seja nula no estado inicial, esta restrição pode ser escrita como $\ddot{q}_i = 0$. Note que se restringirmos desta maneira as três coordenadas de um vértice, ele ficará fixo a uma posição no espaço; aliás, esta é de longe a restrição mais útil que conhecemos.

Tratamos cada restrição do tipo $q_i - c = 0$ através de alterações simples no sistema de equações (6.1). Na fase de pré-processamento, transformamos a matriz de massa $\mathbf{M}$ em uma matriz $\mathbf{M}^*$, atribuindo o valor 0 (zero) a todos os elementos da i-ésima linha e da i-ésima coluna de $\mathbf{M}$, exceto ao elemento M_{ii}, ao qual atribuímos o valor 1 (um). Durante a simulação, atribuímos um 0 (zero) ao i-ésimo elemento do vetor de forças $\mathbf{f}$ e então calculamos as forças de vínculo $\mathbf{f}^{\mathrm{v}} = -\mathbf{D}^{\top}\boldsymbol{\lambda}$, utilizando a matriz $\mathbf{M}^*$. Feito isto, calculamos as acelerações $\ddot{\mathbf{q}}$ resolvendo o sistema $\mathbf{M}^*\ddot{\mathbf{q}} = \mathbf{f}^*$, onde $\mathbf{f}^*$ é a soma dos vetores $\mathbf{f}$ e $\mathbf{f}^{\mathrm{v}}$, com o seu i-ésimo elemento zerado.

Pode-se mostrar que as alterações em $\mathbf{M}$ preservam a sua simetria, bem como o fato dela ser positivo-definida. A restrição mais geral de um vértice preso a um plano ou a uma reta arbitrários poderia ser tratada de forma similar [Pis84].

6.3.2 Restrições holonômicas gerais

Para restrições gerais, precisamos resolver o sistema linear (6.1). Embora esse sistema possa ser resolvido simultaneamente nas acelerações $\ddot{\mathbf{q}}$ e nos multiplicadores de Lagrange $\boldsymbol{\lambda}$, é mais eficiente resolvê-lo separadamente em $\ddot{\mathbf{q}}$ e em $\boldsymbol{\lambda}$, devido à disponibilidade de uma fatoração pré-calculada de $\mathbf{M}$.

Para tanto, reescrevemos (6.1) na forma

$$\begin{cases} \mathbf{J}\mathbf{M}^{-1}\mathbf{D}^\top\boldsymbol{\lambda} & = & \mathbf{J}\mathbf{M}^{-1}\mathbf{f} - \boldsymbol{\psi} \\ \mathbf{M}\ddot{\mathbf{q}} & = & \mathbf{f} - \mathbf{D}^\top\boldsymbol{\lambda} \end{cases}$$

de onde calculamos primeiro $\boldsymbol{\lambda}$ e depois $\ddot{\mathbf{q}}$.

Em nossa implementação, todos os k vínculos no sistema são do tipo $H_l(\mathbf{q}, t) = 0$. A matriz $\mathbf{J}$, para cada novo estado calculado pelo integrador, é inicialmente preenchida com zeros. Em seguida, para cada equação de vínculo $H_l(\mathbf{q}, t) = 0$, para $l = 1, \ldots, k$, atribui-se a J_{li} o valor de $\frac{\partial H_l}{\partial q_i}(\mathbf{q}, t)$, para as coordenadas q_i presentes na função H_l.

6.3.3 Resolução do sistema linear

Dado o sistema linear $\mathbf{J}\mathbf{M}^{-1}\mathbf{D}^\top\boldsymbol{\lambda} = \mathbf{J}\mathbf{M}^{-1}\mathbf{f} - \boldsymbol{\psi}$, utilizamos a seguinte seqüência de passos para resolvê-lo em $\boldsymbol{\lambda}$:

1. Resolver $\mathbf{M}\mathbf{E} = \mathbf{J}^\top$ em relação a $\mathbf{E}$.
2. Calcular $\mathbf{b} \leftarrow \mathbf{E}^\top\mathbf{f} - \boldsymbol{\psi}$
3. Calcular $\mathbf{A} \leftarrow \mathbf{E}^\top\mathbf{D}^\top$
4. Resolver $\mathbf{A}\boldsymbol{\lambda} = \mathbf{b}$ em relação a $\boldsymbol{\lambda}$.

O sistema linear no passo 1 é resolvido aplicando-se os passos (6.2) para cada um dos k pares de colunas das matrizes $\mathbf{J}^\top$ e $\mathbf{E}$. O sistema linear no passo 4 é resolvido por um método de eliminação de Gauss com pivoteamento parcial. Uma vez obtido $\boldsymbol{\lambda}$, calculamos as forças de vínculo $-\mathbf{D}^\top\boldsymbol{\lambda}$ e as somamos ao vetor $\mathbf{f}$.

Não implementamos forças de vínculo com atrito, de modo que a direção $\mathbf{d}_l$ de cada força de vínculo é dada por

$$\mathbf{d}_l = \frac{\partial H_l}{\partial \mathbf{q}}(\mathbf{q}, t)$$

para $l = 1, \ldots, k$. Portanto, a matriz $\mathbf{D}$ é igual à matriz $\mathbf{J}$, e as forças de vínculo são $-\mathbf{J}^\top\boldsymbol{\lambda}$.

Como a matriz $\mathbf{M}^{-1}$ é simétrica e positivo-definida (pois $\mathbf{M}$ assim é), sabemos que a matriz $\mathbf{J}\mathbf{M}^{-1}\mathbf{J}^\top$, além de simétrica, é *positivo-semi-definida*, ou seja, $\boldsymbol{\lambda}^\top\mathbf{J}\mathbf{M}^{-1}\mathbf{J}^\top\boldsymbol{\lambda} \geq 0$, para qualquer vetor $\boldsymbol{\lambda}$. O passo 4 é então resolvido de maneira mais eficiente por uma eliminação de Gauss sem pivoteamento. Note porém que, quando os vetores $\mathbf{d}_l$ têm outras direções, a matriz $\mathbf{J}\mathbf{M}^{-1}\mathbf{D}^\top$ não é simétrica e nem positivo-semi-definida, em geral.

6.3.4 Situações não determinadas

Sejam $\mathbf{A} = \mathbf{J}\mathbf{M}^{-1}\mathbf{D}^{\top}$ e $\mathbf{b} = \mathbf{J}\mathbf{M}^{-1}\mathbf{f} - \boldsymbol{\psi}$. O sistema linear $\mathbf{A}\boldsymbol{\lambda} = \mathbf{b}$ tem uma única solução se e somente se a matriz $\mathbf{A}$ é não-singular. Caso contrário, o vetor $\mathbf{b}$ determina se não há nenhuma solução ou se há infinitas soluções.

Para ver como essas situações podem ocorrer, tomemos um exemplo. Suponha que uma partícula u de posição corrente $\mathbf{p}$ é restrita a permanecer na superfície de duas esferas de raio unitário, a primeira com centro em $[-1, 0, 0]$ e a segunda em $[1, 0, 0]$, e que no estado inicial do sistema essa partícula está na origem, e portanto no ponto único de interseção das duas esferas. A matriz $\mathbf{A}$ nesta situação é singular. As duas forças de vínculo (em $\mathbb{R}^3$) sobre a partícula, enquanto ela permanecer na posição inicial, terão direções dadas pelos vetores $\mathbf{d}_1 = [1, 0, 0]$ e $\mathbf{d}_2 = [-1, 0, 0]$. Se a velocidade inicial da partícula é nula e uma força externa $\mathbf{f}_u = [\gamma, 0, 0]$ age sobre ela, com $\gamma \neq 0$, então qualquer combinação $\lambda_1\mathbf{d}_1 + \lambda_2\mathbf{d}_2$ das duas forças de vínculo, tal que $\lambda_1 - \lambda_2 = -\gamma$, será capaz de manter satisfeitos os dois vínculos. Porém, se a velocidade inicial da partícula tem uma componente tangencial às duas esferas, ou se uma força externa com uma componente tangencial age sobre ela, não há nenhuma combinação de forças de vínculo que impeça que ao menos um dos dois vínculos se torne não satisfeito num instante imediatamente posterior.

Para que $\mathbf{A}$ seja não-singular, é necessário que as forças de vínculo, ou alternativamente as linhas das matrizes $\mathbf{J}$ e $\mathbf{D}$, sejam linearmente independentes — em outros termos, que o posto de $\mathbf{J}$ e de $\mathbf{D}$ seja igual ao número de vínculos k. Em nosso sistema, nós simplesmente paramos a simulação se porventura a matriz $\mathbf{A}$ torna-se singular. Felizmente, devido ao fato de lidarmos com corpos elásticos, essa situação só ocorre quando os vínculos são claramente conflitantes.

6.3.5 Tipos de restrições implementadas

Sejam $\mathbf{p}_i = [x_i, y_i, z_i]$, para $i = 1, \ldots, 4$, as posições correntes dos vértices de um tetraedro T, dados no sentido convencionado. Seja $\mathbf{p} = [x, y, z]$ a posição corrente de um vértice do modelo. A nossa implementação permite especificar os tipos de restrições de movimento descritos a seguir.

Vértice restrito a um plano: Para restringir um vértice a um plano fixo utilizamos o vínculo

$$a\,x + b\,y + c\,z + d = 0$$

onde a, b, c e d são os coeficientes do plano.

Vértice restrito a uma superfície quádrica: Para restringir um vértice a se mover sobre uma superfície quádrica genérica, basta utilizar a equação da quádrica desejada sobre as coordenadas do vértice, ou seja,

$$a\,x^2 + b\,y^2 + c\,z^2 + d\,x\,y + e\,x\,z + f\,y\,z + g\,x + h\,y + i\,z + j = 0$$

onde $a, \ldots, j$ são os coeficientes da superfície quádrica. Por exemplo, para restringir um vértice a uma esfera de centro $\mathbf{c} = [c_x, c_y, c_z]$ e raio r utilizamos o vínculo

$$(x - c_x)^2 + (y - c_y)^2 + (z - c_z)^2 - r^2 = 0$$

Vértice preso a um ponto na superfície de um corpo: Sejam $\mathbf{p}_1$, $\mathbf{p}_2$ e $\mathbf{p}_3$ as posições correntes dos vértices de uma face exposta F, que pertence a um tetraedro diferente do tetraedro que contém o vértice de posição $\mathbf{p}$. Seja $\mathbf{q} = \alpha_1\mathbf{p}_1 + \alpha_2\mathbf{p}_2 + (1 - \alpha_1 - \alpha_2)\mathbf{p}_3$ a posição de uma partícula nessa face, onde α_1 e α_2 são coordenadas baricêntricas válidas dessa partícula em relação a F. Restringimos as partículas em $\mathbf{p}$ e $\mathbf{q}$ a terem a mesma posição através de três vínculos:

$$\left\{\begin{array}{rcl} x - \alpha_1 x_1 - \alpha_2 x_2 - (1 - \alpha_1 - \alpha_2) x_3 & = & 0 \\ y - \alpha_1 y_1 - \alpha_2 y_2 - (1 - \alpha_1 - \alpha_2) y_3 & = & 0 \\ z - \alpha_1 z_1 - \alpha_2 z_2 - (1 - \alpha_1 - \alpha_2) z_3 & = & 0 \end{array}\right.$$

Note que esta restrição permite prender um vértice a outro, um vértice a uma aresta, e um vértice a uma face.

Tetraedro incompressível: Um tetraedro cujo volume não pode variar é obtido através do vínculo

$$\frac{1}{6}\begin{vmatrix} x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\ x_3 - x_1 & y_3 - y_1 & z_3 - z_1 \\ x_4 - x_1 & y_4 - y_1 & z_4 - z_1 \end{vmatrix} - V_0 = 0$$

onde V_0 é o volume de repouso do tetraedro.

Tetraedro rígido: Num tetraedro rígido, a distância entre cada par de seus quatro vértices não varia. Portanto, são necessários os seis vínculos

$$\left\{\begin{array}{rcl} |\mathbf{p}_2 - \mathbf{p}_1|^2 - l_1^2 & = & 0 \\ |\mathbf{p}_3 - \mathbf{p}_1|^2 - l_2^2 & = & 0 \\ |\mathbf{p}_4 - \mathbf{p}_1|^2 - l_3^2 & = & 0 \\ |\mathbf{p}_3 - \mathbf{p}_2|^2 - l_4^2 & = & 0 \\ |\mathbf{p}_4 - \mathbf{p}_2|^2 - l_5^2 & = & 0 \\ |\mathbf{p}_4 - \mathbf{p}_3|^2 - l_6^2 & = & 0 \end{array}\right.$$

onde $l_1, \ldots, l_6$ são as distâncias entre os vértices correspondentes do tetraedro na configuração de repouso.

Restrição cinemática com trajetória de Hermite: Esta restrição permite ao animador especificar que um vértice sai de um ponto $\mathbf{u} = [u_x, u_y, u_z]$ no instante t_a, com velocidade inicial $\dot{\mathbf{u}} = [\dot{u}_x, \dot{u}_y, \dot{u}_z]$, e chega no ponto $\mathbf{v} = [v_x, v_y, v_z]$ com velocidade final $\dot{\mathbf{v}} = [\dot{v}_x, \dot{v}_y, \dot{v}_z]$ no instante t_b. Para obter este comportamento, utilizamos polinômios de Hermite [FvDFH90] para definir a trajetória do vértice (veja a figura (6.1)):

$$\begin{cases} x(t) &= a_x(t-t_a)^3 + b_x(t-t_a)^2 + c_x(t-t_a) + d_x \\ y(t) &= a_y(t-t_a)^3 + b_y(t-t_a)^2 + c_y(t-t_a) + d_y \\ z(t) &= a_z(t-t_a)^3 + b_z(t-t_a)^2 + c_z(t-t_a) + d_z \end{cases}$$

Esta curva interpola os pontos extremos $\mathbf{u}$ e $\mathbf{v}$, e suas tangentes em t_a e t_b igualam as derivadas $\dot{\mathbf{u}}$ e $\dot{\mathbf{v}}$. Os coeficientes do polinômio de x são dados abaixo:

$$\begin{cases} a_x &= \dfrac{2(u_x - v_x)}{(t_b - t_a)^3} + \dfrac{\dot{u}_x + \dot{v}_x}{(t_b - t_a)^2} \\ b_x &= \dfrac{3(v_x - u_x)}{(t_b - t_a)^2} - \dfrac{2\dot{u}_x + \dot{v}_x}{t_b - t_a} \\ c_x &= \dot{u}_x \\ d_x &= u_x \end{cases}$$

Os coeficientes para y e z são análogos. Finalmente, temos os três seguintes vínculos:

$$\begin{cases} x - x(t) &= 0 \\ y - y(t) &= 0 \\ z - z(t) &= 0 \end{cases}$$

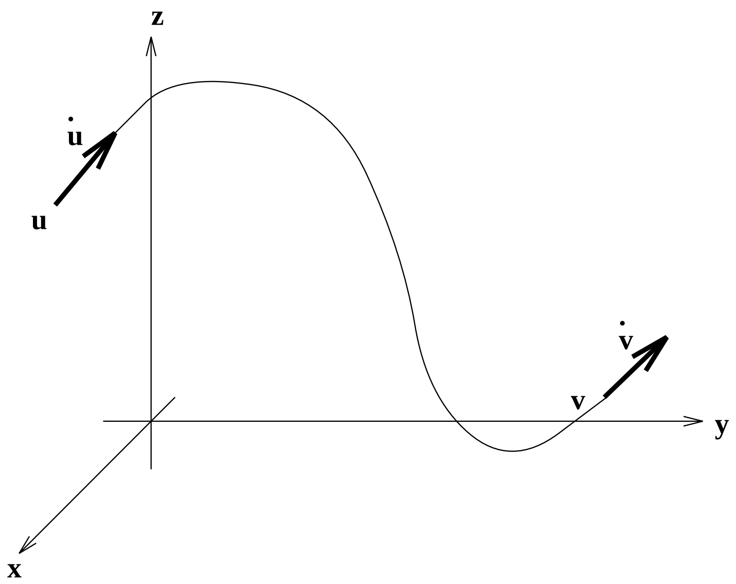


Figura 6.1: Trajetória definida por uma restrição cinemática

6.4 Integração do sistema de equações

As trajetórias "reais" (ou exatas) das partículas em um dado sistema dinâmico são aquelas que satisfazem exatamente as equações diferenciais de movimento do sistema. Uma aproximação para tais trajetórias pode ser calculada através de um método numérico. A escolha de um método específico impõe um limite superior ao grau de precisão dessa aproximação, e um limite inferior ao custo computacional incorrido em sua obtenção.

Para propósitos de animação, as posições e velocidades dos objetos não precisam ser calculadas com precisão muito alta, pois o efeito visual de tais imprecisões é praticamente imperceptível. Precisamos entretanto de um método razoavelmente preciso devido à grande complexidade do modelo, que inclui vários tipos de forças, com componentes não lineares. Com métodos pouco precisos, a trajetória do sistema no espaço de configurações provavelmente logo se afastaria permanentemente da trajetória exata, não conseguindo manter o valor da energia mecânica do sistema e/ou o desvio das restrições de movimento dentro de tolerâncias numéricas aceitáveis.

Em linhas gerais, um algoritmo de integração numérica começa com um estado inicial em um instante arbitrário t_0, e calcula uma seqüência de estados em instantes sucessivos $t_0 < t_1 < t_2 < \ldots < t_m$, para algum m. Um *passo* da integração consiste em calcular um novo estado a partir de um ou mais estados imediatamente anteriores. Para fins de animação, é necessário que esta seqüência de instantes contenha os instantes dos quadros da animação, que são estados separados por intervalos fixos de tempo.

6.4.1 Integração numérica adaptativa

Os algoritmos de integração mais eficientes são *adaptativos*: eles ajustam automaticamente o tamanho τ do intervalo de tempo entre dois estados consecutivos, de modo a manter uma estimativa da diferença entre as trajetórias exata e aproximada abaixo de uma dada tolerância. Esse ajuste é feito de acordo com alguma medida local da "dificuldade" de integração no passo corrente. Quando as forças variam suavemente, o algoritmo pode utilizar um intervalo de tempo maior. Por outro lado, quando as forças variam rapidamente, ele deve adotar um intervalo menor. Quando τ tende a zero, a diferença estimada entre as trajetórias exata e aproximada também tende a zero. Entretanto, quanto menor o tamanho desse intervalo, maior será o efeito cumulativo dos erros de arredondamento, que crescem exponencialmente no número de passos.

Na prática, o tamanho τ do passo deve ser restrito a uma faixa $[\tau_{\min}, \tau_{\max}]$. O limite inferior $\tau_{\min}$ é necessário para evitar que o algoritmo de integração tenha de fazer um número excessivo de passos entre dois quadros da animação, e para reduzir a chance de que os erros de arredondamento se tornem significativos. O limite superior $\tau_{\max}$ evidentemente deve ser menor ou igual ao intervalo entre quadros da animação; mas pode ser ainda menor caso as funções sendo integradas tenham irregularidades que possam ser não detetáveis nessa escala.

6.4.2 Eventos discretos

O passo de integração numérica que leva o estado do sistema de um instante t_i para um instante posterior $t_i + \tau$ é baseado em duas suposições: em primeiro lugar, as equações de movimento não mudam nesse passo ou em qualquer outro; em segundo, as derivadas temporais de primeira e segunda ordem das coordenadas são funções contínuas no intervalo $(t_i, t_i + \tau)$, embora não precisem ser contínuas nos instantes t_i e $t_i + \tau$. Entretanto, em certos sistemas dinâmicos, tais como os que descrevemos neste trabalho, é possível que uma dessas suposições, ou ambas, não sejam válidas em um dado passo de integração.

Como veremos, certos eventos especiais podem ocorrer em instantes arbitrários entre dois estados consecutivos. Um *evento discreto* é um acontecimento significativo para a simulação que ocorre em um instante t_e, além do qual a integração não pode continuar normalmente. Em termos gerais, é preciso integrar as equações de movimento até t_e, realizar alguma mudança específica nas equações de movimento e/ou no estado do sistema em t_e, e então continuar integrando a partir de t_e. No capítulo 7 veremos como detetar esses eventos, e como tratá-los de modo a continuar a integração.

6.4.3 Algoritmo de simulação

O algoritmo de simulação utilizado em nosso sistema integra a equação diferencial $\dot{\mathbf{u}} = F(\mathbf{u}, t)$, com $\mathbf{u} = (\mathbf{q}, \dot{\mathbf{q}})$, em um intervalo de tempo $I = [t_a, t_b]$, detetando e tratando quaisquer eventos discretos encontrados nesse intervalo.

