

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
INSTITUTO DE INFORMÁTICA  
CURSO DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**Simulação de Objetos  
Deformáveis Baseada na  
Análise Dinâmica**

por

Luciana Porcher Nedel

Dissertação submetida como requisito parcial  
para a obtenção do grau de  
Mestre em Ciência da Computação

Prof. Flávio Rech Wagner  
Orientador

Prof<sup>a</sup>. Carla Maria Dal Sasso Freitas  
Co-orientadora

Porto Alegre, 21 de Junho de 1993.

## CIP - CATALOGAÇÃO NA PUBLICAÇÃO

Nedel, Luciana Porcher

Simulação de objetos deformáveis baseada na análise dinâmica  
/ Luciana Porcher Nedel. – Porto Alegre: CPGCC da UFRGS, 1993.

151p.: il.

Dissertação (mestrado) - Universidade Federal do Rio Grande do Sul, Curso de Pós-Graduação em Ciência da Computação, Porto Alegre, 1993. Orientador: Wagner, Flávio Rech. Co-orientadora: Freitas, Carla M.D.S.

Dissertação: Animação, Animação Baseada em Física, Objetos Deformáveis

*"(...) toda ciência seria  
supérflua se a aparência, a forma  
das coisas, fosse totalmente  
idêntica à sua natureza"*

*Karl Marx*

## AGRADECIMENTOS

Gostaria de agradecer ao Niuton pela paciência, compreensão e principalmente por estar sempre ao meu lado. Aos meus pais, à Claudia e à Laura pela paciência e incentivo. Aos professores Carla e Flávio pela orientação e apoio durante o desenvolvimento deste trabalho. Ao professor Laschuk pelas valiosas contribuições durante o período de definição do modelo. Ao Pedro pelo auxílio na época da proposta de trabalho e ao Miguel pelas dicas e interesse.

Queria ainda agradecer à Soraia por compartilhar comigo dos problemas e das soluções encontradas durante todo o trabalho. À professora Silvia pelo incentivo e oportunidades proporcionadas. À Otilia pelo auxílio na edição do texto. A todos os funcionários do Instituto de Informática e do Centro de Supercomputação pela eficiência e boa vontade. Aos colegas do grupo de computação gráfica da UFRGS pelo apoio recebido e à CAPES e CNPq por financiarem parcialmente este trabalho.

Finalmente, gostaria de agradecer a todas essas pessoas por acreditarem no meu trabalho. Obrigada.

# SUMÁRIO

<b>LISTA DE FIGURAS .....</b>	<b>9</b>
<b>LISTA DE TABELAS.....</b>	<b>12</b>
<b>RESUMO .....</b>	<b>13</b>
<b>ABSTRACT .....</b>	<b>15</b>
<b>1 INTRODUÇÃO.....</b>	<b>16</b>
<b>1.1 Métodos de Controle do Movimento .....</b>	<b>18</b>
1.1.1 Controle explícito .....	18
1.1.2 Controle procedural .....	19
1.1.3 Sistemas baseados em restrições .....	20
1.1.4 Modelos gravados.....	21
1.1.5 Uso de atores.....	22
1.1.6 Cinemática e dinâmica .....	22
1.1.7 Animação baseada em leis físicas .....	23
<b>1.2 Modelagem Algorítmica .....</b>	<b>24</b>
<b>1.3 Modelos Baseados em Leis Físicas .....</b>	<b>27</b>
1.3.1 Tipos de objetos e seus movimentos.....	27
1.3.2 Análise dinâmica .....	30
1.3.3 Problemas envolvidos no uso da análise dinâmica.....	33
<b>1.4 Modelagem Teleológica.....</b>	<b>35</b>
<b>1.5 Restrições em Sistemas de Simulação Dinâmica.....</b>	<b>37</b>
1.5.1 Restrições cinemáticas.....	37
1.5.2 Restrições dinâmicas .....	38