Algoritmo 1 (`Simula`) *Recebe função F, instante inicial t_a, instante final t_b, estado inicial $\mathbf{u}_a = (\mathbf{q}_a, \dot{\mathbf{q}}_a)$, tolerância de erro ϵ, incremento máximo de tempo $\tau_{\max}$ e incremento mínimo de tempo $\tau_{\min}$.*

$t \leftarrow t_a$
$\mathbf{u} \leftarrow \mathbf{u}_a$
$\tau \leftarrow \tau_{\min}$
enquanto $t < t_b$
    $(\mathbf{u}', E) \leftarrow$ `Integra`$(\mathbf{u}, t, \tau, F)$
    se $E > \epsilon$ **e** $\tau > \tau_{\min}$ **então** (1)
        $\tau \leftarrow$ `NovoIncremento`$(E, \epsilon, \tau_{\min}, \tau_{\max})$
    senão
        $t_e \leftarrow$ `DetetaEvento`$(\mathbf{u}, \mathbf{u}', t, t + \tau)$
        se $t_e =$ `NenhumEvento` **então**

    $t \leftarrow t + \tau$
    $\mathbf{u} \leftarrow \mathbf{u}'$
    $\tau \leftarrow$ `NovoIncremento`$(E, \epsilon, \tau_{\min}, \tau_{\max})$
senão
    se $\tau > \tau_{\min}$ **então** (2)
        $\tau_{\mathrm{e}} \leftarrow t_e - t$
        $h \leftarrow 0.001 * \mathbf{min}(\tau_{\mathrm{e}}, \tau - \tau_{\mathrm{e}})$ (3)
        $\tau \leftarrow \mathbf{max}(\tau_{\min}, \mathbf{min}(\tau - \tau_{\min}, \tau_{\mathrm{e}} - h))$ (4)
    senão
        $t_e \leftarrow \mathbf{min}(t + \tau, \mathbf{max}(t + \tau_{\min}/2, t_e))$ (5)
        $\tau \leftarrow t_e - t$
        $(\mathbf{u}', E) \leftarrow$ `Integra`$(\mathbf{u}, t, \tau, \mathrm{F})$
        $t \leftarrow t_e$
        $\mathbf{u} \leftarrow \mathbf{u}'$
        significante $\leftarrow$ `TrataEvento`$(\mathbf{u}', t_e)$
        se *significante* **então**
            $\tau \leftarrow \tau_{\min}$
        senão
            $\tau \leftarrow$ `NovoIncremento`$(E, \epsilon, \tau_{\min}, \tau_{\max})$

O procedimento (não iterativo) `Integra` calcula a partir do estado $\mathbf{u} = (\mathbf{q}, \dot{\mathbf{q}})$ no instante t um novo estado $\mathbf{u}' = (\mathbf{q}', \dot{\mathbf{q}}')$ para um instante $t' = t + \tau$, e uma estimativa E do erro numérico envolvido nesse cálculo. Quando esse erro é excessivo, o simulador (na linha (1) acima) repete a integração com um incremento de tempo menor, desde que τ ainda possa ser diminuído. Se τ tiver atingido o valor mínimo $\tau_{\min}$, o valor de E é desconsiderado, e a simulação avança para o estado $\mathbf{u}'$.

O procedimento `DetetaEvento` devolve o instante t_e do primeiro evento ocorrido no intervalo $[t, t+\tau]$, ou a constante `NenhumEvento`, se não houverem eventos nesse intervalo.

Quando o algoritmo chega à linha (2), ele sabe que pode haver um evento no intervalo $I = [t, t+\tau]$, num instante próximo a t_e. Neste ponto, entretanto, é preciso levar em conta que a precisão da estimativa t_e é inversamente proporcional ao tamanho τ do intervalo I, e à distância de t_e aos extremos desse intervalo (o valor h calculado na linha (3) mede a precisão de t_e segundo este último critério). Se o integrador simplesmente caminhasse até t_e, quando τ é grande ou quando t_e está próximo ao instante médio de I, verificaria muitas vezes que a posição prevista do evento mudou para algum instante um pouco anterior a t_e. Portanto, o passo de integração é repetido para um intervalo definitivamente menor que $t_e - t$, até que τ finalmente atinja o valor mínimo $\tau_{\min}$. O mínimo calculado na linha (4) serve para garantir a convergência de τ até $\tau_{\min}$ num número finito de passos.

Quando τ atinge $\tau_{\min}$, o algoritmo supõe que o instante t_e é suficientemente preciso, integra até esse instante e trata o evento. O ajuste de t_e feito na linha (5) visa garantir que o integrador avance pelo menos de $\tau_{\min}/2$, a fim de prevenir lentidão excessiva na presença de muitos eventos.

O valor de retorno do procedimento `TrataEvento` diz se o evento que foi tratado em t_e é ou não significativo para a simulação, isto é, se houve ou não uma mudança no sistema de equações ou uma alteração descontínua no estado do sistema. Excluindo-se a gravação de novos quadros, todos os eventos detetáveis em nosso sistema de animação (colisão, quebra de contato, etc.), são significativos neste sentido.

A função F fornecida ao simulador produz $\dot{\mathbf{u}} = (\dot{\mathbf{q}}', \ddot{\mathbf{q}}')$ a partir de $\mathbf{u} = (\mathbf{q}, \dot{\mathbf{q}})$, copiando $\dot{\mathbf{q}}$ para $\dot{\mathbf{q}}'$ e calculando $\ddot{\mathbf{q}}'$ através do procedimento `CalculaAcelerações`, que realiza o cálculo das acelerações $\ddot{\mathbf{q}}$ a partir das forças diretas $\mathbf{f}$ e de vínculo $\mathbf{f}^{\mathrm{v}}$, resolvendo um sistema linear $\mathbf{M}\ddot{\mathbf{q}} = \mathbf{f} + \mathbf{f}^{\mathrm{v}}$ da maneira que descrevemos na seção (6.1.2).

Note que os procedimentos `CalculaAcelerações`, `DetetaEvento` e `TrataEvento` dependem do problema específico de simulação, ao passo que os procedimentos `Integra` e `NovoIncremento` dependem apenas do método específico de integração.

6.4.4 Método de integração de Runge-Kutta-Fehlberg

Em nosso sistema de animação, utilizamos o método de integração numérica de Runge-Kutta-Fehlberg [BF89], que é uma variante adaptativa do conhecido método de Runge-Kutta de 4ª ordem. Os procedimentos `Integra` e `NovoIncremento` que implementam esse método são descritos abaixo.

Algoritmo 2 (`Integra`) *Recebe vetor* $\mathbf{u}$ *de variáveis de estado, instante* t *do estado, incremento de tempo* τ *e função* F *que fornece* $\dot{\mathbf{u}}$ *a partir de* $\mathbf{u}$ *e* t*. Devolve novo estado* $\mathbf{u}'$ *e estimativa de erro* E*.*

$\mathbf{v}_1 \leftarrow F(\mathbf{u}, t)$
$\mathbf{u}' \leftarrow \mathbf{u} + \tau(c_{12}\mathbf{v}_1)$
$\mathbf{v}_2 \leftarrow F(\mathbf{u}', t + s_2\tau)$
$\mathbf{u}' \leftarrow \mathbf{u} + \tau(c_{13}\mathbf{v}_1 + c_{23}\mathbf{v}_2)$
$\mathbf{v}_3 \leftarrow F(\mathbf{u}', t + s_3\tau)$
$\mathbf{u}' \leftarrow \mathbf{u} + \tau(c_{14}\mathbf{v}_1 + c_{24}\mathbf{v}_2 + c_{34}\mathbf{v}_3)$
$\mathbf{v}_4 \leftarrow F(\mathbf{u}', t + s_4\tau)$
$\mathbf{u}' \leftarrow \mathbf{u} + \tau(c_{15}\mathbf{v}_1 + c_{25}\mathbf{v}_2 + c_{35}\mathbf{v}_3 + c_{45}\mathbf{v}_4)$
$\mathbf{v}_5 \leftarrow F(\mathbf{u}', t + s_5\tau)$
$\mathbf{u}' \leftarrow \mathbf{u} + \tau(c_{16}\mathbf{v}_1 + c_{26}\mathbf{v}_2 + c_{36}\mathbf{v}_3 + c_{46}\mathbf{v}_4 + c_{56}\mathbf{v}_5)$
$\mathbf{v}_6 \leftarrow F(\mathbf{u}', t + s_6\tau)$
$\mathbf{u}' \leftarrow \mathbf{u} + \tau(d_1\mathbf{v}_1 + d_3\mathbf{v}_3 + d_4\mathbf{v}_4 + d_5\mathbf{v}_5)$

$\mathbf{v} \leftarrow e_1\mathbf{v}_1 + e_3\mathbf{v}_3 + e_4\mathbf{v}_4 + e_5\mathbf{v}_5 + e_6\mathbf{v}_6$
$E \leftarrow \mathbf{max}|v_i|$
retorna $(\mathbf{u}', E)$

Algoritmo 3 (`NovoIncremento`) *Recebe estimativa de erro E, tolerância ϵ, incremento mínimo $\tau_{\min}$ e incremento máximo $\tau_{\max}$. Devolve novo incremento de tempo τ.*

$\delta \leftarrow \sqrt[4]{2\frac{\epsilon}{E}}$
$\tau' \leftarrow \delta\,\tau$
$\tau \leftarrow \mathbf{max}(\mathbf{min}(\tau', \tau_{\max}), \tau_{\min})$
retorna τ

Os valores das constantes utilizadas no algoritmo de integração são dadas na tabela abaixo. Note que as constantes s_j, para $j = 2, \ldots, 6$, são dadas por $s_j = \sum_{i=1}^{j-1} c_{ij}$.

c_{12} 1/4					
c_{13} 3/32	c_{23} 9/32				
c_{14} 1932/2197	c_{24} -7200/2197	c_{34} 7296/2197			
c_{15} 439/216	c_{25} -8	c_{35} 3680/513	c_{45} -845/4104		
c_{16} -8/27	c_{26} 2	c_{36} -3544/2565	c_{46} 1859/4104	c_{56} -11/40	
d_1 25/216		d_3 -1408/2565	d_4 2197/4104	d_5 -1/5	
e_1 1/360		e_3 -128/4275	e_4 -2197/75240	e_5 1/50	e_6 2/55
	s_2 1/4	s_3 3/8	s_4 12/13	s_5 1	s_6 1/2

Tabela 6.1: Valores das constantes do algoritmo de integração

Capítulo 7

Eventos discretos

Um *evento discreto* é um acontecimento definido em um instante arbitrário ao longo da evolução do sistema, e que requer um tratamento específico. No intervalo de tempo entre dois estados consecutivos, a evolução do sistema prossegue normalmente, ao longo da trajetória determinada pelas equações diferenciais de movimento. Para certos eventos, o tratamento consiste em uma mudança instantânea no estado do sistema e/ou uma mudança naquelas equações diferenciais. Para outros, ele consiste em uma operação arbitrária, tal como a gravação de um novo quadro da animação, que não afeta o estado do sistema ou as equações de movimento. Alguns eventos podem ocorrer em instantes predeterminados da simulação (a gravação de quadros, por exemplo), enquanto que outros são gerados internamente quando o estado do sistema ou as forças de vínculo atingem certos valores (por exemplo, quando dois corpos colidem).

7.1 Caracterização de eventos discretos

O *estado estendido* de um sistema dinâmico, em um instante t, é o conjunto dos valores das suas coordenadas generalizadas $\mathbf{q}$, velocidades generalizadas $\dot{\mathbf{q}}$, e multiplicadores de Lagrange $\boldsymbol{\lambda}$, em t.

7.1.1 Regimes de integração

Um *regime* é um conjunto $\mathcal{R}$ de estados estendidos $(\mathbf{q}, \dot{\mathbf{q}}, \boldsymbol{\lambda})$ no qual a evolução do sistema é contínua e determinada por um único sistema de equações diferenciais. O conjunto de *estados válidos* do sistema é a união de todos os seus regimes.

Vamos considerar apenas regimes que podem ser definidos por um predicado B de m argumentos binários $b_1, \ldots, b_m$, cujos valores ($\mathbf{0}$ para falso e $\mathbf{1}$ para verdadeiro) são dados por

$$b_i = \begin{cases} \mathbf{1} & \text{se} \quad G_i(\mathbf{q}, \dot{\mathbf{q}}, \boldsymbol{\lambda}, t) > 0 \\ \mathbf{0} & \text{se} \quad G_i(\mathbf{q}, \dot{\mathbf{q}}, \boldsymbol{\lambda}, t) < 0 \\ \text{indefinido} & \text{se} \quad G_i(\mathbf{q}, \dot{\mathbf{q}}, \boldsymbol{\lambda}, t) = 0 \end{cases} \tag{7.1}$$

para $i = 1, \ldots, m$, onde G_i é uma função escalar contínua do estado estendido e do tempo. Dizemos que um estado estendido é válido se e somente se $B = \mathbf{1}$.

Por exemplo, para um sistema de uma partícula de posição $\mathbf{p} = [x, y, z]$ que é restrita a permanecer entre dois planos paralelos $z = 0$ e $z - 10 = 0$, ou seja, $(z > 0) \wedge (z < 10)$, o conjunto de estados válidos é constituído de um único regime $\mathcal{R}$, definido pelo predicado $B = b_1 \wedge b_2$, com $G_1(x, y, z) = z$ e $G_2(x, y, z) = -z + 10$.

Note que o vetor $\boldsymbol{\lambda}$ é determinado pelo estado $(\mathbf{q}, \dot{\mathbf{q}})$, pelo tempo t e pelas forças externas $\mathbf{f}^{\mathrm{e}}$. Portanto, poderíamos em princípio substituir $\boldsymbol{\lambda}$ por $\mathbf{f}^{\mathrm{e}}$ nas funções G_i. Entretanto, na prática certas funções G_i provêm de processos físicos que dependem diretamente das forças de vínculo $\mathbf{f}^{\mathrm{v}}$ (e portanto do vetor $\boldsymbol{\lambda}$), e não das forças externas $\mathbf{f}^{\mathrm{e}}$. Contudo, note que se desejássemos definir eventos diretamente relacionados às forças externas, teríamos que incluir o vetor $\mathbf{f}^{\mathrm{e}}$ explicitamente nas funções G_i. Em nosso modelo, optamos por deixar de fora tais eventos.

Para uma dada trajetória do sistema no espaço de estados válidos, definimos a *folga* de cada condição $G_i(\mathbf{q}, \dot{\mathbf{q}}, \boldsymbol{\lambda}, t) > 0$ como uma função do tempo g_i tal que

$$g_i(t) = G_i(\mathbf{q}(t), \dot{\mathbf{q}}(t), \boldsymbol{\lambda}(t), t)$$

para todo instante t ao longo dessa trajetória. Se G_i é razoavelmente bem comportada e $g_i(t)$ suficientemente pequeno, este valor é uma medida da menor distância entre o estado estendido do sistema no instante t e um estado estendido no qual G_i é nula.

7.1.2 Deteção de eventos discretos

Enquanto o estado do sistema se encontra no interior do regime corrente $\mathcal{R}$, o predicado B vale $\mathbf{1}$. O próximo evento discreto ocorre quando a trajetória do sistema atinge a fronteira de $\mathcal{R}$, isto é, B deixa de ser $\mathbf{1}$. Nesse instante, um dos argumentos b_i deve mudar de valor, passando de $\mathbf{0}$ ou $\mathbf{1}$ para indefinido; ou seja, uma função de folga g_i deve se tornar nula. Note que nem toda transição em algum b_i implica uma transição em B; por exemplo, se $B = b_i \vee b_j$ e $b_j = \mathbf{1}$, o valor de b_i é irrelevante para B, enquanto b_j permanecer inalterado.

Dado um estado válido no instante t_a e um estado possivelmente não válido no instante $t_b > t_a$, queremos determinar se existe um instante no intervalo de tempo $I = (t_a, t_b]$ no qual o estado estendido do sistema deixa de ser válido; e, em caso afirmativo, queremos determinar o primeiro instante t_e no qual esse evento ocorre. Note que não basta verificar se o estado estendido em t_b é válido: precisamos também verificar se não existe nenhum estado estendido não válido em um instante intermediário a t_a e t_b.

7.1.3 Eventos simultâneos

A rigor, é possível que duas folgas g_i e g_j se anulem no mesmo instante. Na prática, porém, devido à falta de exatidão no cálculo dos estados estendidos através de métodos numéricos, não é possível determinar corretamente qual desses dois eventos ocorre primeiro ou se são realmente simultâneos, e muito menos tratá-los de maneira correta, mesmo que os ordenássemos segundo algum critério.

Felizmente, para muitos sistemas de interesse prático, a ocorrência de eventos simultâneos é extremamente improvável, de modo que podemos ignorar essa possibilidade. Portanto, em nosso modelo vamos supor que há exatamente uma função de folga g_i que se torna nula no instante de ocorrência de um evento discreto. Em particular, vamos ignorar a possibilidade de ocorrerem colisões simultâneas entre dois pares de pontos de superfície.

7.1.4 Folgas irrestritas

Para estimar o instante em que ocorre o primeiro evento discreto num dado intervalo $I = (t_a, t_b]$, adotamos uma aproximação polinomial para cada função de folga g_i, nesse intervalo. Na prática, é necessário utilizar polinômios de graus diferentes para cada função g_i, dependendo de quanta informação temos sobre suas derivadas.

Dizemos que uma folga g_i é *irrestrita* se a função G_i depende diretamente de $\dot{\mathbf{q}}$ ou de $\boldsymbol{\lambda}$. Em geral, as acelerações generalizadas $\ddot{\mathbf{q}}$ no instante t_b só podem ser determinadas com segurança se a trajetória do sistema de t_a até t_b for conhecida. Além disso, as derivadas temporais dos multiplicadores $\boldsymbol{\lambda}$ não podem ser determinadas, em geral, a menos que as derivadas temporais das forças externas possam ser calculadas. Em conseqüência destas limitações, a derivada primeira de g_i não pode ser calculada quando G_i depende de $\dot{\mathbf{q}}$ ou de $\boldsymbol{\lambda}$. Portanto, precisamos adotar uma aproximação linear para funções de folga irrestritas.

Por hipótese, g_i é positiva em t_a. Supondo-se que g_i é linear em I, ela se torna nula neste intervalo somente se $g_i(t_b)$ é negativo ou nulo. Neste caso, calculamos o instante aproximado t_e em que g_i se anula como

$$t_e = t_a - g_i(t_a)\,\frac{t_b - t_a}{g_i(t_b) - g_i(t_a)} \tag{7.2}$$

7.1.5 Folgas geométricas

Dizemos que uma folga g_i é *geométrica* se a função G_i não depende de $\dot{\mathbf{q}}$ nem de $\boldsymbol{\lambda}$, mas apenas de $\mathbf{q}$ e/ou t. Devido à disponibilidade das velocidades generalizadas $\dot{\mathbf{q}}$ em t_a e t_b, podemos calcular a derivada $\dot{g}_i$ nesses dois instantes, como

$$\dot{g}_i(t) = \frac{\partial G_i}{\partial \mathbf{q}}(\mathbf{q}(t), t) \cdot \dot{\mathbf{q}}(t) + \frac{\partial G_i}{\partial t}(\mathbf{q}(t), t)$$

para $t = t_a$ e $t = t_b$. Portanto, podemos adotar uma aproximação polinomial de grau superior a 1 para funções de folga geométricas.

Especificamente, aproximamos tal função g_i por um *polinômio interpolador de Hermite* [FvDFH90] P sobre t, de terceiro grau e definido no intervalo I, que interpola g_i em t_a e t_b, e cuja derivada primeira iguala $\dot{g}_i$ nesses dois instantes.

Seja $u = (t - t_a)/(t_b - t_a)$. O polinômio de Hermite pode ser escrito na forma de Bernstein-Bézier $P(u) = P_1 \, B_1(u) + P_2 \, B_2(u) + P_3 \, B_3(u) + P_4 \, B_4(u)$, onde

$$\begin{aligned} B_1(u) &= (1-u)^3 \\ B_2(u) &= 3u(1-u)^2 \\ B_3(u) &= 3u^2(1-u) \\ B_4(u) &= u^3 \end{aligned}$$

são as funções da base de Bernstein. Os coeficientes $P_1, \ldots, P_4$ são calculados como

$$\begin{aligned} P_1 &= g_i(t_a) \\ P_2 &= P_1 + \frac{t_b - t_a}{3} \, \dot{g}_i(t_a) \\ P_3 &= P_4 - \frac{t_b - t_a}{3} \, \dot{g}_i(t_b) \\ P_4 &= g_i(t_b) \end{aligned}$$

Esta representação tem a propriedade de que todo valor $P(u)$ para u no intervalo $U = [0, 1]$ é uma combinação linear convexa dos coeficientes $P_1, \ldots, P_4$. Assim, se estes quatro coeficientes são todos positivos ou todos negativos, pode-se concluir que todos os valores de P em U são positivos ou negativos, respectivamente, e portanto não há raiz neste intervalo.

Quando este teste falha, nós aplicamos um algoritmo de subdivisão binária [FvDFH90] sobre cada polinômio P para procurar as suas raízes em U. A cada passo, o intervalo U é dividido ao meio, sendo calculado um polinômio L na metade esquerda de U e um polinômio R na metade direita de U. Denotamos os coeficientes de L por L_1, L_2, L_3, L_4, e os de R por R_1, R_2, R_3, R_4. Estes coeficientes são calculados através do seguinte algoritmo:

$$\begin{aligned} L_1 &\leftarrow P_1 \\ L_2 &\leftarrow \frac{P_1 + P_2}{2} \\ M &\leftarrow \frac{P_2 + P_3}{2} \\ L_3 &\leftarrow \frac{L_2 + M}{2} \\ R_3 &\leftarrow \frac{P_3 + P_4}{2} \end{aligned}$$

$$\begin{aligned} R_2 &\leftarrow \frac{R_3 + M}{2} \\ L_4 &\leftarrow \frac{L_3 + R_2}{2} \\ R_1 &\leftarrow L_4 \\ R_4 &\leftarrow P_4 \end{aligned}$$

Em seguida, o algoritmo de busca é repetido recursivamente para cada metade do intervalo corrente, começando pela metade esquerda. Quando a condição

$$(|P_1 + P_3 - 2P_2| < \epsilon) \wedge (|P_2 + P_4 - 2P_3| < \epsilon)$$

se verifica para um dado ϵ pequeno, nós encerramos a busca recursiva, aproximando P no intervalo corrente por uma reta. Neste momento, se P_1 e P_4 têm sinais opostos, calculamos a raiz aproximada u_e através da equação

$$u_e = -\frac{P_1}{P_4 - P_1}$$

7.2 Deteção de colisões

Uma colisão ocorre quando dois pontos de superfície distintos entram em contato. No nosso modelo, a superfície de cada corpo consiste das *faces expostas* dos seus tetraedros, isto é, aquelas que não são compartilhadas por dois tetraedros diferentes. Os vértices e arestas das faces expostas também fazem parte da superfície do corpo. Classificamos cada elemento (vértice, aresta, ou face) de um tetraedro em um dado corpo como *externo* ou *de superfície*, se o mesmo pertence à superfície do corpo, ou como *interno*, em caso contrário.

Dentre todas as possíveis combinações em pares de elementos de superfície (vértice-vértice, vértice-face, aresta-face, etc.), vamos supor que apenas os pares vértice-face e aresta-aresta têm probabilidade não nula de entrar em colisão em um instante qualquer. Além disso, vamos supor também que colisões múltiplas simultâneas, ou seja, dois ou mais pares vértice-face ou aresta-aresta colidindo no mesmo instante, têm probabilidade nula de ocorrência. Estas suposições são justificadas pelo fato de que qualquer perturbação infinitesimal nos valores das coordenadas dos vértices ou nos cálculos numéricos, elimina as colisões indesejadas ou as substitui por uma única colisão vértice-face ou aresta-aresta, com probabilidade 1.

Assim, em uma dada colisão há exatamente quatro vértices $u_1, \ldots, u_4$ envolvidos, cujas coordenadas cartesianas na configuração corrente são $\mathbf{p}_i = [x_i, y_i, z_i]$, para $i = 1, 2, 3, 4$.

7.2.1 Colisão vértice-face

Uma colisão do vértice u_4 contra a face $\mathsf{F} = (u_1, u_2, u_3)$ ocorre em um dado intervalo $I = (t_a, t_b]$ se e somente se em algum instante $t_c \in I$ a trajetória de u_4 cruza um ponto interior de F, passando do lado externo para o lado interno de F.

Uma condição suficiente para que o vértice u_4 não colida com a face F no intervalo I é que ele permaneça do mesmo lado do plano π de F, ao longo de I. Seja $\mathbf{n}$ um vetor ortogonal ao plano π, dado por

$$\mathbf{n} = (\mathbf{p}_2 - \mathbf{p}_1) \times (\mathbf{p}_3 - \mathbf{p}_1)$$

Vamos supor a partir daqui, sem perda de generalidade, que a ordem dos vértices u_1, u_2, u_3 é tal que $\mathbf{n}$ aponta para o lado externo de F. Portanto, se a função G_1 dada por

$$G_1(\mathbf{q}) = \mathbf{n} \cdot (\mathbf{p}_4 - \mathbf{p}_1)$$

é tal que $G_1(\mathbf{q}) > 0 (b_1)$ ou $G_1(\mathbf{q}) < 0 (\neg b_1)$ para toda configuração $\mathbf{q}$ ao longo de I, então o par vértice-face (u_4, F) não colide nesse intervalo.

Se a folga g_1 associada à função G_1 se torna nula em um instante $t_c \in I$, é preciso verificar se nesse instante o vértice u_4 está no interior de F. Para tanto, vamos considerar os planos π_1, π_2 e π_3 perpendiculares ao plano π de F e que contêm as arestas $\mathsf{A}_1 = (u_2, u_3)$, $\mathsf{A}_2 = (u_1, u_3)$ e $\mathsf{A}_3 = (u_1, u_2)$, respectivamente. Se u_4 está no interior do prisma definido por estes três planos no instante t_c, então sabemos que esse vértice está no interior da face F nesse instante. (Outra forma de realizar este teste seria usar os planos das outras três faces do tetraedro que possui a face externa F.) Seja $\mathbf{n}_i$ um vetor ortogonal ao plano π_i, que aponta para o lado do vértice u_i, para $i = 1, 2, 3$. Veja a figura (7.1).

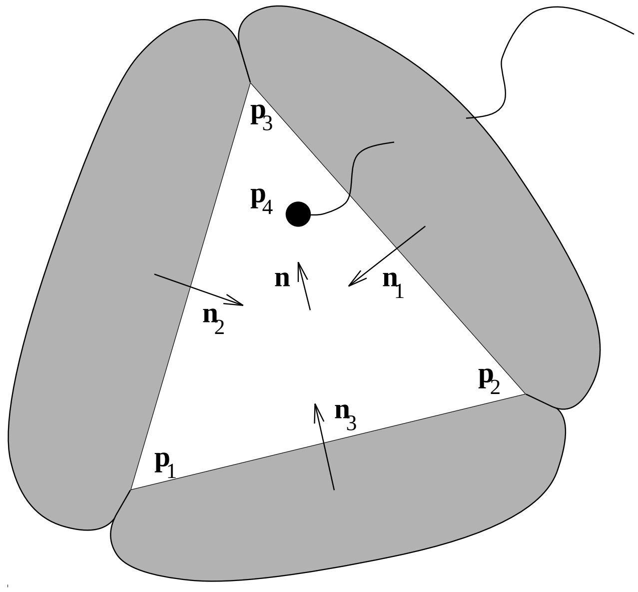


Figura 7.1: Modelo geométrico para deteção de uma colisão vértice-face

Calculamos estes três vetores como

$$\begin{aligned}\mathbf{n}_1 &= \mathbf{n} \times (\mathbf{p}_3 - \mathbf{p}_2)\\ \mathbf{n}_2 &= \mathbf{n} \times (\mathbf{p}_1 - \mathbf{p}_3)\\ \mathbf{n}_3 &= \mathbf{n} \times (\mathbf{p}_2 - \mathbf{p}_1)\end{aligned}$$

Assim, outras três condições suficientes para que o vértice u_4 não colida com a face F ao longo de I, são descritas pelas condições $G_2(\mathbf{q}) > 0$ (b_2), $G_3(\mathbf{q}) > 0$ (b_3) e $G_4(\mathbf{q}) > 0$ (b_4), onde as funções G_2, G_3 e G_4 são dadas por

$$\begin{aligned}G_2(\mathbf{q}) &= \mathbf{n}_1 \cdot (\mathbf{p}_2 - \mathbf{p}_4)\\ G_3(\mathbf{q}) &= \mathbf{n}_2 \cdot (\mathbf{p}_1 - \mathbf{p}_4)\\ G_4(\mathbf{q}) &= \mathbf{n}_3 \cdot (\mathbf{p}_1 - \mathbf{p}_4)\end{aligned}$$

para toda configuração $\mathbf{q}$ ao longo de I.

Em nosso modelo, cada corpo possui sempre uma superfície fechada. No instante t_a, nós supomos que cada vértice está no lado externo de cada uma dessas superfícies. Portanto, não precisamos nos preocupar com a possibilidade de um vértice cruzar uma superfície de dentro para fora, pois esse vértice deveria primeiro atravessar a superfície de fora para dentro.

Em conclusão, definimos o predicado B para cada par vértice-face como $B = b_1 \vee b_2 \vee b_3 \vee b_4$. Sejam $g_1, \ldots, g_4$ as folgas associadas às funções $G_1, \ldots, G_4$, respectivamente. Uma colisão entre o vértice u_4 e a face F ocorre em um instante t_c se e somente se

$$\begin{cases} g_1(t_c) = 0\\ g_2(t_c) < 0\\ g_3(t_c) < 0\\ g_4(t_c) < 0\end{cases}$$

Note que não estamos interessados nos instantes em que o vértice u_4 cruza o plano da face F em um ponto fora da mesma, isto é, b_1 se torna indefinido quando b_2, b_3 ou b_4 não são todos iguais a $\mathbf{0}$. Também não estamos interessados nos instantes em que b_2, b_3 e b_4 se tornam indefinidos.

7.2.2 Colisão aresta-aresta

Para um par aresta-aresta, uma colisão ocorre num instante t_c se e somente se a reta que contêm uma das arestas intercepta a reta que contém a outra num ponto interior a ambas as arestas.

Sejam $\mathsf{A}_1 = (u_1, u_2)$ e $\mathsf{A}_2 = (u_3, u_4)$ duas arestas que não possuem vértices em comum e pertencem a tetraedros diferentes. Suponha que as retas de A_1 e A_2 se interceptam num instante $t_c \in I$. Para verificar se a interseção ocorre no interior de cada aresta, utilizamos um plano π_1 que contém A_1 e um plano π_2 que contém A_2. As arestas A_1 e A_2 têm um ponto interior comum se e somente se A_1 cruza π_2 e A_2 cruza π_1. Para garantir que nenhum dos planos π_1 e π_2 passa pelos quatro vértices $u_1, \ldots, u_4$, vamos encontrar dois pontos adicionais, um para cada aresta, que nunca ficam coplanares a esses quatro vértices.

Sejam $\mathsf{T}_1 = (u_1, u_2, v_1, v_2)$ e $\mathsf{T}_2 = (u_3, u_4, v_3, v_4)$ dois tetraedros que contêm as arestas A_1 e A_2, respectivamente, onde v_1, v_2 e v_3, v_4 são os vértices complementares aos vértices de A_1 e A_2 nesses dois tetraedros. Denotamos as posições desses vértices por $\mathbf{p}'_1, \mathbf{p}'_2$ e $\mathbf{p}'_3, \mathbf{p}'_4$, respectivamente. Sejam

$$\begin{aligned} \mathbf{m}_1 &= \frac{\mathbf{p}'_1 + \mathbf{p}'_2}{2} \\ \mathbf{m}_2 &= \frac{\mathbf{p}'_3 + \mathbf{p}'_4}{2} \end{aligned}$$

Veja a figura (7.2). Os planos π_1 e π_2 são, portanto, os planos que passam por $\mathbf{p}_1, \mathbf{p}_2, \mathbf{m}_1$ e por $\mathbf{p}_3, \mathbf{p}_4, \mathbf{m}_2$, respectivamente.

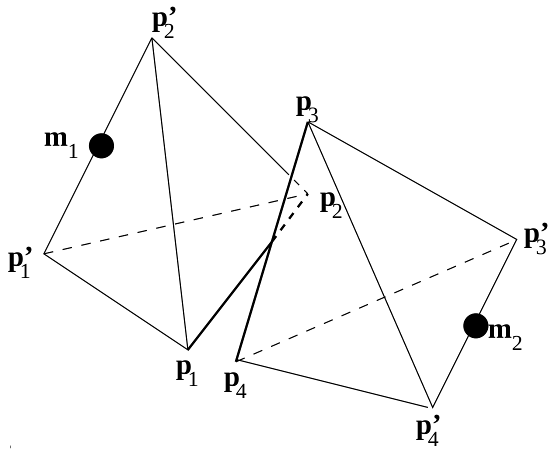


Figura 7.2: Modelo geométrico para deteção de uma colisão aresta-aresta

A reta que contém A_1 intercepta a que contém A_2 quando os vértices $u_1, \ldots, u_4$ se tornam coplanares. Assim, definimos a função G_1 como

$$G_1(\mathbf{q}) = \begin{vmatrix} 1 & x_1 & y_1 & z_1 \\ 1 & x_2 & y_2 & z_2 \\ 1 & x_3 & y_3 & z_3 \\ 1 & x_4 & y_4 & z_4 \end{vmatrix}$$

Sejam $\mathbf{n}_1$ e $\mathbf{n}_2$ vetores ortogonais aos planos π_1 e π_2, respectivamente:

$$\begin{aligned} \mathbf{n}_1 &= (\mathbf{p}_2 - \mathbf{p}_1) \times (\mathbf{m}_1 - \mathbf{p}_1) \\ \mathbf{n}_2 &= (\mathbf{p}_4 - \mathbf{p}_3) \times (\mathbf{m}_2 - \mathbf{p}_3) \end{aligned}$$

Escrevemos então

$$\begin{aligned} G_2(\mathbf{q}) &= \mathbf{n}_1 \cdot (\mathbf{p}_3 - \mathbf{p}_1) \\ G_3(\mathbf{q}) &= \mathbf{n}_1 \cdot (\mathbf{p}_4 - \mathbf{p}_1) \\ G_4(\mathbf{q}) &= \mathbf{n}_2 \cdot (\mathbf{p}_1 - \mathbf{p}_3) \\ G_5(\mathbf{q}) &= \mathbf{n}_2 \cdot (\mathbf{p}_2 - \mathbf{p}_3) \end{aligned}$$

Seja b_i a condição $G_i(\mathbf{q}) > 0$, para $i = 1, \ldots, 5$. Pode-se verificar que, partindo de um estado sem interpenetração de corpos, a região de estados válidos $\mathcal{R}$ para cada par aresta-aresta é descrita pelo predicado

$$\begin{aligned} B = {} & (b_2 \wedge b_3) \vee (\neg b_2 \wedge \neg b_3) \vee (b_4 \wedge b_5) \vee (\neg b_4 \wedge \neg b_5) \\ & \vee (b_1 \wedge b_2 \wedge \neg b_3 \wedge b_4 \wedge \neg b_5) \\ & \vee (\neg b_1 \wedge \neg b_2 \wedge b_3 \wedge \neg b_4 \wedge b_5) \\ & \vee (b_2 \wedge \neg b_3 \wedge \neg b_4 \wedge b_5) \\ & \vee (\neg b_2 \wedge b_3 \wedge b_4 \wedge \neg b_5) \end{aligned}$$

Sejam $g_1, \ldots, g_5$ as folgas associadas às funções $G_1, \ldots, G_5$, respectivamente. Uma colisão aresta-aresta ocorre em um instante t_c se e somente se exatamente um dos dois seguintes grupos de condições se verifica:

$$\left\{ \begin{array}{rcl} g_1(t_c) & = & 0 \\ g_2(t_c) & > & 0 \\ g_3(t_c) & < & 0 \\ g_4(t_c) & > & 0 \\ g_5(t_c) & < & 0 \end{array} \right. \quad \text{ou} \quad \left\{ \begin{array}{rcl} g_1(t_c) & = & 0 \\ g_2(t_c) & < & 0 \\ g_3(t_c) & > & 0 \\ g_4(t_c) & < & 0 \\ g_5(t_c) & > & 0 \end{array} \right. \tag{7.3}$$

7.3 Aceleração da deteção de colisões

A deteção de colisões é uma das principais dificuldades computacionais na simulação de um sistema dinâmico. Em nosso caso, a dificuldade é ainda maior pelo fato de que a geometria dos corpos varia ao longo da simulação. Em decorrência, não podemos utilizar (pelo menos sem adaptação) os métodos mais eficientes de deteção de colisões disponíveis na literatura da área, que são orientados para corpos rígidos [Bar90, LM94, PML95].

Se m é o número total de elementos geométricos que podem entrar em colisão, em um dado modelo, então a cada instante há $\Theta(m^2)$ possíveis colisões. Ponamgi, Manocha e Lin [PML95] descrevem um método eficiente de deteção de colisões entre pares de corpos rígidos, cuja primeira etapa consiste em reduzir o número de pares que precisam ser testados, selecionando apenas aqueles pares que têm alguma chance de colisão em um dado intervalo de tempo. Nesta seção nós descrevemos um algoritmo, baseado nessa primeira etapa, que é capaz de reduzir significativamente o número de pares vértice-face e aresta-aresta a testar em cada passo da simulação. Como veremos, o tempo de pior caso desse algoritmo é $O(m^2)$, mas seu tempo de execução médio na grande maioria dos casos práticos é aproximadamente linear em m.

A idéia básica do método de Ponamgi, Manocha e Lin é utilizar a *coerência temporal* inerente ao sistema dinâmico: de um estado do sistema para o próximo, as posições dos vértices mudam pouco, em relação ao tamanho dos objetos. Assim, se dois elementos de superfície estavam próximos/afastados no estado anterior, em comparação à distância média entre elementos, é bastante provável que continuem próximos/afastados no estado seguinte.

7.3.1 Estrutura geral do algoritmo

Na nossa implementação, o algoritmo de deteção de colisões usa certas estruturas de dados auxiliares descritas mais adiante. Essas estruturas permitem localizar rapidamente um conjunto pequeno de pares de elementos que estão suficientemente próximos para permitir sua colisão. Os testes geométricos de colisão vértice-face e aresta-aresta (descritos na seção (7.2)) são aplicados apenas a estes pares. A cada passo da simulação, o algoritmo faz modificações locais nestas estruturas para refletir o movimento dos objetos entre os dois últimos estados calculados.

7.3.2 Caixas envoltórias

Dado um intervalo $I = [t_a, t_b]$ entre dois estados consecutivos do sistema, calculamos uma *caixa envoltória*, de faces paralelas aos planos cartesianos, para cada vértice externo, aresta externa e face externa do modelo. O tamanho de cada caixa é o mínimo necessário para conter as trajetórias aproximadas das partículas do elemento de superfície (vértice, aresta ou face) ao longo do intervalo I.

A caixa que envolve a trajetória de um vértice é obtida diretamente dos coeficientes de Bernstein-Bézier para os três polinômios de Hermite que aproximam as coordenadas x, y e z desse vértice no intervalo I — veja a seção (7.1.5). A caixa de uma aresta, por sua vez, é a menor caixa que contém as caixas dos seus dois vértices, e a caixa de uma face é a menor caixa que contém a caixa de uma aresta da face e a caixa do terceiro vértice da face.

7.3.3 Listas de intervalos

Para cada eixo cartesiano Ox, Oy, Oz, nós mantemos uma *lista de extremos de intervalos* (L^x, L^y, L^z), sendo que cada intervalo é a projeção de uma caixa envoltória sobre um dos três eixos. Cada lista é mantida em ordem não decrescente das coordenadas dos extremos "esquerdo" e "direito" dos seus intervalos.

Sejam e_i e e_j dois elementos de superfície distintos de um dado modelo, e B_i e B_j suas respectivas caixas envoltórias, para um dado intervalo I. O par de elementos (e_i, e_j) pode colidir em um instante $t_c \in I$ somente se B_i e B_j têm interseção não nula em I. Verifica-se que um par de caixas tem interseção não nula se e somente se a projeção de uma caixa sobre cada eixo intercepta a projeção da outra caixa no mesmo eixo.

7.3.4 Tabelas de interseção

A fim de registrar se duas caixas se interceptam ou não, utilizamos três tabelas de bits S^x, S^y e S^z, uma para cada eixo, com um bit para cada par de elementos de superfície. Sejam P_i^x e P_j^x os intervalos de projeção sobre o eixo Ox das caixas envoltórias de dois elementos de superfície e_i e e_j, respectivamente. O valor de S_{ij}^x é $\mathbf{0}$ quando a interseção entre os intervalos P_i^x e P_j^x é nula, e $\mathbf{1}$ quando a interseção é não nula. As tabelas S^y e S^z são análogas.

7.3.5 Tabela de relevância

Utilizamos também uma quarta tabela de bits, a *tabela de relevância* R, que registra os pares de elementos de superfície cuja colisão em um dado intervalo I é possível e permitida. Em nossa formulação, R_{ij} é $\mathbf{0}$ quando o par (e_i, e_j) é composto de dois vértices, duas faces, um vértice e uma aresta, ou uma aresta e uma face. Além disso, R_{ij} é $\mathbf{0}$ quando os elementos e_i e e_j pertencem ao mesmo tetraedro ou possuem um vértice em comum. Nos demais casos, R_{ij} é $\mathbf{1}$.

Esta tabela é calculada numa fase de pré-processamento, sendo esporadicamente alterada ao longo da simulação — veja a seção (7.4).

7.3.6 Conjunto de candidatos

Finalmente, o algoritmo mantém um *conjunto de candidatos* C, que consiste de todos os pares vértice-face e aresta-aresta cujas caixas envoltórias se interceptam num dado intervalo I.

7.3.7 Algoritmo detalhado

Inicialmente, as listas L^x, L^y e L^z estão arbitrariamente ordenadas, as tabelas S^x, S^y e S^z estão inteiramente nulas (por hipótese supomos que no instante inicial da simulação não há interpenetração entre corpos), e o conjunto C está vazio.

Para cada novo intervalo $I = [t_a, t_b]$, o algoritmo recebe como entrada: as listas L^x, L^y e L^z, as tabelas S^x, S^y, S^z e R, e o conjunto de candidatos C, utilizados no intervalo anterior, mais os estados do sistema nos instantes t_a e t_b. Em seguida, o algoritmo atualiza o conjunto de candidatos e testa os pares resultantes, como descrito a seguir.

1. Recalcula as caixas envoltórias dos vértices, das arestas e das faces, para o novo intervalo I.

2. Reordena as listas de extremos de intervalos por inserção simples. Durante a ordenação da lista L^x, sempre que dois extremos pertencentes a um par de intervalos (P_i^x, P_j^x) trocam de lugar, o valor de S_{ij}^x é atualizado. Se após a atualização de S_{ij}^x tem-se que $S_{ij}^x = S_{ij}^y = S_{ij}^z = \mathbf{1}$ (ou seja, as caixas envoltórias B_i e B_j dos elementos e_i e e_j se sobrepõem nos três eixos), o par (B_i, B_j) é adicionado ao conjunto de candidatos C. Operações similares são realizadas durante a ordenação das listas L^y e L^z.

3. Para cada par candidato $(B_i, B_j) \in C$, se as caixas B_i e B_j não se sobrepõem em algum eixo, esse par é eliminado do conjunto. Caso contrário, o teste de colisão apropriado é aplicado às trajetórias dos elementos de superfície e_i, e_j, no intervalo I.

7.3.8 Análise do algoritmo acelerado

Seja m o número total de vértices externos, arestas externas e faces externas de um dado modelo. O tempo para recalcular as caixas envoltórias (passo 1 na seção anterior) é $\Theta(m)$. O exame do conjunto de candidatos (passo 3) leva, no pior caso, tempo $\Theta(m^2)$, mas nos casos práticos o conjunto C tem tamanho médio $O(m)$. Na etapa de ordenação das listas de extremos de intervalos (passo 2), o tempo de execução é $O(m + p)$, onde p é o número de trocas realizadas na reordenação das três listas.