1.5.3 Restrições espaço-temporais .....	39
<b>1.6 Objetivos e Organização do Trabalho.....</b>	<b>40</b>
<b>2 MODELOS DEFORMÁVEIS.....</b>	<b>44</b>
<b>2.1 Deformação Geométrica .....</b>	<b>45</b>
2.1.1 Deformação de primitivas sólidas .....	46
2.1.2 Deformação em formas livres (FFD) .....	50
2.1.3 Método baseado em restrições .....	54
2.1.4 Aplicações .....	56
<b>2.2 Deformação Baseada em Leis Físicas .....</b>	<b>56</b>
2.2.1 Deformação estática de superfícies por curvas catenárias.....	58
2.2.2 Modelo energético para deformação estática de superfícies .....	61
2.2.3 Modelo dinâmico de deformação.....	63
2.2.4 Método baseado em restrições .....	67
2.2.5 Modelo baseado em sistemas de partículas .....	69
2.2.6 Aplicações.....	71
<b>3 MODELO DE REPRESENTAÇÃO DE OBJETOS FLEXÍVEIS .....</b>	<b>73</b>
<b>3.1 Modelo Geométrico .....</b>	<b>74</b>
<b>3.2 Modelo Físico.....</b>	<b>75</b>
3.2.1 Força da gravidade .....	76
3.2.2 Vetor de força externa.....	77
3.2.3 Elasticidade.....	77
3.2.4 Força de curvatura e torção.....	79
3.2.5 Colisão .....	81
3.2.6 Restrições .....	85
3.2.7 Força de atrito.....	87

<b>4 SIMULAÇÃO DO MOVIMENTO</b> .....	88
4.1 Equação do Movimento de uma Partícula.....	88
4.2 Métodos Numéricos para EDO de Primeira Ordem .....	89
4.2.1 Método de Euler .....	90
4.2.2 Método de Runge-Kutta.....	91
4.2.3 Métodos de múltiplos passos .....	92
4.2.4 Métodos preditores-corretores .....	93
4.3 O Problema da Descontinuidade .....	94
4.4 Algoritmo de Geração do Movimento .....	95
<b>5 IMPLEMENTAÇÃO DO PROTÓTIPO FLEX3D</b> .....	102
5.1 Organização do Protótipo .....	102
5.2 Parametrização do Modelo.....	103
5.2.1 Sintaxe.....	104
5.2.2 Semântica.....	105
5.3 Estratégia de Detecção do Choque.....	108
5.4 Estruturas de Dados .....	109
5.5 Modelagem das Superfícies.....	111
5.6 Geração do Movimento .....	111
5.7 Geração dos Arquivos de Saída.....	112
5.8 Linguagem de Programação e Equipamentos Utilizados .....	113
5.9 Apresentação dos Resultados .....	114
<b>6 CONCLUSÕES</b> .....	115
6.1 Análise do Trabalho.....	115
6.2 Validação do Modelo.....	116
6.3 Perspectivas Futuras.....	120

<b>ANEXO A-1 FORMATO DO ARQUIVO DE DESCRIÇÃO</b>	
<b>GEOMÉTRICA DE OBJETOS.....</b>	<b>123</b>
<b>A1.1 Sintaxe .....</b>	<b>123</b>
<b>A1.2 Semântica.....</b>	<b>123</b>
<b>ANEXO A-2 EXEMPLOS DE SEQÜÊNCIAS ANIMADAS E CENAS</b>	
<b>GERADAS PELO FLEX3D.....</b>	<b>125</b>
<b>BIBLIOGRAFIA.....</b>	<b>140</b>



## LISTA DE FIGURAS

Figura 1.1	Graus de liberdade e de movimento para (a) partículas e (b) corpos rígidos [WIL 87a].	29
Figura 1.2	Graus de liberdade de movimento para um corpo flexível [WIL 87a].	29
Figura 1.3	Graus de liberdade de movimento para um corpo articulado.	31
Figura 1.4	Forças e torques que contribuem na geração do movimento.	32
Figura 1.5	Representações de objetos e conversões entre elas [BAR 88a].	37
Figura 2.1	Transformação de afinamento em uma faixa.	50
Figura 2.2	(a) retângulo transparente contendo quadrados e círculos; (b) retângulo transparente após sofrer o processo de deformação.	51
Figura 2.3	Sistema de coordenadas (s,t,u) [SED 86].	52
Figura 2.4	(a) pontos de controle sobre o retângulo; (b) posição dos pontos de controle, após a deformação.	53
Figura 2.5	(a) Pontos de restrição; (b) Triângulos formados a partir da conexão dos pontos; (c) e (d) Subdivisão dos triângulos [WEI 86].	59
Figura 2.6	(a) Ponto de restrição $P$ , seus 4 vizinhos ( $P_1, P_2, P_3, P_4$ ) e os vetores $\bar{v}_1, \bar{v}_2, \bar{v}_3, \bar{v}_4$ ; (b) Soma dos vetores ( $\bar{v}_1, \bar{v}_2, \bar{v}_3, \bar{v}_4$ ) a fim de encontrar a distância de repouso e conseqüentemente a nova posição de $P$ ( $P'$ ).	60
Figura 2.7	Componente elástica do modelo.	61
Figura 2.8	Representação geométrica de um corpo deformável [TER 88a].	66
Figura 2.9	Hierarquia dos métodos de restrição [PLA 88a].	68