No pior caso, $p = \Theta(m^2)$; em média, entretanto, é muito menor. Seja v a velocidade média dos elementos de superfície ao longo do intervalo de tempo $I = [t, t + \tau]$ entre dois estados consecutivos, e D o diâmetro médio da cena em I. Supondo que as arestas de superfície têm comprimentos uniformes, e as faces triangulares são aproximadamente eqüiláteras, podemos concluir que o número esperado p de trocas em I é

$$p \in O\left(\frac{\tau\, v}{D} m^2\right)$$

Supondo agora que o deslocamento médio $\tau\, v$ dos elementos é comparável ao seu tamanho médio D/m, concluimos que $p \in O(m)$.

7.3.9 Redução das tabelas

Quanto ao espaço ocupado pelas tabelas de bits, podemos tirar proveito do fato de que somente os pares vértice-face e aresta-aresta são permitidos. Sejam m_v, m_a e m_f os números de vértices, arestas e faces externas do modelo, respectivamente. Temos então um máximo de $m_v m_f$ pares vértice-face relevantes, e um máximo de $m_a(m_a-1)/2$ pares aresta-aresta relevantes. Para reduzir o espaço de armazenamento, nós efetivamente utilizamos dois grupos de quatro tabelas, um para os pares vértice-face e outro para os pares aresta-aresta.

Note que cada uma das quatro tabelas dos pares aresta-aresta é simétrica, e todos os elementos da diagonal principal de cada tabela são nulos.

7.4 Molas de contato

Sempre que uma colisão é detetada entre dois elementos de superfície e_i, e_j, nós fixamos uma mola aos dois pontos de contato e deixamos de considerar, para efeitos de deteção de colisão, o par (e_i, e_j) nos passos seguintes da simulação (atribuindo $\mathbf{0}$ à R_{ij} na tabela de relevância R), até que a mola seja eliminada (quando então fazemos $R_{ij} = \mathbf{1}$). Enquanto há interpenetração, essa mola aplica uma força de igual intensidade e direções contrárias àqueles dois pontos, resistindo ao aumento da distância entre os mesmos. Quando a interpenetração cessa, a mola é removida.

7.4.1 A força da mola

A força $\mathbf{f}^{\mathrm{m}}$ de cada mola é dada por

$$\mathbf{f}^{\mathrm{m}} = k(\mathbf{b} - \mathbf{a})$$

onde $\mathbf{a}, \mathbf{b}$ são as posições dos pontos de contato num dado estado, e k é a constante da mola, dada por um valor especificado pelo usuário no início da simulação para todas as possíveis molas de contato.

7.4.2 Colocação da mola

Sejam u_1, u_2, u_3, u_4 os quatro vértices envolvidos em uma colisão, cujas posições no instante t_c da colisão são $\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4$, respectivamente. Para um contato entre vértice e face, u_4 é o vértice (de posição $\mathbf{a}$) e $\mathsf{F} = (u_1, u_2, u_3)$ é a face, e $\mathbf{n}_1, \mathbf{n}_2, \mathbf{n}_3$ são calculados conforme descrito na seção (7.2.1). Calculamos as coordenadas baricêntricas de $\mathbf{b} = \alpha_1\mathbf{p}_1 + \alpha_2\mathbf{p}_2 + \alpha_3\mathbf{p}_3$ em relação a F como

$$\alpha_1 = \frac{\mathbf{n}_1 \cdot (\mathbf{b} - \mathbf{p}_1)}{\mathbf{n}_1 \cdot (\mathbf{p}_2 - \mathbf{p}_1)}$$

$$\alpha_2 = \frac{\mathbf{n}_2 \cdot (\mathbf{b} - \mathbf{p}_1)}{\mathbf{n}_2 \cdot (\mathbf{p}_2 - \mathbf{p}_1)}$$

$$\alpha_3 = 1 - \alpha_1 - \alpha_2$$

Quando a colisão ocorre, nós retiramos todos os pares vértice-face que possuem o vértice u_4 da deteção de colisões (efetivamente, zerando a i-ésima linha da tabela R).

Para um contato entre aresta e aresta, $\mathbf{a}$ é a posição de uma partícula na aresta $\mathsf{A}_1 = (u_1, u_2)$ e $\mathbf{b}$ é a posição de uma partícula na aresta $\mathsf{A}_2 = (u_3, u_4)$. As coordenadas baricêntricas de $\mathbf{a} = \alpha_1\mathbf{p}_1 + \alpha_2\mathbf{p}_2$ em relação a A_1 e de $\mathbf{b} = \alpha_3\mathbf{p}_3 + \alpha_4\mathbf{p}_4$ em relação a A_2 são dadas por

$$\alpha_1 = \frac{(\mathbf{p}_2 - \mathbf{a}) \cdot (\mathbf{p}_2 - \mathbf{p}_1)}{(\mathbf{p}_2 - \mathbf{p}_1) \cdot (\mathbf{p}_2 - \mathbf{p}_1)}$$

$$\alpha_2 = 1 - \alpha_1$$

$$\alpha_3 = \frac{(\mathbf{p}_4 - \mathbf{b}) \cdot (\mathbf{p}_4 - \mathbf{p}_3)}{(\mathbf{p}_4 - \mathbf{p}_3) \cdot (\mathbf{p}_4 - \mathbf{p}_3)}$$

$$\alpha_4 = 1 - \alpha_3$$

7.4.3 Retirada da mola

A mola entre um vértice v_i e uma face f_j é retirada quando v_i retorna para o lado positivo da face f_j; mais precisamente, quando a folga g_1 atinge um valor positivo e pequeno ϵ.

A mola entre duas arestas a_i e a_j é retirada quando a folga g_1 correspondente (o determinante dos quatro vértices de a_i e a_j) atinge o valor $\delta\epsilon$, onde ϵ é uma constante positiva pequena, e δ é o sinal de g_1 imediatamente antes da colisão:

$$\delta = \begin{cases} +1 & \text{se } g_1(t_c) > 0 \\ -1 & \text{se } g_1(t_c) < 0 \end{cases}$$

7.5 Eventos relacionados a vínculos

A nossa implementação permite associar a cada vínculo $H(\mathbf{q}, t) = 0$ um intervalo de tempo $I = [t_a, t_b]$ ao longo do qual o vínculo tem efeito: no instante t_a o vínculo é incluído no sistema de equações, e no instante t_b é removido.

Note que a inclusão ao sistema de um vínculo do tipo $H(\mathbf{q}, t) = 0$ requer alterações triviais no sistema de equações de movimento (3.5): basta adicionar uma linha à matriz $\mathbf{J}$ e um elemento aos vetores $\boldsymbol{\psi}$ e $\boldsymbol{\lambda}$. Para remover o vínculo do sistema de equações, basta eliminar a devida linha de $\mathbf{J}$ e os devidos elementos de $\boldsymbol{\psi}$ e $\boldsymbol{\lambda}$.

Um outro exemplo interessante, mas não implementado em nosso sistema, é o seguinte: suponha que a magnitude da força

$$\mathbf{f}^{\mathrm{v}} = \lambda \frac{\partial H}{\partial \mathbf{q}}(\mathbf{q}, t)$$

associada ao vínculo $H(\mathbf{q}, t) = 0$ é limitada a um dado intervalo, através da condição $|\mathbf{f}^{\mathrm{v}}| < f_{\max}$ onde $f_{\max}$ é uma constante positiva. Tal mecanismo permitiria simular a quebra de articulações entre corpos. Em particular, seria possível simular fraturas num corpo elástico modelando-o como um conjunto de tetraedros unidos explicitamente por vínculos quebráveis.

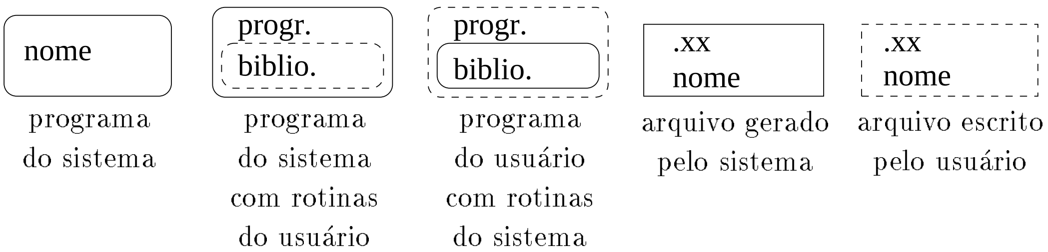
Capítulo 8

Implementação

O sistema de animação de corpos elásticos que implementamos consiste de aproximadamente 10000 linhas de código na linguagem `Modula-3` [Nel91], e 1300 linhas na linguagem `C++` [ES93], distribuídas em vários programas e bibliotecas. Os programas em `C++` utilizam também a biblioteca gráfica `OpenGL` [NDW94], e rodam atualmente apenas em estações `Silicon Graphics Indigo`[2]. Os programas em `Modula-3` rodam tanto em `Silicon Graphics` quanto em estações `Sun`, sob `Solaris 5.5`.

8.1 Estrutura do sistema de animação

A produção de uma animação com nosso sistema consiste de três grandes etapas: a modelagem, a simulação e a visualização. A comunicação entre estas etapas se dá através de sete diferentes tipos de arquivos de texto, identificados por nomes com um sufixo de dois caracteres: estados (`.st`), tetraedros (`.tp`), texturas (`.tx`), superfície (`.su`), restrições (`.co`), luzes (`.li`) e energias (`.en`). A figura (8.1) mostra a estrutura do sistema de animação, indicando cada programa com suas respectivas entradas e saídas, de acordo com as seguintes convenções:



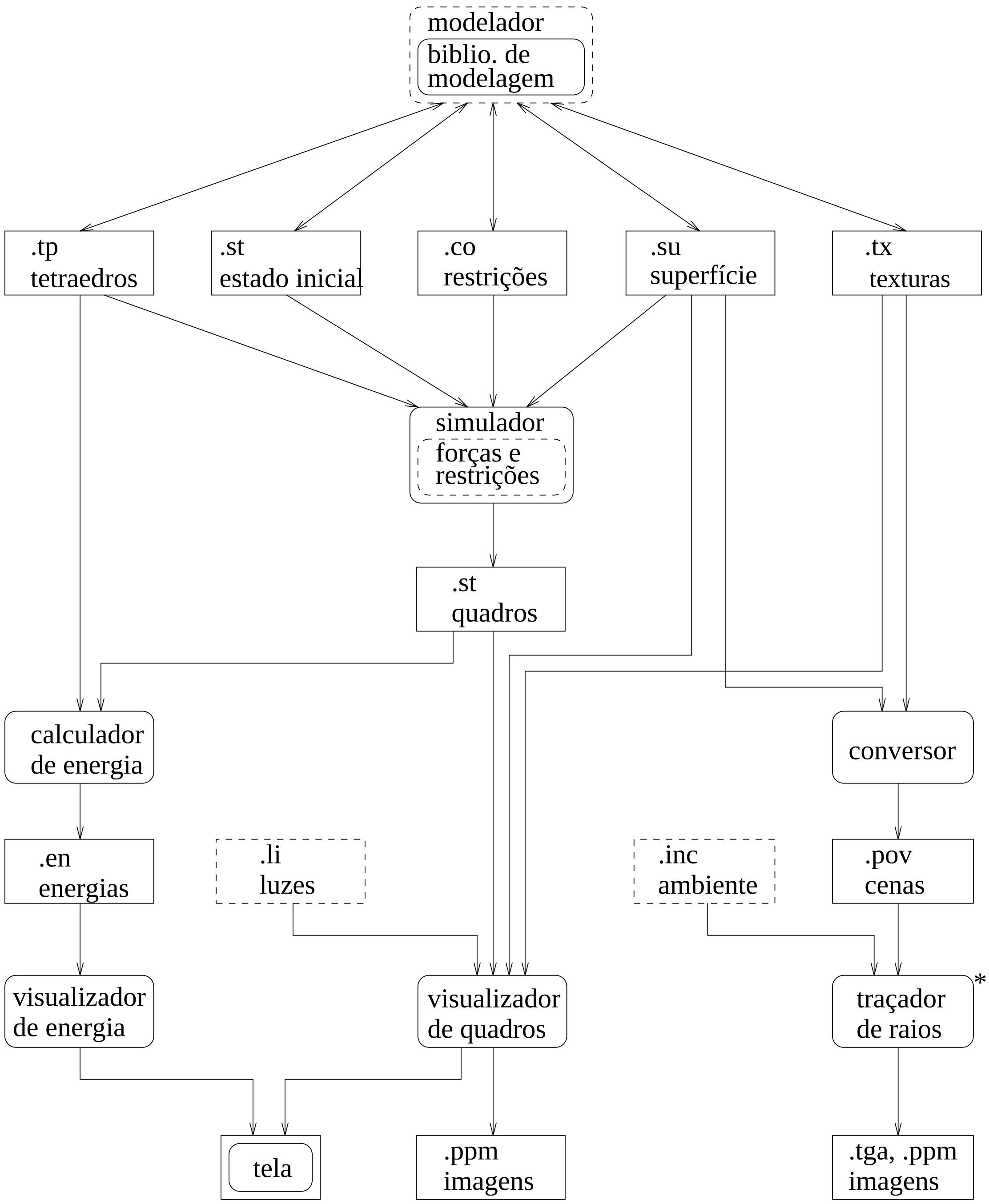


Figura 8.1: Estrutura do sistema de animação

Exceto pelo traçador de raios (marcado com um asterisco), todos os programas foram implementados por nós.

8.2 Modelagem

Na etapa de modelagem o usuário constrói o modelo do sistema dinâmico que será simulado. Esse modelo contém a topologia dos corpos e as configurações de repouso dos tetraedros, as propriedades físicas dos corpos e de suas superfícies, as texturas das superfícies, algumas restrições holonômicas, e o estado inicial do sistema.

8.2.1 Biblioteca de modelagem

O nosso sistema de animação suporta esta etapa através de uma biblioteca de modelagem (em `Modula-3`) que possui uma variedade de ferramentas encapsuladas em diversas classes de objetos e em outros módulos. Para criar os arquivos que descrevem um modelo, o usuário deve escrever um módulo principal que chama os procedimentos nessa biblioteca para criar o modelo, realizar operações de transformação sobre o mesmo, e então gravá-lo em um conjunto de arquivos.

Especificamente, a biblioteca contém procedimentos para: 1) criar blocos, esferas, cilindros, toros, e outros corpos elásticos simples; 2) realizar operações topológicas, como a união de dois corpos e a remoção de tetraedros selecionados de um corpo; 3) definir o estado inicial de um corpo, através de operações de translação, rotação, escala e ajuste das velocidades linear e/ou angular; 4) atribuir propriedades mecânicas e ópticas à superfície de cada corpo; e 5) ler e escrever os arquivos de modelo.

8.2.2 Outros programas

Há ainda um programa gerador de superfície (`surf`) que gera um arquivo de superfície a partir de um arquivo de tetraedros, um programa coloridor (`coloring`) que calcula uma 3-coloração aproximada das faces de um corpo, e outros.

8.3 Simulação

Nesta etapa é realizada a simulação da evolução no tempo do sistema dinâmico, sendo os resultados guardados para posterior visualização.

O programa simulador (`sim`) lê os arquivos de estado inicial (`.st`), de tetraedros (`.tp`), de superfície (`.su`) e de restrições (`.co`), monta as equações de movimento e roda a simulação, escrevendo os quadros em um arquivo de estados. Possui vários parâmetros de linha de comando, tais quais: os instantes inicial e final da simulação, o erro numérico máximo que pode ser tolerado, os intervalos máximo e mínimo entre estados, o módulo da aceleração da gravidade, o número de quadros por segundo a serem salvos, e outros.

8.3.1 Biblioteca de simulação

O simulador utiliza algumas ferramentas de uso geral, coletadas em uma biblioteca de simulação (escrita em `Modula-3`). Em nossa implementação, essa biblioteca contém: 1) um simulador dinâmico que utiliza o método de integração de Runge-Kutta-Fehlberg para equações diferenciais de 2ª ordem; 2) armazenamento e manipulação de matrizes simétricas esparsas; 3) fatoração de Choleski para matrizes simétricas positivo-definidas; 4) resolução de sistemas lineares utilizando fatoração de Choleski; 5) eliminação de Gauss; e outras utilidades diversas.

8.3.2 Forças externas e vínculos

O simulador permite ao usuário especificar a força da gravidade através de parâmetros da linha de comando. Se o sistema dinâmico a ser simulado inclui forças externas (além da gravidade) e/ou restrições de movimento (que não vértices fixos, especificados através de um arquivo `.co`), as classes de objetos que as descrevem devem ser instanciadas no módulo principal do programa simulador, escrito (em `Modula-3`) pelo usuário.

A nossa implementação inclui uma biblioteca de restrições, que implementa várias restrições holonômicas, tais como: tetraedro rígido, tetraedro incompressível, vértice limitado a superfície quádrica, junção entre dois tetraedros, e vértice arrastado ao longo de uma trajetória de Hermite (vide capítulo 6). Para utilizar outras forças externas ou restrições, o usuário terá também de escrever o código que as implementa. Note que quando o sistema dinâmico não inclui forças externas nem restrições, o usuário não precisa escrever nenhum código.

8.4 Visualização

O sistema inclui três ferramentas para a visualização dos resultados da simulação: um visualizador de quadros (`show3d`), um conversor para traçado de raios (`convert`), e um visualizador de energia (`showen`). Os dois visualizadores são os únicos programas do sistema escritos em `C++`.

8.4.1 Visualizador de quadros

O visualizador de quadros lê um arquivo de texturas (`.tx`), um arquivo de superfície (`.su`), um arquivo de luzes (`.li`), e um arquivo com os estados (`.st`) correspondentes aos quadros da animação. O programa exibe interativamente a seqüência de quadros, utilizando os recursos gráficos de uma estação `Silicon Graphics` através da interface fornecida pela biblioteca `OpenGL`. O programa `show3d` possui vários comandos acionados por teclado e *mouse* para movimento da câmara e outras facilidades. Na linha de comando, o usuário fornece parâmetros como resolução horizontal e vertical, posição inicial da câmara, etc.

8.4.2 Conversor para traçado de raios

O conversor (`convert`) lê um arquivo de estados, um arquivo de texturas e um arquivo de superfície e converte a descrição de cada quadro em um arquivo de cena (`.pov`) no formato do programa de traçado de raios `POV-Ray 2.2`, que gera imagens de melhor qualidade, incluindo sombras, reflexões, etc.

8.4.3 Gráficos de energia

O visualizador de energia (`showen`) lê um arquivo de energias (`.en`) e desenha gráficos sobrepostos de tempo $\times$ energia. O arquivo de energias é produzido por um programa calculador de energia (`energy`), que lê um arquivo de tetraedros (`.tp`) e um arquivo de estados.

8.5 Formatos de arquivos

As seções seguintes descrevem o formato dos vários tipos de arquivo. Para melhorar a legibilidade e facilitar o controle de versões, todos eles começam com um cabeçalho da forma "`begin` *tipo* `format of` (*aa-mm-dd*)", e terminam com um rodapé da forma "`end` *tipo*", onde *tipo* é o nome do tipo do arquivo e *aa-mm-dd* é a data (ano-mês-dia) da versão do formato.

8.5.1 Arquivo de estados

Este tipo de arquivo (`.st`) é usado para o estado inicial do sistema, mas também para a saída do simulador, de modo que pode conter vários estados. Seu formato consiste de um ou mais dos grupos `begin state` ... `end state` abaixo, separados por zero ou mais linhas em branco:

```
begin state (format of aa-mm-dd)
vertices = n
t = t
⋮
i:  px py pz vx vy vz
⋮
end state
```

onde n é o número de vértices, t é o instante do estado em segundos, $i = 0, \ldots, n-1$ é o número de um vértice, $\mathbf{p} = [p_x, p_y, p_z]$ é a posição do vértice e $\mathbf{v} = [v_x, v_y, v_z]$ é sua velocidade.

8.5.2 Arquivo de tetraedros

Este arquivo (`.tp`) contém a lista de tetraedros que formam os corpos elásticos de um sistema. Para cada tetraedro são dados os quatro vértices que o identificam, sua configuração de repouso e suas propriedades materiais.

$$
\begin{array}{l}
\texttt{begin topology (format of } \textit{aa-mm-dd}\texttt{)} \\
\texttt{tetrahedrons = } m \\
\vdots \\
\begin{array}{lccccccccc}
i: & a_{11} & a_{12} & a_{13} & a_{21} & a_{22} & a_{23} & a_{31} & a_{32} & a_{33} \\
 & p_1 & p_2 & p_3 & p_4 & & & & & \\
 & \rho & \alpha & \beta & \eta_1 & \eta_2 & & & &
\end{array} \\
\vdots \\
\texttt{end topology}
\end{array}
$$

onde m é o número de tetraedros, $i = 0, \ldots, m-1$ é o número de um tetraedro, os a_{ij} são os elementos de uma matriz $\mathbf{A}$ (3×3) que descreve a forma de repouso do tetraedro, $p_1, \ldots, p_4$ são os números dos vértices do tetraedro, referentes à numeração utilizada no arquivo de estados, e $\rho, \alpha, \beta, \eta_1$ e η_2 são a densidade do tetraedro, os módulos de elasticidade (volumétrico e de rigidez), e os coeficientes de viscosidade (volumétrico e de forma), respectivamente.

8.5.3 Arquivo de texturas

Cada textura em um arquivo deste tipo (`.tx`) é descrita por um certo conjunto de valores que definem as propriedades ópticas de uma superfície, sendo que a cada face da superfície de um corpo é atribuída uma textura. A implementação corrente suporta apenas uma textura uniforme para cada face externa do modelo.

$$
\begin{array}{l}
\texttt{begin textures (format of } \textit{aa-mm-dd}\texttt{)} \\
\texttt{textures = } n_t \\
\vdots \\
a_R \; a_G \; a_B \; d_R \; d_G \; d_B \; s_R \; s_G \; s_B \; t_R \; t_G \; t_B \; ir \; p \\
\vdots \\
\texttt{end textures}
\end{array}
$$

onde n_t é o número de texturas, a, d, s e t referem-se às intensidades (R, G e B correspondem aos componentes vermelho, verde e azul) de luz ambiente, difusa, especular e transmissiva, respectivamente, ir é o índice de refração e p é o expoente especular para o modelo de iluminação de Phong. A numeração das texturas segue a ordem das linhas do arquivo de texturas, começando com zero para a primeira textura.

8.5.4 Arquivo de superfície

Neste tipo de arquivo (`.su`), são descritas as faces externas do modelo.

```
begin surface (format of aa-mm-dd)
faces = n_f
⋮
a b c tx μ1 μ2 e
⋮
end surface
```

onde n_f é o número de faces, a, b e c são os números dos vértices da face em ordem anti-horária, tx é o número da textura a ser utilizada na visualização da face, e μ_1, μ_2 e e são os coeficientes de atrito estático, atrito dinâmico e restituição, respectivamente. A numeração das faces é implícita, começando com zero para a primeira face no arquivo.

8.5.5 Arquivo de restrições simples

Este arquivo (`.co`) permite especificar vértices fixos do modelo, que são um caso particular de restrição holonômica.

```
begin constraints (format of aa-mm-dd)
constraints = n_c
⋮
p
⋮
end constraints
```

onde n_c é o número de restrições e p é o número de um vértice.

8.5.6 Arquivo de luzes

Este arquivo (`.li`) é escrito pelo próprio usuário, e fornece as fontes pontuais de luz que podem ser utilizadas para a visualização de uma animação.

> `begin lights (format of` *aa-mm-dd*`)`
> `lights =` n_l
> $\vdots$
> $x\ y\ z\ w\ R\ G\ B$
> $\vdots$
> `end lights`

onde n_l é o número de luzes, $\mathbf{p} = [x, y, z, w]$ são as coordenadas homogêneas da fonte de luz (pontual) e R, G e B dão a cor da luz emitida pela fonte.

8.5.7 Arquivo de energias

O arquivo de energias (`.en`) contém, para cada estado em um dado arquivo de estados, uma linha com os valores de algumas quantidades físicas.

> `begin energies (format of` *aa-mm-dd*`)`
> `states =` s
> $\vdots$
> $t\ K\ P\ G\ W$
> $\vdots$
> `end energies`

onde s é o número de estados, t é o instante de um estado e K, P, G e W são os valores da energia cinética, energia potencial elástica, energia potencial gravitacional e potência dissipada pelas forças de viscosidade interna, respectivamente, todos referentes ao estado do sistema no instante t.

Capítulo 9

Animações de exemplo

As animações apresentadas neste capítulo tentam demonstrar a maioria das técnicas descritas nesta dissertação, e que foram implementadas no sistema de animação. Cada animação é acompanhada de uma tabela com os parâmetros relevantes e o tempo de processamento, e de uma seqüência selecionada de quadros, que deve ser lida na ordem usual de leitura.

Todas as animações foram simuladas utilizando um processador numa `Sparc 1000`, sendo gerados 30 quadros por segundo para cada uma.

Para estas animações mostramos também gráficos da energia do sistema em função do tempo, calculada nos estados correspondentes aos quadros. Os gráficos mostrados são os de energia cinética (K), energia potencial elástica (P), energia potencial gravitacional (G, em relação a um nível de referência dependente da animação), energia total dissipada desde o instante inicial (D), energia útil ($U = K + P + G$), e energia total ($T = U + D$). Note que esta última curva, quando não há forças externas além da gravidade, nem restrições holonômicas dependentes do tempo, deveria ser uma reta horizontal. Entretanto, nós não contabilizamos a energia potencial das molas de contato.

A seção final sumariza e compara os dados das animações apresentadas.

9.1 Toro caindo com vértice preso

Na animação `torus`, um toro inicialmente em repouso, com um vértice interno fixo, é largado no espaço para cair sob a ação da gravidade. O toro se deforma à medida que cai, acabando por ficar balançando pendurado no vértice fixo. A orientação inicial do toro é assimétrica em relação à direção da força de gravidade, de modo a evitar deformações simétricas.

Em nenhum momento o toro colide com si mesmo. Por isso, a simulação foi realizada com a deteção de colisões desativada. Ao realizarmos a mesma simulação com a deteção de colisões ativada, verificamos que o tempo real aumentou aproximadamente 50%.

Densidade (ρ)	1 $\texttt{g/cm}^3$
Módulo volumétrico (α)	220000 $\texttt{dyna/cm}^2$
Módulo de rigidez (β)	160000 $\texttt{dyna/cm}^2$
Coef. de viscosidade volumétrico (η_1)	30 `joule/s`
Coef. de viscosidade laminar (η_2)	20 `joule/s`
Número de tetraedros	480
Número de vértices do modelo	180
Número de vértices fixos	1
Número de vértices externos	160
Número de arestas externas	480
Número de faces externas	320
Números de pares a testar	0
Aceleração da gravidade	980 $\texttt{cm/s}^2$
Tempo simulado	10 `s`
Tempo real (hh:mm:ss)	03:36:11
Razão tempo real/tempo simulado	1297

Tabela 9.1: Parâmetros e tempo de processamento da animação `torus`

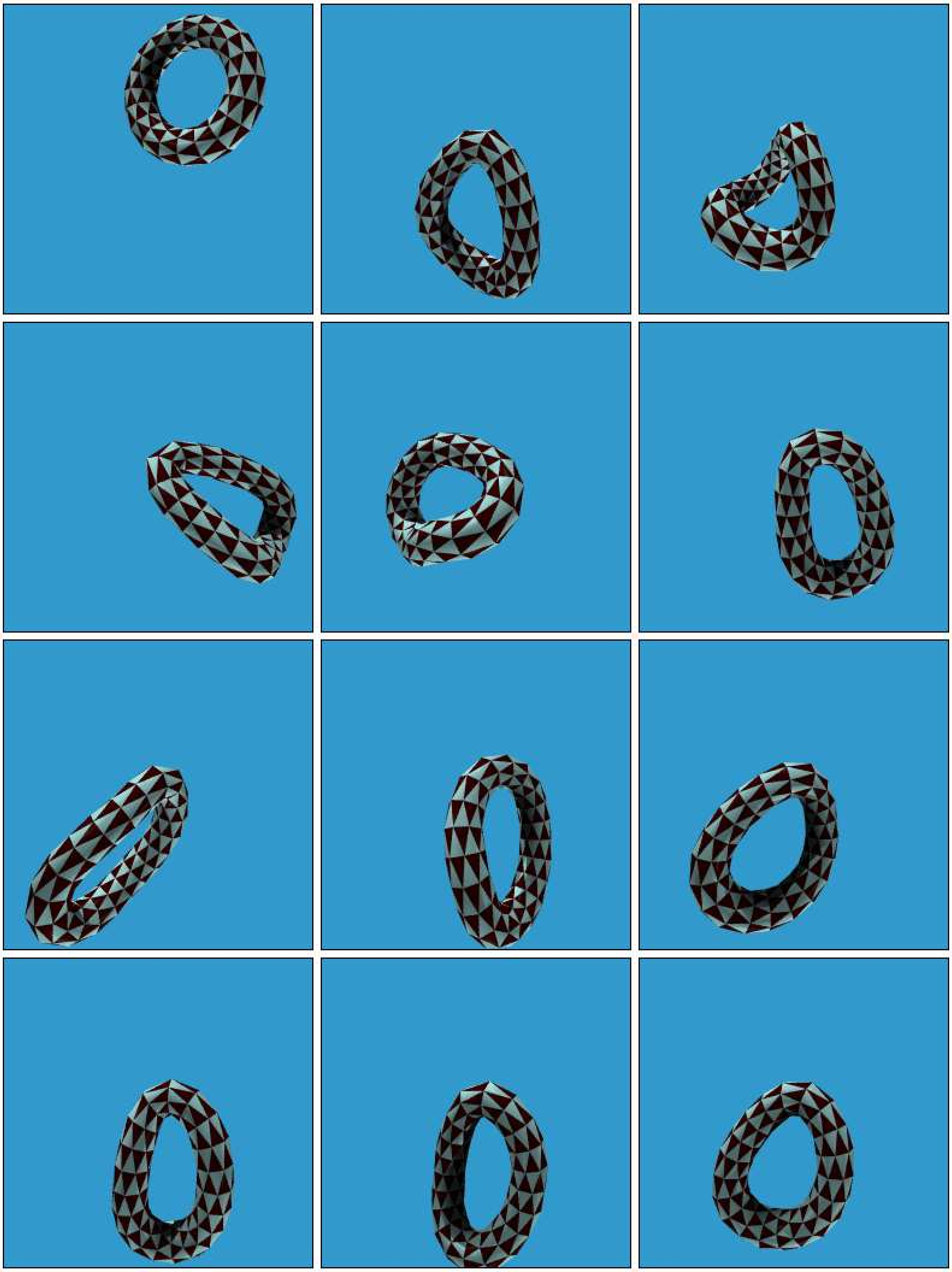


Figura 9.1: Quadros selecionados da animação `torus`

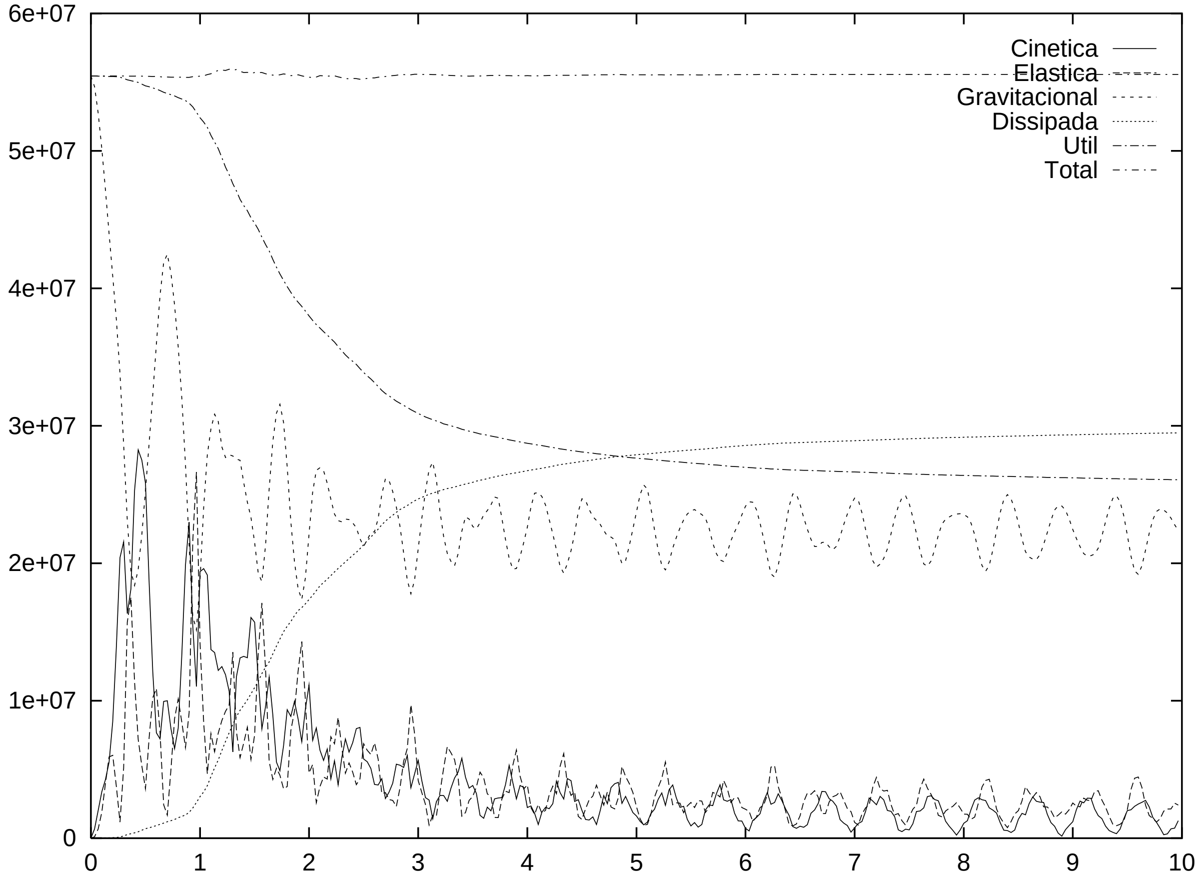


Figura 9.2: Gráficos de energia para a animação `torus`

9.2 Cama elástica

Na animação `elast`, dois objetos deformáveis (um cubo e uma bola, sendo esta última aproximada por um icosaedro) caem sobre uma cama elástica, modelada como um bloco de $8 \times 8 \times 2$ blocos elementares. Os vértices nos cantos da cama são fixos. Observe que no terceiro quadro, o cubo está caindo de cima da cama, desaparecendo da cena a partir do quadro seguinte.

Objeto	**cubo**
Densidade (ρ)	2 `g/cm`^{3}
Módulo volumétrico (α)	20000 `dyna/cm`^{2}
Módulo de rigidez (β)	20000 `dyna/cm`^{2}
Coef. de viscosidade volumétrico (η_1)	16 `joule/s`
Coef. de viscosidade laminar (η_2)	16 `joule/s`
Objeto	**bola**
Densidade (ρ)	2 `g/cm`^{3}
Módulo volumétrico (α)	10000 `dyna/cm`^{2}
Módulo de rigidez (β)	10000 `dyna/cm`^{2}
Coef. de viscosidade volumétrico (η_1)	40 `joule/s`
Coef. de viscosidade laminar (η_2)	40 `joule/s`
Objeto	**cama**
Densidade (ρ)	0,8 `g/cm`^{3}
Módulo volumétrico (α)	100000 `dyna/cm`^{2}
Módulo de rigidez (β)	80000 `dyna/cm`^{2}
Coef. de viscosidade volumétrico (η_1)	8 `joule/s`
Coef. de viscosidade laminar (η_2)	6 `joule/s`
Número de tetraedros	836
Número de vértices do modelo	283
Número de vértices fixos	12
Número de vértices externos	232
Número de arestas externas	678
Número de faces externas	452
Números de pares a testar	329339
Aceleração da gravidade	400 `cm/s`^{2}
Tempo simulado	10 `s`
Tempo real (hh:mm:ss)	19:08:28
Razão tempo real/tempo simulado	6891

Tabela 9.2: Parâmetros e tempo de processamento da animação `elast`

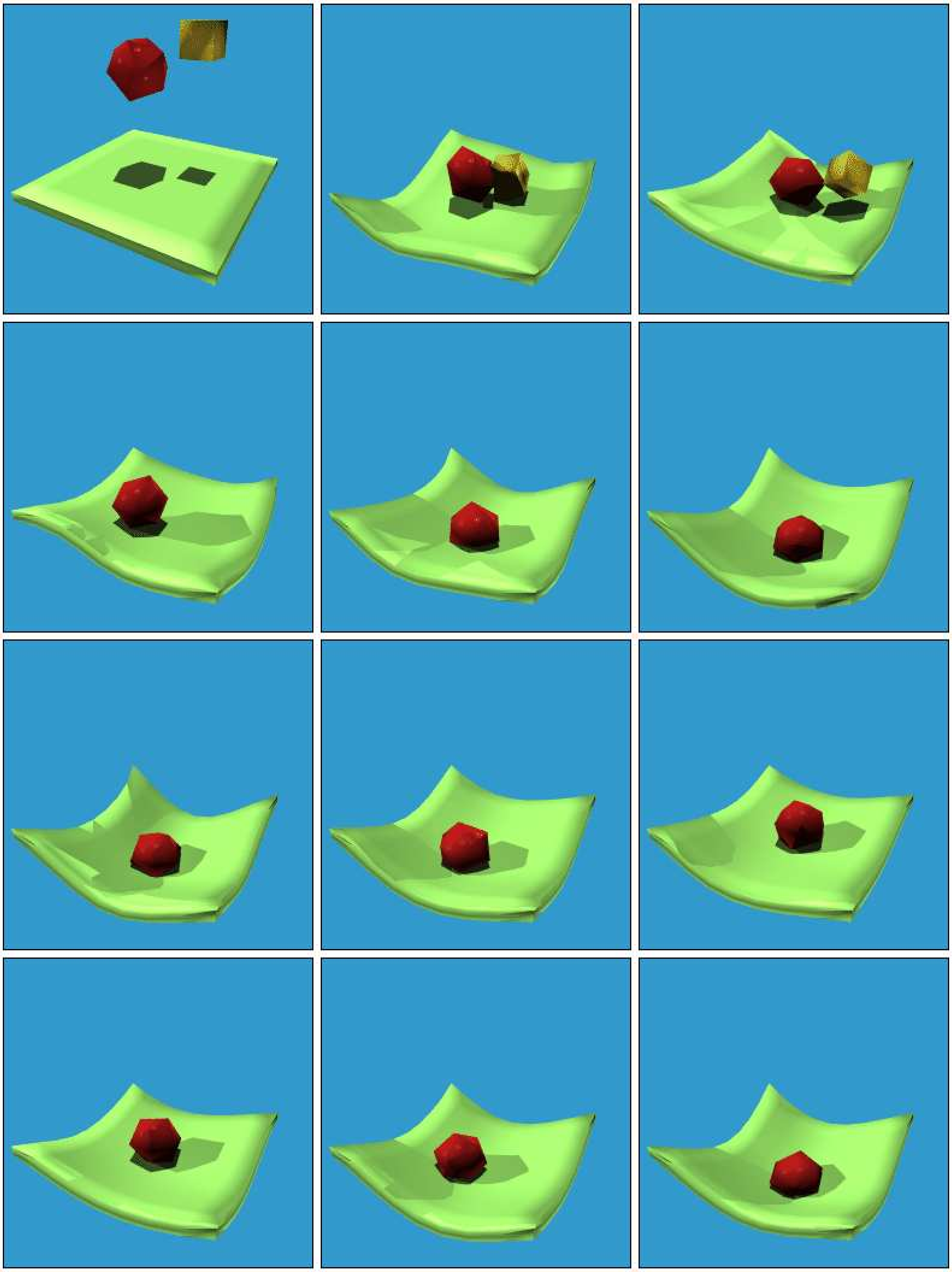


Figura 9.3: Quadros selecionados da animação `elast`

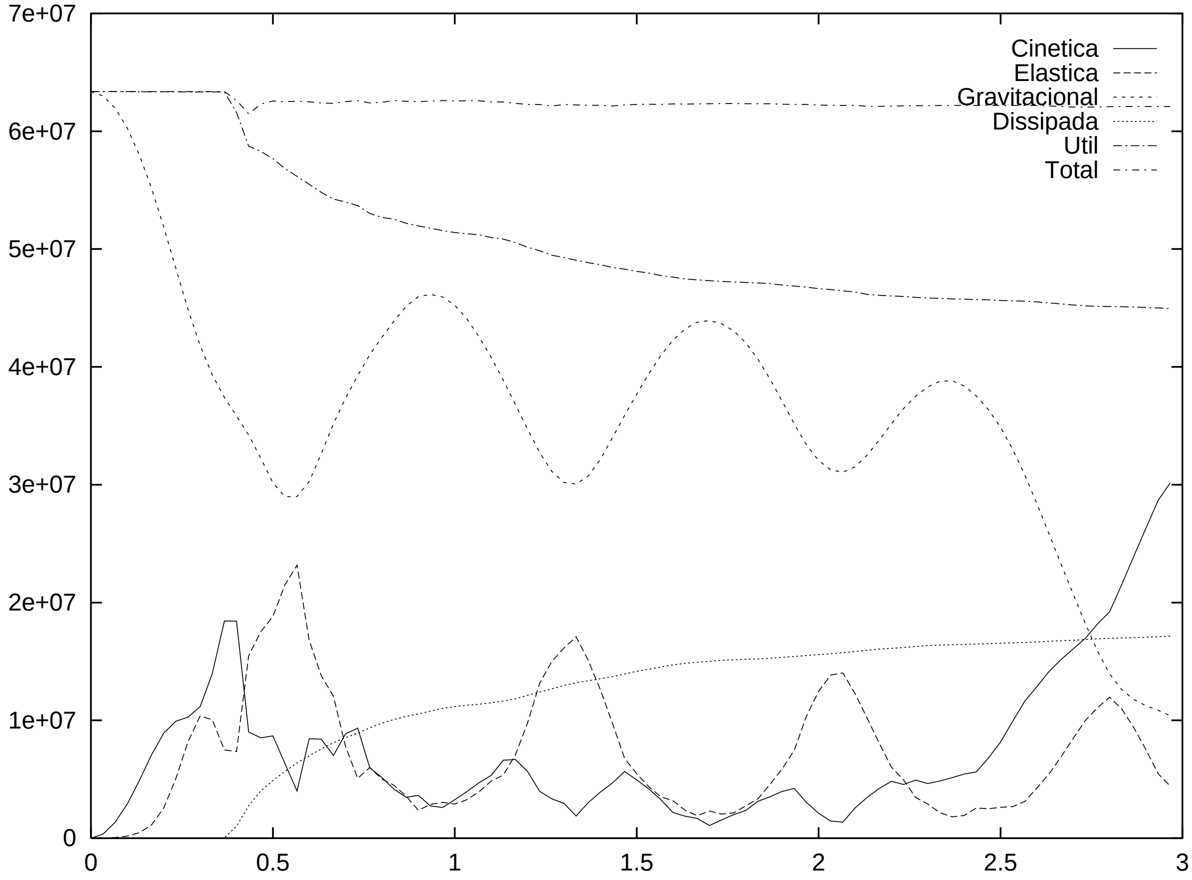


Figura 9.4: Curvas de energia para a animação `elast`

9.3 Alça cinemática

A cena da animação `loop` é constituída de três corpos: um cilindro na vertical, no lado esquerdo da cena, com as duas extremidades fixas, um outro cilindro na vertical, no lado direito, com a extremidade inferior fixa, e um toro inicialmente na horizontal, em torno do primeiro cilindro. As velocidades iniciais (no instante $t_0 = 0$) de todos os vértices são nulas. Uma restrição cinemática é aplicada a um vértice interno do toro para que ele seja levado de sua posição inicial (entre os dois cilindros) até uma posição mais a direita e acima, porém ainda entre os cilindros. Quando o vértice chega a essa posição, no instante $t_1 = 1,5s$, uma segunda restrição cinemática entra em ação, levando-o acima do segundo cilindro. A partir desta posição, atingida no instante $t_2 = 3s$, uma terceira e última restrição cinemática começa a arrastar o vértice para a direita e para baixo, até o instante $t_3 = 5s$. O objetivo do animador era prender o toro em volta dos dois cilindros, mas esse objetivo foi frustrado pela deformação do segundo cilindro, terminando o toro por se soltar do primeiro cilindro e sair da cena. O restante do toro acompanha passivamente o vértice arrastado, esticando-se enquanto é segurado pelo cilindro da esquerda. Os cilindros também são deformados à medida que colidem com o toro.