Figura 2.10 (a) Malha que representa a superfície deformável; (b) Representação geométrica através de uma grade de partículas [HOU 92].	70
Figura 2.11 Funções energéticas do modelo de partículas [HOU 92].	71
Figura 3.1 Modelo geométrico de uma malha deformável.	75
Figura 3.2 Modelo elástico da superfície.	77
Figura 3.3 Mola angular entre os pontos $P_0, P$ e $P_4$ , com ângulo inferior a $180^\circ$ .	80
Figura 3.4 Colisão de uma partícula com um triângulo.	85
Figura 4.1 Gráfico posição x tempo para simulação do movimento em um sistema massa-mola.	99
Figura 4.2 Resultado da animação de uma malha com parâmetros idênticos e $h$ diferente. (a) quadro gerado considerando $h = 0.1$ ; (b) $h = 0.05$ ; (c) $h = 0.01$ ; (d) $h = 0.005$ ; (e) $h = 0.001$ .	100
Figura 5.1 Etapas do protótipo FLEX3D.	103
Figura 5.2 Lista de triângulos.	109
Figura 5.3 Dados referentes às superfícies.	110
Figura 6.1 Animação de uma superfície a partir de uma forma inicial, até a sua situação de repouso. (a) Superfície inicial; (b) superfície em repouso (quadro 25).	117
Figura 6.2 Roteiro de animação usado na geração da animação apresentada na figura 6.1.	118
Figura 6.3 Animação de uma superfície que se projeta de um plano. São apresentados os quadros da animação de número 3, 5, 7, 9, 11, 13, 15, 17 e 19 da esquerda para a direita e de cima para baixo.	119
Figura 6.4 Parâmetros da superfície apresentada na figura 6.3.	120
Figura 6.5 Modelo de representação de um objeto 3D.	121
Figura A2.1 Animação de uma superfície ao vento.	126
Figura A2.2 Roteiro para parametrização da superfície exposta ao vento.	127

Figura A2.3 Simulação do movimento das cortinas de um teatro. ....	132
Figura A2.4 Parâmetros de configuração das cortinas.....	134
Figura A2.5 Animação de uma superfície com baixa capacidade de curvatura e torção e que se projeta de um plano. ....	135
Figura A2.6 Parâmetros de configuração da superfície apresentada na figura A2.5.....	136
Figura A2.7 Simulação da queda de uma toalha sobre uma mesa.....	138
Figura A2.8 Parâmetros de configuração da animação apresentada na figura A2.8.....	139

## LISTA DE TABELAS

Tabela 1.1	Classes de objetos e seus parâmetros na modelagem de pontes.....	25
Tabela 4.1	Comportamento do algoritmo relativo a um sistema massa-mola em diversas condições diferentes.....	98

## RESUMO

O crescente número de sistemas de animação que utilizam a cinemática para gerar movimento de objetos vem levando os pesquisadores a buscar outras alternativas para produzir resultados mais realistas. Com base nesta premissa, vários autores começaram a estudar a geração de movimento de objetos sintéticos através da aplicação da dinâmica. Assim surgiram os modelos baseados em leis físicas. Num primeiro momento foram abordados apenas objetos rígidos, passando-se mais tarde a considerar objetos articulados e, por fim, aqueles com características elásticas, também denominados de **objetos flexíveis** ou **deformáveis**.