Objeto	**cilindro da esquerda**
Densidade (ρ)	3 `g/cm`^{3}
Módulo volumétrico (α)	10000 `dyna/cm`^{2}
Módulo de rigidez (β)	9000 `dyna/cm`^{2}
Coef. de viscosidade volumétrico (η_1)	10 `joule/s`
Coef. de viscosidade laminar (η_2)	10 `joule/s`
Objeto	**cilindro da direita**
Densidade (ρ)	1,5 `g/cm`^{3}
Módulo volumétrico (α)	40000 `dyna/cm`^{2}
Módulo de rigidez (β)	25000 `dyna/cm`^{2}
Coef. de viscosidade volumétrico (η_1)	4 `joule/s`
Coef. de viscosidade laminar (η_2)	4 `joule/s`
Objeto	**toro**
Densidade (ρ)	3 `g/cm`^{3}
Módulo volumétrico (α)	8000 `dyna/cm`^{2}
Módulo de rigidez (β)	8000 `dyna/cm`^{2}
Coef. de viscosidade volumétrico (η_1)	20 `joule/s`
Coef. de viscosidade laminar (η_2)	20 `joule/s`
Número de tetraedros	906
Número de vértices do modelo	353
Número de vértices fixos	34
Número de vértices externos	321
Número de arestas externas	945
Número de faces externas	630
Números de pares a testar	639101
Aceleração da gravidade	0 `cm/s`^{2}
Tempo simulado	15 `s`
Tempo real (hh:mm:ss)	36:32:14
Razão tempo real/tempo simulado	8769

Tabela 9.3: Parâmetros e tempo de processamento da animação `loop`

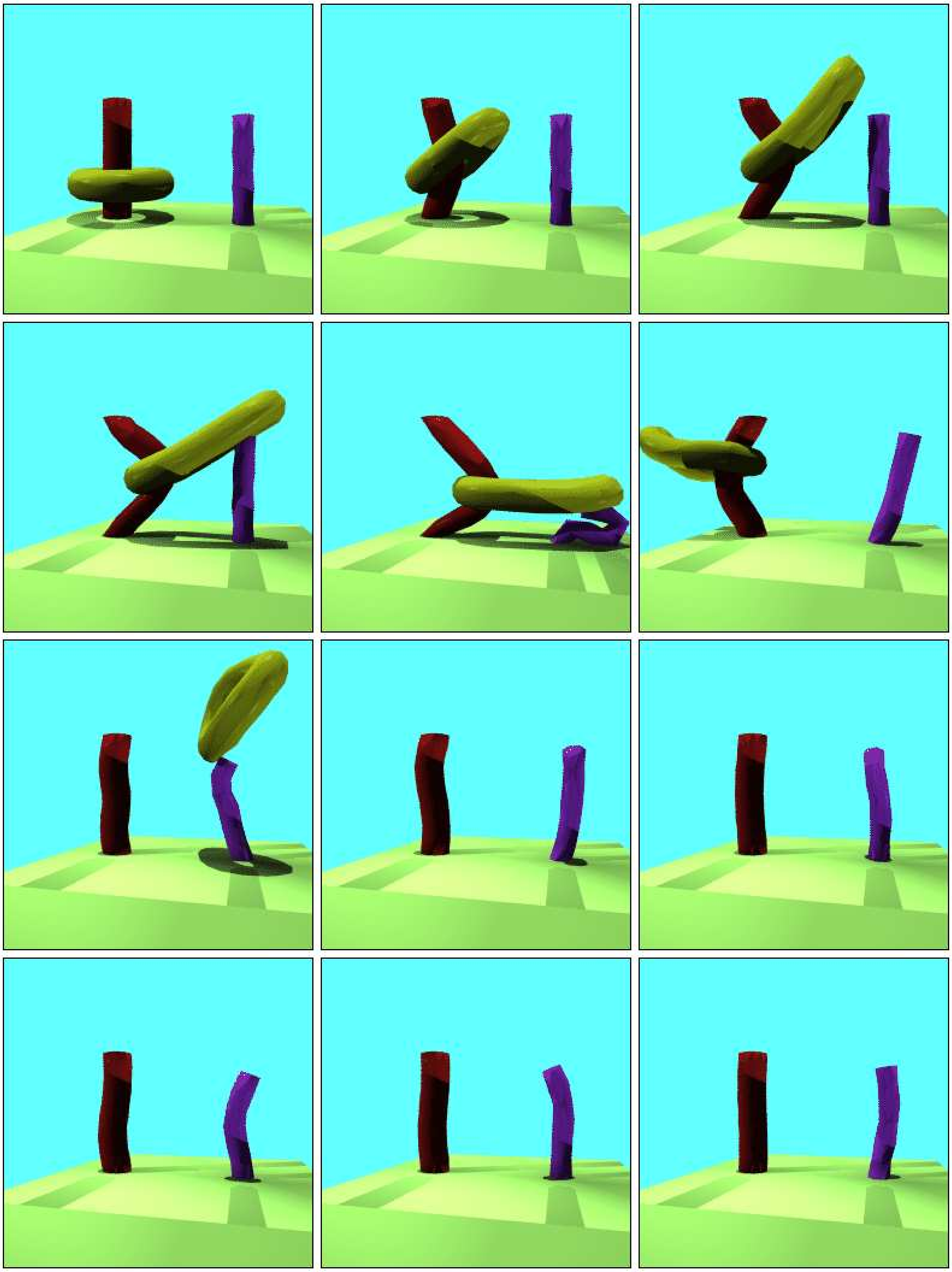


Figura 9.5: Quadros selecionados da animação `loop`

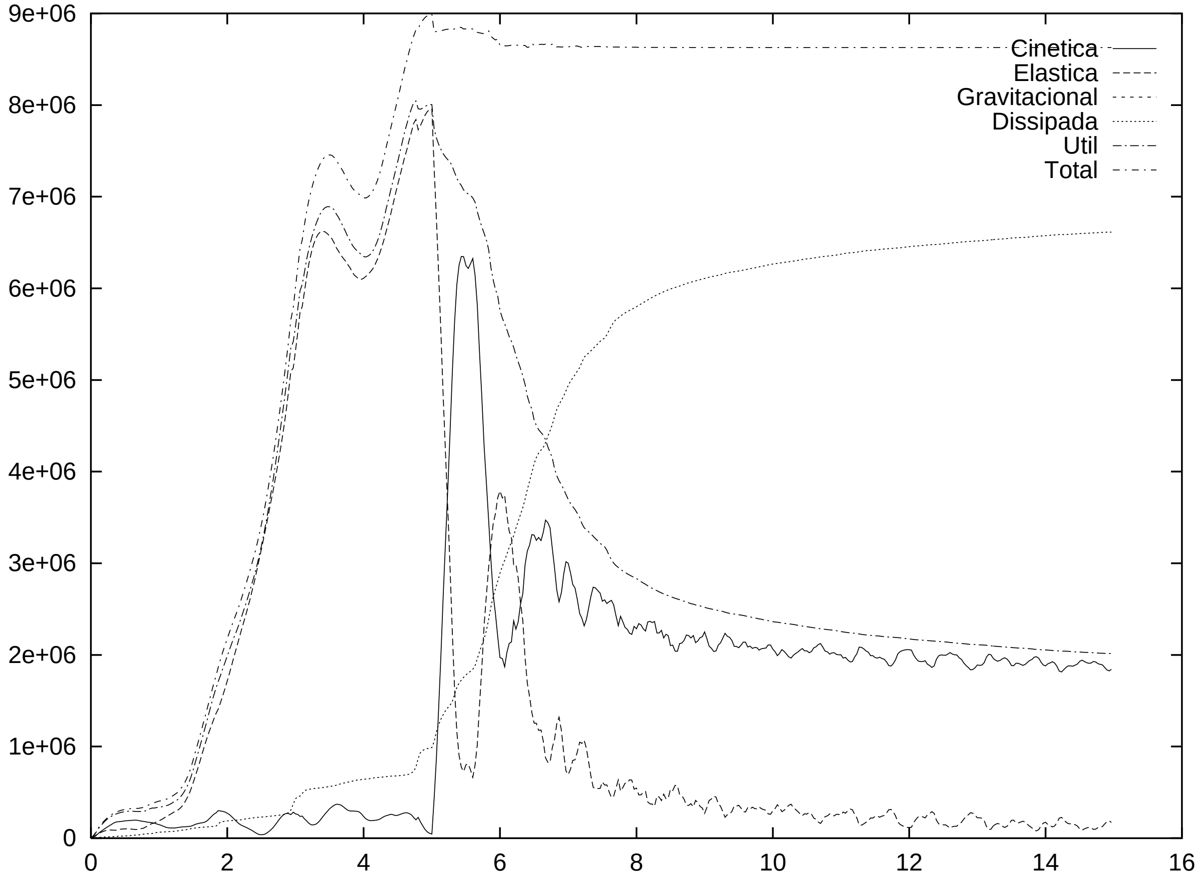


Figura 9.6: Gráficos de energia para a animação `loop`

9.4 Boneco

A animação `gizmo` mostra um boneco com quatro "pernas", que cai sobre um piso rígido e fixo (modelado como um tetraedro grande com os quatro vértices fixados). O modelo do boneco foi construído a partir de uma esfera e quatro cilindros, que foram colados à esfera. Como se pode observar nos gráficos de energia (figura (9.8)), o movimento do boneco permanece simétrico em relação ao eixo vertical, que coincide com a direção da força de gravidade.

Densidade (ρ)	1 `g/cm`^{3}
Módulo volumétrico (α)	10000 `dyna/cm`^{2}
Módulo de rigidez (β)	10000 `dyna/cm`^{2}
Coef. de viscosidade volumétrico (η_1)	1 `joule/s`
Coef. de viscosidade laminar (η_2)	1 `joule/s`
Número de tetraedros	659
Número de vértices do modelo	249
Número de vértices fixos	4
Número de vértices externos	204
Número de arestas externas	600
Número de faces externas	400
Números de pares a testar	257121
Aceleração da gravidade	327 `cm/s`^{2}
Tempo simulado	10 `s`
Tempo real (hh:mm:ss)	24:35:14
Razão tempo real/tempo simulado	8851

Tabela 9.4: Parâmetros e tempo de processamento da animação `gizmo`

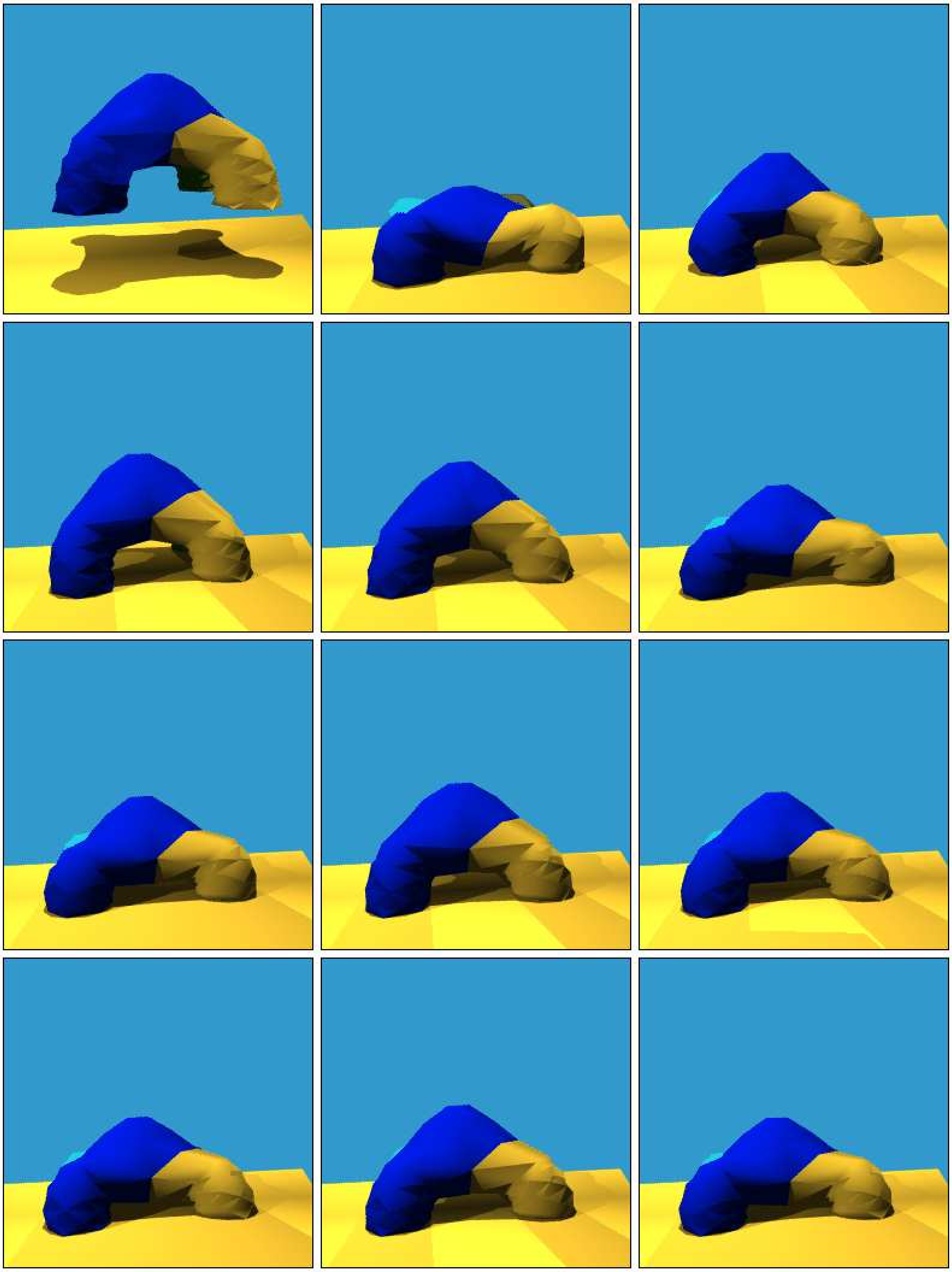


Figura 9.7: Quadros selecionados da animação `gizmo`

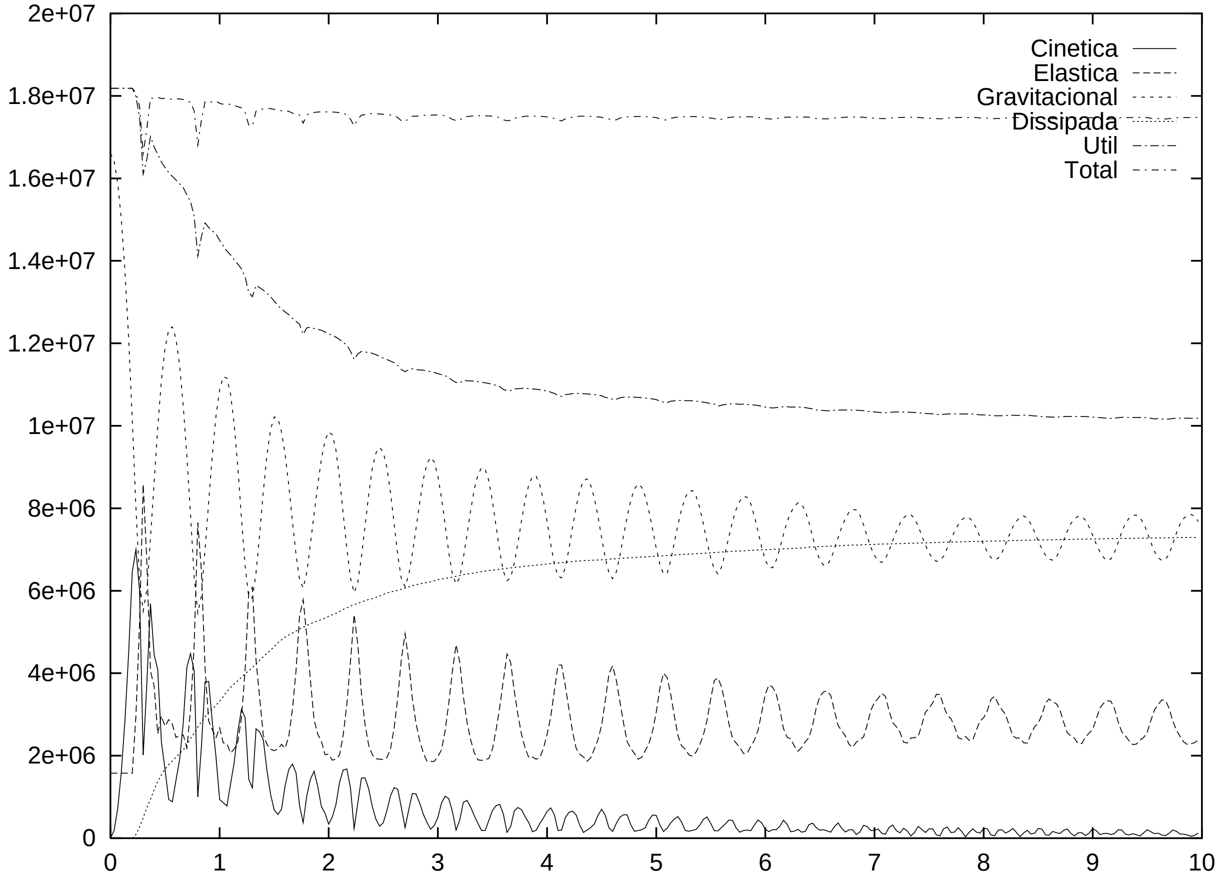


Figura 9.8: Gráficos de energia para a animação `gizmo`

9.5 Sumário

	`torus`	`elast`	`loop`	`gizmo`
Número de tetraedros	480	836	906	659
Número de vértices do modelo	180	283	353	249
Número de pares a testar	0	329339	639101	257121
Tempo simulado (`s`)	10	10	15	10
Tempo real (hh:mm:ss)	03:36:11	19:08:28	36:32:14	24:35:14
Razão tempo real/simulado	1297	6891	8769	8851

Tabela 9.5: Sumário das animações

Capítulo 10

Conclusão

10.1 Avaliação

O sistema de animação que implementamos mostrou-se capaz de gerar animações com objetos razoavelmente complexos (cerca de 1000 tetraedros) e colisões (cerca de 640000 pares vértice-face e aresta-aresta).

Os movimentos obtidos parecem bastante realistas. Globalmente, os corpos reagem da maneira esperada às deformações sofridas. Em animações com bolas caindo sobre um chão rígido, por exemplo, verifica-se que elas quicam da maneira esperada. Nota-se também as vibrações elásticas do material em torno de sua forma original de repouso. Nas colisões, o material se deforma localmente.

A função de energia elástica que desenvolvemos atingiu seus objetivos de ser insensível a movimentos rígidos e de não permitir compressões a volume zero dos tetraedros.

Embora sempre haja interpenetração após a colocação de uma mola, os corpos geralmente acabam por se separar logo depois, mesmo quando a colisão tem muita energia e a interpenetração é maior.

Esta experiência confirma que a produção de animações de corpos deformáveis por meio de modelos físicos é uma abordagem eficaz e prática para o problema. Mais do que isto, para corpos complexos ela é a única abordagem viável, dado o grande número de graus de liberdade de um corpo deformável não trivial.

10.1.1 Custo de processamento

Por outro lado, o elevado tempo de processamento das animações que produzimos deixou claro que o alto custo computacional desta técnica é um sério impedimento a seu uso. É praticamente certo, contudo, que os métodos numéricos e algoritmos que utilizamos possam ser aperfeiçoados para melhorar a eficiência do sistema.

10.1.2 Controle da animação

Outro problema intrínseco à animação dinâmica, que justifica em parte o seu uso limitado na área comercial, é a dificuldade de controlar o movimento sem perder o realismo físico. Enquanto que na animação puramente cinemática o animador tem total controle sobre os objetos, na animação dinâmica ele pode controlar diretamente apenas alguns poucos parâmetros do modelo. Em conseqüência, o animador pode ter de fazer várias tentativas — experimentando com outros parâmetros, forças externas, restrições, etc. — até obter o movimento desejado.

10.1.3 Colisões e contatos

A solução das molas de contato para o problema da interpenetração ainda permite que os corpos rolem, mas não permite que eles deslizem. Isto faz com que os corpos pareçam "grudentos". Além disso, como essa solução permite interpenetrações, a suposição fundamental que utilizamos na deteção de colisões, de que não há interpenetração no estado inicial a cada passo da simulação, fica falsa.

Os critérios que utilizamos para a retirada das molas algumas vezes falham. Por exemplo, se um vértice que penetrou uma face sai por uma face vizinha, mas permanece abaixo do plano da primeira face, a mola não é retirada. Situação análoga pode ocorrer entre pares aresta-aresta. Este defeito faz com que algumas vezes as molas sejam mantidas muito depois que os corpos se separaram, ou ainda, que os corpos fiquem presos em contato.

10.2 Extensões e aperfeiçoamentos

10.2.1 Modelagem geométrica

A diferença entre um modelo do material com massa distribuída e um modelo com massa concentrada nos vértices ainda não é clara, do ponto de vista da precisão e qualidade dos movimentos obtidos. Nem tampouco sabemos se o ganho de eficiência, devido ao uso de uma matriz de massa *diagonal* em vez de apenas esparsa, seria realmente significativo. Uma análise cuidadosa desta questão provavelmente seria um bom ponto de partida para trabalhos futuros.

Um tópico promissor para a extensão de nosso modelo é o uso de elementos finitos sujeitos a deformações não lineares. Por exemplo, poderíamos utilizar tetraedros modelados por retalhos de Bézier, onde haveria pontos de controle no interior das faces, além de apenas nos vértices. Isto permitiria reduzir consideravelmente o número de tetraedros de um corpo, mas complicaria a deteção e o tratamento de colisões, pois os corpos teriam superfícies curvas.

10.2.2 Paralelização

Processamento paralelo pode ser empregado para aumentar significativamente a velocidade do simulador, uma vez que os cálculos realizados em cada passo de integração podem ser facilmente distribuídos em vários processadores — por exemplo, as forças internas em cada tetraedro podem ser calculadas por um processador independente.

10.2.3 Modelagem dos materiais

É possível estender sem grandes dificuldades a expressão de energia elástica derivada neste trabalho para materiais elásticos não isotrópicos. Para modelar materiais plásticos, seria preciso evoluir ao longo do tempo a forma de repouso de cada elemento, conforme a amplitude das deformações.

10.2.4 Colisões e eventos discretos em geral

Com certeza, o principal tópico a requerer atenção, com relação ao nosso trabalho, é o da interpenetração entre os objetos. Quanto a este problema, poderíamos indicar o desenvolvimento do modelo geral de eventos discretos que introduzimos no capítulo 7. Lá, não especificamos completamente como se detetam eventos associados a regimes descritos por predicados B arbitrariamente complexos, com condições possivelmente redundantes ou conflitantes, nem tampouco descrevemos como se escolhe um novo regime da maneira mais eficiente. Tal modelo completo de eventos discretos poderia ser aplicado aos resultados de Baraff sobre o cálculo das forças de contato e atrito entre múltiplos pontos de contato.

Apêndice A

Tratamento de colisões por impulsos

Nossa abordagem inicial para o tratamento das colisões foi a de aplicar impulsos aos pontos de contato no instante de cada colisão, alterando instantaneamente as velocidades dos vértices do modelo. Essa solução foi implementada, e algumas animações bem sucedidas, com grande quantidade de colisões (e impulsos aplicados), foram produzidas. Porém, devido às dificuldades encontradas para simular contatos não instantâneos, que surgiam quando os objetos entravam em repouso mantendo contato, decidimos finalmente utilizar molas, como descrito no capítulo 7. Assim, o material já escrito sobre impulsos foi relegado a este apêndice.

Para simular a colisão de dois corpos perfeitamente rígidos, é necessário modificar instantaneamente a velocidade de pelo menos um deles, a fim de impedir a interpenetração imediatamente após a colisão. Essa variação de velocidade é devida ao surgimento de uma força instantânea e infinita — um *impulso* — que age sobre os dois corpos nos pontos de contato. Para a simulação correta da colisão, é necessário calcular este impulso.

Na literatura de computação gráfica, vários trabalhos sobre colisões de corpos rígidos [MW88, Hah88, MC94] descrevem o cálculo de impulsos para colisões que acontecem uma de cada vez, desconsiderando quaisquer outros pontos de contato entre os corpos no instante da colisão. Nesses trabalhos, o comportamento de corpos em contato contínuo (isto é, corpos que mantêm contato durante intervalos finitos de tempo) é modelado através de uma seqüência de colisões separadas no tempo por curtos intervalos. O problema com essa abordagem é que, se o intervalo de tempo entre colisões não for limitado a um valor mínimo arbitrário, o simulador poderá diminuí-lo (enquanto os corpos mantiverem contato) até um valor arbitrariamente pequeno em iterações sucessivas, impedindo o progresso da simulação e aumentando o acúmulo de erros de arredondamento. Por outro lado, se tal intervalo mínimo for imposto, interpenetrações possivelmente excessivas terão de ser permitidas.

Uma abordagem mais elaborada, descrita por D. Baraff numa série de artigos [Bar89, Bar91, BW92, Bar93, Bar94], consiste em calcular simultaneamente o conjunto dos impulsos no instante de cada colisão, e as forças de reação nos instantes seguintes. Tal abordagem, porém, torna-se bastante complexa quando há vários pontos de contato no instante de uma colisão, ou quando dois ou mais corpos mantêm simultaneamente um contato prolongado entre vários pares de pontos de suas superfícies. Na verdade, o problema de calcular os impulsos e/ou as forças de contato, levando-se em conta o atrito entre as superfícies, ainda está essencialmente em aberto.

As duas abordagens citadas acima podem também ser aplicadas a corpos deformáveis [BW92].

A.1 Impulsos

Um *impulso* é equivalente a uma força infinita $\mathbf{f}$ atuando num intervalo de tempo infinitesimal em torno de um instante t_0, tal que a integral

$$\mathbf{I} = \lim_{\tau \to 0} \int_{t_0-\tau}^{t_0+\tau} \mathbf{f}\, dt$$

é finita. A quantidade $\mathbf{I}$ é o valor do impulso.

Um impulso é capaz de mudar instantaneamente as velocidades das partículas. Note porém que a velocidade permanece finita, mesmo durante o impulso, e portanto as coordenadas das partículas permanecem inalteradas antes e depois do impulso. Integrando a equação diferencial $\mathbf{f} = \mathbf{M}\,\ddot{\mathbf{q}}$ em um intervalo infinitesimal em torno de t_0, temos que

$$\mathbf{I} = \mathbf{M}\,\Delta\dot{\mathbf{q}}$$

onde $\Delta\dot{\mathbf{q}}$ é a variação instantânea das velocidades generalizadas no instante t_0, em decorrência do *impulso generalizado* $\mathbf{I}$.

Seja $I = (t_0 - \tau, t_0 + \tau)$ o intervalo (infinitesimal) de duração de um impulso $\mathbf{I}$, para algum instante t_0. Quaisquer forças finitas que atuem ao longo desse intervalo, por exemplo a força da gravidade, são negligíveis. Suponha, porém, que o sistema dinâmico está sujeito a um ou mais vínculos holonômicos ao longo do intervalo I. Nesse caso, a derivada $\dot{h}$ de cada desvio h associado a um dos vínculos deve permanecer constante em I. Para tanto, algumas das respectivas forças de vínculo devem se tornar infinitas. Portanto, devemos calcular *impulsos de vínculo* no instante t_0, de modo que as variações instantâneas de velocidade causadas pelo impulso $\mathbf{I}$ não alterem os valores das derivadas $\dot{h}$ nesse instante.

A.2 Cálculo dos impulsos

Dado um conjunto de k vínculos $H_i(\mathbf{q}, t) = 0$, as derivadas das funções h_i tal que $h_i(t) = H_i(\mathbf{q}(t), t)$, para $i = 1, \ldots, k$, podem ser expressas na forma matricial $\mathbf{J}\dot{\mathbf{q}} + \boldsymbol{\psi}$, onde $J_{ij} = \frac{\partial H_i}{\partial q_j}(\mathbf{q}, t)$ e $\psi_i = \frac{\partial H_i}{\partial t}(\mathbf{q}, t)$, para $i = 1, \ldots, k$ e $j = 1, \ldots, n$. Se um impulso $\mathbf{I}$ é aplicado ao sistema num instante t_0, as variações instantâneas de velocidade $\Delta\dot{\mathbf{q}}$ devem ser tais que

$$\mathbf{J}\dot{\mathbf{q}} = \mathbf{J}(\dot{\mathbf{q}} + \Delta\dot{\mathbf{q}})$$

Portanto, dado um impulso generalizado $\mathbf{I}^{\mathrm{d}}$, podemos calcular as variações de velocidade $\Delta\dot{\mathbf{q}}$ que preservam os vínculos resolvendo o sistema

$$\begin{bmatrix} \mathbf{M} & \mathbf{J}^{\top} \\ \mathbf{J} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\dot{\mathbf{q}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{I}^{\mathrm{d}} \\ \mathbf{0} \end{bmatrix}$$

Note que neste caso $-\mathbf{J}^{\top}\boldsymbol{\lambda}$ é o vetor de impulsos de vínculo.

A.3 Cálculo dos impulsos de colisão

O que ocorre na natureza durante uma colisão entre dois corpos pode ser dividido em duas fases: uma fase de compressão, na qual os corpos se deformam localmente, transformando parte de sua energia cinética em energia potencial; e uma fase de restituição, na qual a energia potencial armazenada é reconvertida total ou parcialmente em energia cinética (se parcialmente, o restante é perdido em forma de calor). A parcela da energia cinética que é perdida durante a colisão depende das propriedades do material nas vizinhanças dos pontos de contato.

No caso de corpos rígidos, essas propriedades geralmente são modeladas através de um *coeficiente de restituição* e (calculado a partir do respectivo coeficiente para cada ponto de contato), que dá a razão entre as velocidades relativas na direção normal às superfícies nos pontos de contato antes e depois da colisão. Se $e = 1$, nenhuma energia cinética é perdida na colisão e a velocidade relativa normal antes da colisão é simplesmente invertida durante a colisão. Se $e = 0$, a velocidade relativa normal após a colisão é nula.

Por outro lado, no caso de corpos elásticos com viscosidade, a perda de energia pode ser modelada pelos coeficientes de viscosidade do material nas vizinhanças dos pontos de contato. De fato, se dividíssemos o material ao redor de cada ponto de contato em mais e mais tetraedros, teríamos, no limite em que o diâmetro dos tetraedros tendesse a zero, a situação de um corpo contínuo cuja viscosidade local modela com grande precisão o que acontece na realidade. Em nossa implementação, o impulso aplicado aos pontos de contato é calculado de modo a anular a velocidade relativa normal após a colisão, para que o máximo possível de energia cinética seja perdida. A eventual restituição dessa energia perdida depende da viscosidade dos corpos envolvidos na colisão.

No restante deste apêndice vamos desenvolver o cálculo do impulso de colisão, incluindo os coeficientes de restituição. Trataremos primeiro de impulsos sem atrito, e depois adicionaremos as forças de atrito estático e dinâmico ao cálculo do impulso.

A.3.1 Geometria dos pontos de contato

Suponha que A e B são dois corpos cujas superfícies se tocam num ponto de contato, sendo descritas por funções contínuas e diferenciáveis nas vizinhanças desse ponto. Sejam u e v as partículas das superfícies de A e B, respectivamente, que se tocam no ponto de contato, e $\mathbf{a}$ e $\mathbf{b}$ as posições de u e v na configuração corrente, respectivamente. Veja a figura (A.1).

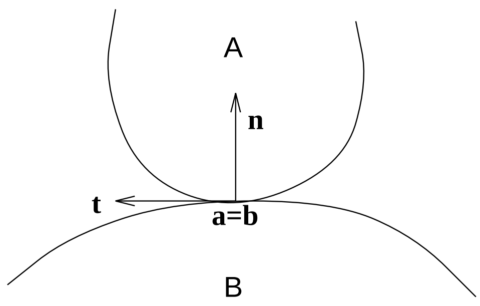


Figura A.1: Geometria dos pontos de contato

Se as superfícies de A e B não se cruzam nas vizinhanças do ponto de contato, então os vetores normais às mesmas em $\mathbf{a}$ e em $\mathbf{b}$ têm a mesma direção e sentidos opostos. No restante deste apêndice denotaremos por $\mathbf{n}$ o vetor unitário perpendicular às superfícies de A e B nos pontos $\mathbf{a}$ e $\mathbf{b}$ quando $\mathbf{a} = \mathbf{b}$, sendo que $\mathbf{n}$ aponta de B para A, e por $\mathbf{t}$ um vetor unitário tangente às superfícies de A e B no ponto de contato, isto é, um vetor unitário qualquer ortogonal a $\mathbf{n}$.

No caso de corpos com superfícies poliédricas estes vetores obviamente não são bem definidos em todo ponto, mas apenas nos pontos interiores das faces planas de cada superfície. Assim, se um contato envolve um vértice e uma face, o vetor $\mathbf{n}$ pode ser tomado como a normal unitária da face, e o vetor $\mathbf{t}$ como o vetor (normalizado) que vai desse vértice a um dos três vértices daquela face. Se um contato envolve duas arestas, podemos tomar $\mathbf{n}$ como um vetor unitário ortogonal às duas arestas, e $\mathbf{t}$ como um vetor unitário paralelo a uma das arestas.

A.3.2 Cálculo do impulso de colisão

Suponha que os corpos A e B colidem num instante t_c, e que as superfícies de contato neste instante se reduzem a uma partícula u de posição $\mathbf{a}$ no corpo A e uma partícula v de posição $\mathbf{b}$ no corpo B. Note que A e B podem ser o mesmo corpo. No instante t_c, $\mathbf{a} = \mathbf{b}$ e a velocidade relativa $\mathbf{v} = \dot{\mathbf{a}} - \dot{\mathbf{b}}$ entre u e v tem uma componente não nula na direção do vetor normal $\mathbf{n}$. Seja $\mathbf{u} = (\mathbf{q}, \dot{\mathbf{q}})$ o estado do sistema no instante t_c, antes da colisão ser tratada. Após a aplicação de um impulso de colisão $\mathbf{I}$, o estado será $\mathbf{u}' = (\mathbf{q}, \dot{\mathbf{q}} + \Delta\dot{\mathbf{q}})$, onde $\Delta\dot{\mathbf{q}}$ é a variação instantânea nas velocidades generalizadas.

Se $\mathbf{I}_u$ e $\mathbf{I}_v$ são os impulsos (em $\mathbb{R}^3$) sobre as partículas u e v, tal que $\mathbf{I}_u$ aponta do corpo B para o corpo A, então pela lei da ação e reação temos que $\mathbf{I}_u = -\mathbf{I}_v$. Veja a figura (A.2). Além disso, a energia cinética total dos dois corpos antes e depois da colisão deve ser a mesma, a menos da perda de energia, que é controlada pelo coeficiente de restituição e. Sejam e_u e e_v os coeficientes de restituição de u e u, respectivamente. Vamos definir arbitrariamente o coeficiente de restituição combinado e como $e = \frac{e_u+e_v}{2}$.

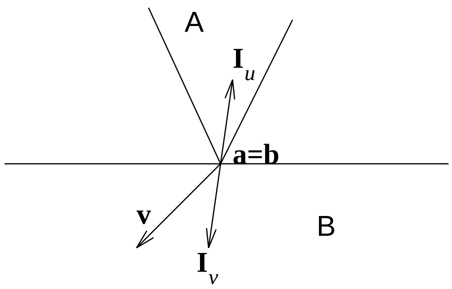


Figura A.2: Impulsos de colisão

Para calcular o impulso $\mathbf{I}$, adicionamos um vínculo instantâneo (que só existe no instante t_c) dado por uma função $H_n(\mathbf{q})$ que é positiva quando os corpos A e B estão afastados e nula quando eles se tocam num par de pontos de superfície. O desvio do vínculo $H_n(\mathbf{q}) = 0$ até o instante t_c é dado pela função h_n, enquanto que a partir de t_c, pela função h'_n. No instante t_c, temos que $h_n(t_c) = h'_n(t_c) = 0$ e $\dot{h}_n(t_c) < 0$. A partir de t_c, o desvio h'_n deve permanecer nulo ou tornar-se positivo, caso contrário haveria interpenetração. Portanto, $\dot{h}'_n(t_c) \geq 0$. Por definição, o coeficiente de restituição e é

$$e = -\frac{\dot{h}'_n(t_c)}{\dot{h}_n(t_c)}$$

A função H_n pode ser definida como

$$H_n(\mathbf{q}) = (\mathbf{a} - \mathbf{b}) \cdot \mathbf{n}$$

Note que no instante t_c e antes da aplicação dos impulsos,

$$\dot{h}_n(t_c) = (\dot{\mathbf{a}} - \dot{\mathbf{b}}) \cdot \mathbf{n} < 0$$

como era esperado. Note também que no caso de um impulso sem atrito, $\mathbf{I}_u = \lambda\mathbf{n}$ e $\mathbf{I}_v = -\lambda\mathbf{n}$, para algum escalar $\lambda > 0$.

Vamos supor que pelo menos no instante t_c todos os demais vínculos do sistema são holonômicos e válidos. Seja $\mathbf{J}_c$ a matriz “jacobiana” dos vínculos no instante t_c, ou seja, a matriz usual $\mathbf{J}$ com uma linha adicional para as derivadas de H_n. Imediatamente antes de t_c, o vetor $\mathbf{v} = \mathbf{J}_c\,\dot{\mathbf{q}}$ é $\mathbf{v} = [0, \ldots, 0, \dot{h}_n(t_c)]$. Como $\mathbf{J}_c$ é a mesma tanto antes quanto após a aplicação do impulso, podemos escrever

$$\mathbf{J}_c\,(\dot{\mathbf{q}} + \Delta\dot{\mathbf{q}}) = \mathbf{v} + \mathbf{J}_c\,\Delta\dot{\mathbf{q}} = -e\,\mathbf{v} \tag{A.1}$$

e então temos

$$\mathbf{J}_c\,\Delta\dot{\mathbf{q}} = -(e+1)\,\mathbf{v}$$

Tomando $\mathbf{I} = -\mathbf{J}_c^\top\boldsymbol{\lambda}$, obtemos o sistema

$$\begin{bmatrix} \mathbf{M} & \mathbf{J}_c^\top \\ \mathbf{J}_c & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\dot{\mathbf{q}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ -(e+1)\,\mathbf{v} \end{bmatrix}$$

que pode ser reescrito na forma

$$\begin{cases} \mathbf{J}_c\mathbf{M}^{-1}\mathbf{J}_c^\top\boldsymbol{\lambda} & = & (e+1)\,\mathbf{v} \\ \mathbf{M}\Delta\dot{\mathbf{q}} & = & -\mathbf{J}_c^\top\boldsymbol{\lambda} \end{cases} \tag{A.2}$$

A variação de energia cinética $\Delta K \leq 0$ na colisão é dada por

$$2\,\Delta K = (\dot{\mathbf{q}} + \Delta\dot{\mathbf{q}})^\top\mathbf{M}(\dot{\mathbf{q}} + \Delta\dot{\mathbf{q}}) - \dot{\mathbf{q}}^\top\mathbf{M}\dot{\mathbf{q}}$$

Substituindo $\Delta\dot{\mathbf{q}}$ por $-\mathbf{M}^{-1}\mathbf{J}_c^\top\boldsymbol{\lambda}$ temos

$$2\,\Delta K = -2\,\boldsymbol{\lambda}^\top\mathbf{J}_c\dot{\mathbf{q}} + \boldsymbol{\lambda}^\top\mathbf{J}_c\mathbf{M}^{-1}\mathbf{J}_c^\top\boldsymbol{\lambda}$$

e de (A.1) e (A.2) vem que

$$\Delta K = \frac{e-1}{2}\,\lambda_n\,\dot{h}_n(t_c)$$

onde $\lambda_n < 0$ é o multiplicador de Lagrange correspondente a H_n. Daí vemos que não há perda de energia somente quando $e = 1$ ou quando $\lambda_n = 0$ (isto é, se a colisão fosse ignorada).

Note que poderíamos facilmente generalizar esta formulação para permitir vínculos holonômicos não válidos em t_c, simplesmente substituindo os zeros no vetor $\mathbf{v}$ pelos valores das derivadas das respectivas funções de desvio, e trazendo e para dentro de $\mathbf{v}$. Poderíamos também estendê-la para vínculos não holonômicos descritos por equações $H(\mathbf{q}, \dot{\mathbf{q}}, t) = 0$ onde H é linear nas velocidades generalizadas $\dot{q}_i$. Finalmente, poderíamos também tratar várias colisões simultâneas: bastaria acrescentar uma linha à matriz $\mathbf{J}_c$ e uma componente incluindo o correspondente coeficiente de restituição ao vetor $\mathbf{v}$, para a função H_n de cada par de pontos de contato.

A.3.3 Propriedades da superfície

Seja U o conjunto (contínuo) de partículas do sistema e $S \subset U$ a superfície (ou fronteira) do mesmo. Para cada partícula $u \in S$ vamos associar coeficientes de atrito $\mu_s(u)$ e $\mu_d(u)$, ambos contidos no intervalo $[0, 1]$, que descrevem a rugosidade da superfície nas vizinhanças de u. No modelo de Coulomb para o atrito entre superfícies secas, estes coeficientes afetam a magnitude da força de atrito estático (que tenta impedir o início do deslizamento) e da força de atrito dinâmico (que tenta parar o deslizamento), respectivamente. Uma superfície seca absolutamente lisa tem coeficientes de atrito nulos. O coeficiente de atrito estático μ_s é usualmente maior que o coeficiente de atrito dinâmico μ_d, devido ao fato de que na maioria dos casos a força necessária para que um corpo em repouso sobre outro comece a deslizar é maior que a força necessária para manter esse corpo deslizando sobre o outro.

O coeficiente de restituição $e(u)$ de cada partícula $u \in S$ é uma constante adimensional no intervalo $[0, 1]$, que controla quanta energia é perdida durante uma colisão ocorrida nas vizinhanças de u.

Para cada face externa do modelo é dado o coeficiente de atrito estático μ_s, o coeficiente de atrito dinâmico μ_d e o coeficiente de restituição e. Os coeficientes de atrito e restituição dos vértices e arestas podem ser obtidos dos respectivos coeficientes das faces (a partir de médias aritméticas, por exemplo).

A.3.4 Colisões com atrito

Para simular o atrito entre as superfícies dos corpos A e B nos pontos de contato vamos utilizar o modelo de Coulomb [Sym82] para atrito entre superfícies secas, de maneira similar a trabalhos anteriores [MW88, Hah88].

Seja $\mathbf{v} = \dot{\mathbf{a}} - \dot{\mathbf{b}}$ a velocidade relativa dos pontos de contato (isto é, as partículas u e v tais que $\mathbf{a} = \mathbf{b}$) no instante t_c. A componente tangencial de $\mathbf{v}$ (ortogonal a $\mathbf{n}$) é dada por $\mathbf{v}_t = \mathbf{v} - (\mathbf{v} \cdot \mathbf{n})\mathbf{n}$. Seja $\mathbf{f}_u = \mathbf{f}_u^{\mathrm{n}} + \mathbf{f}_u^{\mathrm{t}}$ a da força de contato sobre a partícula u, onde $\mathbf{f}_u^{\mathrm{n}}$ e $\mathbf{f}_u^{\mathrm{t}}$ são suas componentes nas direções de $\mathbf{n}$ e $\mathbf{t}$, respectivamente. A componente normal $\mathbf{f}_u^{\mathrm{n}}$ é responsável por impedir a interpenetração nos pontos de contato, enquanto que a componente tangencial $\mathbf{f}_u^{\mathrm{t}}$ é a força de atrito entre as superfícies de A e B naqueles pontos de contato. O modelo de Coulomb diz que se $\mathbf{v}_t$ é nula então $\mathbf{f}_u^{\mathrm{t}}$ é uma força de atrito estático cuja magnitude obedece à relação

$$|\mathbf{f}_u^{\mathrm{t}}| \leq \mu_s \, |\mathbf{f}_u^{\mathrm{n}}| \tag{A.3}$$

Note que a direção de $\mathbf{f}_u^{\mathrm{t}}$ no plano de contato não é especificada por esta relação. Quando $\mathbf{v}_t$ não é nula, $\mathbf{f}_u^{\mathrm{t}}$ uma força de atrito dinâmico dada por

$$\mathbf{f}_u^{\mathrm{t}} = -\mu_d \, |\mathbf{f}_u^{\mathrm{n}}| \frac{\mathbf{v}_t}{|\mathbf{v}_t|} \tag{A.4}$$

Quando aplicamos um impulso instantâneo aos pontos de contato estamos desconsiderando toda variação de velocidade ou forças de contato durante o intervalo infinitesimal de tempo que dura o impulso. Portanto, não podemos saber *a priori* se o atrito que ocorre durante o impulso é estático ou dinâmico, ou quando muda de um tipo para outro durante o impulso. Assim, vamos aproximar o atrito durante uma colisão calculando dois impulsos. O primeiro é calculado sob a hipótese de que o atrito é estático, e portanto a velocidade relativa tangencial após a colisão deve ser nula. O segundo impulso é calculado sob a hipótese de que o atrito é dinâmico, o que significa que os corpos estarão deslizando nos pontos de contato imediatamente após a colisão.

Para calcular o impulso com atrito estático, utilizamos três vínculos instantâneos cujas funções são definidas como

$$\begin{aligned} H_n(\mathbf{q}) &= (\mathbf{a}-\mathbf{b})\cdot\mathbf{n} \\ H_u(\mathbf{q}) &= (\mathbf{a}-\mathbf{b})\cdot\mathbf{u} \\ H_t(\mathbf{q}) &= (\mathbf{a}-\mathbf{b})\cdot\mathbf{t} \end{aligned}$$

onde $\mathbf{u}$ e $\mathbf{t}$ são dois vetores unitários ortogonais a $\mathbf{n}$ e um ao outro, sendo que $\mathbf{t}$ é um vetor ortogonal a $\mathbf{n}$ qualquer e $\mathbf{u} = \mathbf{n} \times \mathbf{t}$. Sejam h'_n, h'_u e h'_t as respectivas funções de desvio desses três vínculos, definidas a partir do instante t_c após a aplicação do impulso de colisão. Se h_n é o desvio do vínculo $H_n(\mathbf{q}) = 0$ até o instante t_c, então as derivadas destas funções no instante t_c são dadas por

$$\begin{aligned} \dot{h}'_n(t_c) &= -e\,\dot{h}_n(t_c) \\ \dot{h}'_u(t_c) &= 0 \\ \dot{h}'_t(t_c) &= 0 \end{aligned}$$

Inserindo estas relações no sistema de equações, obtemos o impulso $\mathbf{I}$, de modo similar ao descrito na seção anterior.

Para verificar a hipótese do atrito estático precisamos traduzir os impulsos $\mathbf{I}$ sobre as coordenadas generalizadas em impulsos sobre as coordenadas cartesianas das partículas u e v. Para fazer isto de modo geral, observe que os impulsos sobre os vértices são trivialmente obtidos de $\mathbf{I}$, pegando as triplas de elementos de $\mathbf{I}$ que correspondem a cada vértice. Em seguida, pode-se calcular os impulsos sobre as partículas que não estão nos vértices como combinações lineares convexas (via coordenadas baricêntricas) dos impulsos nos vértices.

Seja $\mathbf{I}_u \in \mathbb{R}^3$ o impulso sobre a partícula u, obtido de $\mathbf{I}$ conforme descrito acima. Sejam $\mathbf{I}_u^{\mathrm{n}}$ e $\mathbf{I}_u^{\mathrm{t}}$ as componentes normal e tangencial de $\mathbf{I}_u$, calculadas como $\mathbf{I}_u^{\mathrm{n}} = (\mathbf{I}_u \cdot \mathbf{n})\mathbf{n}$ e $\mathbf{I}_u^{\mathrm{t}} = \mathbf{I}_u - \mathbf{I}_u^{\mathrm{n}}$. Se $|\mathbf{I}_u^{\mathrm{t}}| \geq \mu_s |\mathbf{I}_u^{\mathrm{n}}|$, ou de modo equivalente, se

$$\mathbf{I}_u^2 \geq (\mathbf{I}_u \cdot \mathbf{n})^2 (1 + \mu_s^2)$$

então a força de atrito estático nos pontos de contato durante a colisão não é suficiente para impedir que os corpos comecem a deslizar naqueles pontos. Portanto, vamos recalcular o impulso considerando que o atrito é dinâmico e que os corpos irão deslizar nos pontos de contato imediatamente após a colisão.

Para embutir a equação (A.4) no cálculo do impulso $\mathbf{I}$ vamos decompô-lo numa soma $\mathbf{I} = -(\mathbf{J}_c + \mathbf{G}_c)^\top \boldsymbol{\lambda}$, onde $\mathbf{J}_c$ é a mesma que usamos no caso do impulso sem atrito, e $\mathbf{G}_c$ inclui os impulsos de atrito dinâmico sobre as partículas u e v dados por $\mathbf{g}_u(\mathbf{q}, \dot{\mathbf{q}}, \boldsymbol{\lambda}) = \mu_d\, \lambda_n \frac{\mathbf{v}}{|\mathbf{v}_t|}$ e $\mathbf{g}_v = -\mathbf{g}_u$, respectivamente, onde $\mathbf{v}_t$ é a velocidade relativa tangencial nos pontos de contato, como anteriormente calculada, e $\lambda_n < 0$ é a magnitude da componente normal do impulso $\mathbf{I}_u$. O sistema de equações para calcular este impulso é

$$\begin{bmatrix} \mathbf{M} & \mathbf{F}_c^\top \\ \mathbf{J}_c & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\dot{\mathbf{q}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ -(e+1)\,\mathbf{v} \end{bmatrix}$$

onde $\mathbf{F}_c = \mathbf{J}_c + \mathbf{G}_c$ e $\mathbf{v}$ é o mesmo que usamos para o impulso sem atrito.

Bibliografia

[AG85] William W. Armstrong e Mark W. Green. The dynamics of articulated rigid bodies for purposes of animation. In *Proceedings of Graphics Interface*, pp. 407–415. Computer Graphics Society, 1985.

[Bar89] David Baraff. Analytical methods for dynamic simulation of non-penetrating rigid bodies. *Computer Graphics (Proc. SIGGRAPH)*, 23(3):223–232, Julho de 1989.

[Bar90] David Baraff. Curved surfaces and coherence for non-penetrating rigid body simulation. *Computer Graphics (Proc. SIGGRAPH)*, 24(4):19–28, Agosto de 1990.

[Bar91] David Baraff. Coping with friction for non-penetrating rigid body simulation. *Computer Graphics (Proc. SIGGRAPH)*, 25(4):31–40, Julho de 1991.

[Bar93] David Baraff. Issues in computing contact forces for non-penetrating rigid bodies. *Algorithmica*, 10:292–352, 1993.

[Bar94] David Baraff. Fast contact force computation for non-penetrating rigid bodies. *Computer Graphics (Proc. SIGGRAPH)*, 28(3):23–34, Julho de 1994.

[Bar96] David Baraff. Linear-time dynamics using lagrange multipliers. Technical Report CMU-RI-TR-95-44, Carnegie Mellon University, 1996.

[BB88] Ronen Barzel e Alan H. Barr. A modeling system based on dynamic constraints. *Computer Graphics (Proc. SIGGRAPH)*, 22(4):179–188, Agosto de 1988.

[BD94] William E. Boyce e Richard D. DiPrima. *Equações Diferenciais Elementares e Problemas de Valores de Contorno*. Guanabara Koogan S.A., Rio de Janeiro, RJ, 5ª edição, 1994.

[BF89] R. Burden e J. Faires. *Numerical Analysis*. PWS-Kent, 4ª edição, 1989.

[BS83] Walter Baur e Volker Strassen. The complexity of partial derivatives. *Theoretical Computer Science*, 22:317–330, 1983.

[Bur87] David S. Burnett. *Finite Element Analysis/From concepts to applications.* Addison-Wesley, 1987.

[BW92] David Baraff e Andrew Witkin. Dynamic simulation of non-penetrating flexible bodies. *Computer Graphics (Proc. SIGGRAPH)*, 26(2):303–309, Julho de 1992.

[CS89] James F. Cremer e A. James Stewart. The architecture of newton, a general-purpose dynamics simulator. *IEEE Computer Graphics and Applications*, pp. 1806–1811, 1989.

[ES93] Margaret A. Ellis e Bjarne Stroustrup. *C++ Manual de Referência Comentado.* Editora Campus, 1993.

[FvDFH90] James D. Foley, Andries van Dam, Steven K. Feiner, e John F. Hughes. *Computer graphics/Principles and practice.* Addison-Wesley, 2ª edição, 1990.

[GL85] G. Golub e C. Van Loan. *Matrix Computations.* Johns Hopkins University Press, 1985.

[Gol80] Herbert Goldstein. *Classical Mechanics.* Addison-Wesley, 2ª edição, 1980.

[Hah88] James K. Hahn. Realistic animation of rigid bodies. *Computer Graphics (Proc. SIGGRAPH)*, 22(4):299–308, Agosto de 1988.

[HW95] Agnus Azevedo Horta e Shin-Ting Wu. Deformação de superfícies não-rígidas baseada em princípios físicos. In *Anais do VIII SIBGRAPI*, pp. 175–182, Outubro de 1995.

[IC87] Paul Isaacs e Michael F. Cohen. Controlling dynamic simulation with kinematic constraints, behavior functions and inverse dynamics. *Computer Graphics (Proc. SIGGRAPH)*, 21(4):215–224, Julho de 1987.

[LM94] Ming C. Lin e Dinesh Manocha. Efficient contact determination between geometric models. Technical report, University of North Carolina, Chapel Hill, NC, 1994.

[MC94] Brian Mirtich e John Canny. Impulse-based dynamic simulation. Technical report, Department of Computer Science,University of California, Berkeley, CA, 1994.

[ML94] Soraia Raupp Musse e Anatólio Laschuk. A model for animation and control of articulated figures using a simplified dynamics approach. In *Anais do VII SIBGRAPI*, pp. 117–123, Novembro de 1994.

[MT92] Dimitri Metaxas e Demetri Terzopoulos. Dynamic deformation of solid primitives with constraints. *Computer Graphics (Proc. SIGGRAPH)*, 26(2):309–312, Julho de 1992.

[MW88] Mathew Moore e Jane Wilhelms. Collision detection and response for computer animation. *Computer Graphics (Proc. SIGGRAPH)*, 22(4):289–298, Agosto de 1988.

[NDW94] Jackie Neider, Tom Davis, e Mason Woo. *OpenGL Programming Guide*. Addison-Wesley, 1994.

[Ned93] Luciana Porcher Nedel. Simulação de objetos deformáveis baseada na análise dinâmica. Master's thesis, Instituto de Informática II/UFRGS, Porto Alegre, RS, Brasil, 1993.

[Nel91] Greg Nelson, editor. *Systems programming with Modula-3*. Prentice Hall Series in Innovative Technology. Prentice Hall, 1991.

[PB88] John Platt e Alan H. Barr. Constraint methods for flexible bodies. *Computer Graphics (Proc. SIGGRAPH)*, 22(4):279–288, Agosto de 1988.

[Pis84] Sérgio Pissanetzky. *Sparse Matrix Technology*. Academic Press, Inc., 1984.

[PML95] Madhav K. Ponamgi, Dinesh Manocha, e Ming C. Lin. Incremental algorithms for collision detection between solid models. In *Proceedings of ACM/SIGGRAPH symposium on solid modeling*, pp. 293–304, 1995.

[PW89] Alex Pentland e J. Williams. Good vibrations: modal dynamics for graphics and animation. *Computer Graphics (Proc. SIGGRAPH)*, 23(3):215–222, Julho de 1989.

[Ra93] Maria Andréia F. Rodrigues e Léo P. Magalh aes. Animado: um protótipo de um sistema de animação modelada por dinâmica. In *Anais do VI SIBGRAPI*, pp. 107–115, Outubro de 1993.

[Sym82] Keith R. Symon. *Mecânica*. Editora Campus, Rio de Janeiro, RJ, 5ª edição, 1982.

[TF88] Demetri Terzopoulos e Kurt Fleischer. Modeling inelastic deformation: viscoelasticity, plasticity, fracture. *Computer Graphics (Proc. SIGGRAPH)*, 22(4):269–278, Agosto de 1988.

[TPBF87] Demetri Terzopoulos, John Platt, Alan H. Barr, e Kurt Fleischer. Elastically deformable models. *Computer Graphics (Proc. SIGGRAPH)*, 21(4):205–214, Julho de 1987.

[TW88] Demetri Terzopoulos e Andrew Witkin. Physically-based models with rigid and deformable components. In *Proceedings of Graphics Interface*, pp. 146–154. Computer Graphics Society, 1988.

[WB85] Jane Wilhelms e Brian A. Barsky. Using dynamic analysis to animate articulated bodies such as humans and robots. In *Proceedings of Graphics Interface*, pp. 97–105. Computer Graphics Society, 1985.

[WGW90] Andrew Witkin, Michael Gleicher, e William Welch. Interactive dynamics. *Computer Graphics*, 24(2):11–22, Março de 1990.

[WK88] Andrew Witkin e Michael Kass. Spacetime constraints. *Computer Graphics (Proc. SIGGRAPH)*, 22(4):159–168, Agosto de 1988.

[WW90] Andrew Witkin e William Welch. Fast animation and control of nonrigid structures. *Computer Graphics (Proc. SIGGRAPH)*, 24(4):243–250, Agosto de 1990.

Índice de símbolos

O número após a definição de cada símbolo é o número da página em que ele primeiro aparece na dissertação.

Letras latinas

$\mathbf{A}$ matriz de mapeamento da configuração de referência para a configuração de repouso de um tetraedro, 48
$\mathbf{B}$ matriz de mapeamento da configuração de referência para a configuração corrente de um tetraedro, 48
B predicado associado a um regime, 68
$\mathbf{C}$ matriz de mapeamento da configuração de repouso para a configuração corrente de um tetraedro, 48
$\mathbf{C}$ tensor de deformação, 32
$\dot{\mathbf{C}}$ tensor de escoamento, 43
$\mathbf{D}$ matriz das direções das forças de vínculo, 27
$\mathbf{D}$ matriz diagonal na decomposição dos valores singulares, 32
$\mathbf{D}$ matriz diagonal da decomposição de Choleski, 54
$\mathbf{d}$ direção de uma força de vínculo, 24
$\mathbf{d}_i$ direção da i-ésima força de vínculo, 25
e_p energia potencial, 14
E erro numérico estimado pelo integrador, 64
F função de cálculo das derivadas primeiras de $\mathbf{u}$, 64
$\mathcal{F}$ função que mede a variação de volume na fórmula de Φ, 38
$\mathcal{F}^*$ função que mede a variação de volume na fórmula de Φ^*, 35
$\mathbf{f}$ vetor de forças diretas, 17
$\mathbf{f}$ função de configuração do sistema, 29
$\mathbf{f}^c$ vetor de forças conservativas internas, 17
f_i^c força conservativa interna, 14

$\mathbf{f}^{\mathrm{d}}$ vetor de forças dissipativas internas, 17
f_i^{d} força dissipativa interna, 15
$\mathbf{f}^{\mathrm{e}}$ vetor de forças externas, 17
f_i^{e} força externa, 15
$\mathbf{f}^{\mathrm{f}}$ vetor de forças fictícias, 17
$\mathbf{f}^{\mathrm{n}}$ vetor de forças normais de contato, 26
$\mathbf{f}^{\mathrm{p}}$ vetor de forças de penalidade, 21
f_i^{p} força de penalidade, 22
$\mathbf{f}^{\mathrm{t}}$ vetor de forças de atrito dinâmico, 26
$\mathbf{f}^{\mathrm{v}}$ vetor de forças de vínculo, 24
$\mathbf{f}_i^{\mathrm{v}}$ vetor de forças da i-ésima força de vínculo, 24
f_i^{v} força de vínculo, 27
G função de condição associada a um regime, 69
$\mathcal{G}$ função que mede a variação de forma na fórmula de Φ, 38
$\mathcal{G}^*$ função que mede a variação de forma na fórmula de Φ^*, 35
H função de vínculo, 19
$\mathbb{H}_3^2$ espaço das funções quadráticas, homogêneas e simétricas de três variáveis, 35
h desvio de um vínculo, 22
I intervalo de tempo, 22
$\mathbf{I}$ matriz identidade, 34
$\mathbf{J}$ matriz jacobiana dos vínculos, 27
$\mathbb{J}_{\mathbf{f}}$ matriz jacobiana da configuração $\mathbf{f}$, 31
K função de energia cinética, 14
K constante das forças de penalidade, 22
k número de vínculos do sistema, 25
$\mathbf{L}$ matriz triangular inferior na fatoração de Choleski, 54
$\mathbf{M}$ matriz de massa generalizada, 16
$\mathbf{M}$ matriz de mapeamento da configuração de referência para outra configuração de um tetraedro, 47
n número de coordenadas generalizadas do sistema, 13
n_v número de vértices do modelo, 49
n_a número de arestas do modelo, 49
n_t número de elementos do modelo, 49
P função de energia potencial, 14
$\mathbf{p}$ configuração corrente, 31
p pressão uniforme aplicada sobre uma superfície, 35

Letras gregas

α	módulo volumétrico, 35
$\alpha_1, \ldots, \alpha_4$	coordenadas baricêntricas de uma partícula, 47
β	módulo de rigidez, 35
Δ	determinante do tensor métrico, 34
Γ	traço do tensor métrico, 34
η_1	coeficiente de viscosidade volumétrico, 44
η_2	coeficiente de viscosidade laminar, 44
$\boldsymbol{\lambda}$	vetor de multiplicadores de Lagrange, 27
λ_i	multiplicador de Lagrange, 25
$\lambda_1, \lambda_2, \lambda_3$	auto-valores do tensor métrico $\mathbf{T}$, 34
ρ	densidade de massa, 29
Φ	função da densidade de energia elástica, 33
Φ^*	função da densidade de energia elástica para pequenas deformações, 35
ϕ	densidade de energia elástica, 33
Π	traço de $\dot{\mathbf{T}}$, 45
μ	coeficiente de atrito dinâmico, 26
Σ	soma dos cofatores da diagonal do tensor métrico, 34
$\sigma_1, \sigma_2, \sigma_3$	elementos diagonais (valores singulares) da matriz $\mathbf{D}$ na decomposição dos valores singulares, 33
$\boldsymbol{\psi}$	vetor de componentes $\psi_1, \ldots, \psi_k$, 27
ψ_l	lado direito da expansão de $\ddot{h}_l(t) = 0$, 27
τ	tamanho do intervalo de tempo entre estados consecutivos calculados na simulação, 63
$\tau_{\max}$	valor máximo permitido a τ, 63
$\tau_{\min}$	valor mínimo permitido a τ, 63
Ω	potência dissipada, 43
ω	densidade de potência dissipada, 43
Ξ	soma dos cofatores da diagonal de $\dot{\mathbf{T}}$, 45
ξ	constante de estabilização de vínculo, 23
χ	polinômio característico do tensor métrico, 34