O objetivo principal do trabalho é a definição de um modelo para simulação de objetos deformáveis no espaço euclidiano. São abordados tanto o modelo geométrico utilizado como o modelo físico, sendo ressaltadas as forças aplicadas sobre o objeto e as restrições que podem ser impostas pelo mundo virtual no qual o mesmo está inserido. Dentre as forças descritas, pode-se destacar: força gravitacional, elasticidade, força de curvatura e torção, colisão e atrito. A fundamentação do trabalho desenvolvido é apresentada na forma de uma introdução aos sistemas de animação, enfatizando os sistemas baseados em leis físicas e de uma revisão bibliográfica dos métodos de deformação existentes.

No que diz respeito à colisão de objetos elásticos, são descritos tanto os métodos estudados para a solução das mesmas, como as técnicas para detecção do choque.

A simulação do movimento é descrita sob dois aspectos: o algoritmo utilizado para a geração do movimento e a integração numérica das equações diferenciais no tempo. É abordado ainda, em detalhe, o protótipo desenvolvido com o propósito de validar o modelo proposto, sendo descrita a

linguagem criada a fim de permitir a especificação da animação e parâmetros diversos do modelo.

Por fim, são apresentados e avaliados os resultados obtidos através do desenvolvimento do modelo proposto por intermédio do protótipo FLEX3D. É dedicada ainda especial atenção às perspectivas futuras deste trabalho.

**PALAVRAS-CHAVE:** Animação, Animação Baseada em Física, Objetos Deformáveis.

TITLE: "DEFORMABLE OBJECT SIMULATION BASED ON DYNAMIC ANALYSIS"

## ABSTRACT

The growing number of animation systems that use kinematics to generate the motion of objects have led to other alternatives to produce more realistic results. Some authors began to study the animation of synthetic objects through the application of the dynamic concepts, creating the modern physically based models. At first, only rigid objects were treated; later on articulated objects were considered. At last, those with elastic characteristics (called flexible or deformable objects) were taken into consideration.

The main goal of this work is to define a simulation model for deformable objects in the euclidean space. Both the geometric and the physical models are presented, considering the forces applied to the object and the constraints defined by the virtual world. Described forces include gravity, elasticity, dumping force, collision and attrition. This work presents an introduction to animation systems focusing the physically based systems. After this, a bibliographic review of the existent deformation methods is made. Methods for detecting and solving the collision between two elastic objects are described.

Two aspects of the motion simulation are described: the algorithm used to generate the motion and the numeric integration of the differential equations in time. A prototype named FLEX3D is presented to validate the proposed model. The language used for specifying the animation is described and results obtained through the use of FLEX3D are also presented. Special attention is given to the possible future works.

**KEYWORDS:** Animation, Physically Based Animation, Deformable Objects.

# 1 INTRODUÇÃO

Ainda na década de 60, o computador começou a ser utilizado como ferramenta de apoio na confecção de filmes criados através do processo de animação tradicional [HUN 77]. Este fato, veio a desencadear o que hoje constitui uma das maiores linhas de pesquisa dentro da computação gráfica, a animação por computador. Inicialmente, surgiu a animação assistida por computador, onde a máquina era responsável apenas por automatizar algumas etapas envolvidas na produção da animação. Entretanto, com o desenvolvimento das pesquisas nesta área, surgiu a animação modelada, onde os atores possuem um modelo geométrico bem definido. Não se pode deixar de citar a animação procedural, onde vários elementos do modelo interagem a fim de determinar propriedades e comportamentos capazes de originar movimento [FOL 90].

Atualmente, a animação por computador vem sendo utilizada para as mais diversas aplicações, como na indústria de entretenimento, na educação, na aeronáutica com os simuladores de voo e na publicidade. Em simulações em geral, é preciso destacar a indústria automobilística, com a simulação de colisões de automóveis, a robótica, com a dinâmica do movimento dos robôs e a física com as pesquisas espaciais [FOL 90] [PUE 88]. É portanto fácil de perceber, em função das aplicações, que existem dois caminhos a serem seguidos nas pesquisas em animação: o caminho artístico, onde os esforços são dirigidos a fim de se obter resultados interessantes, no que diz respeito ao efeito visual, e a visualização científica onde as animações são geradas através da simulação de fenômenos científicos [FOL 90].

No que diz respeito ao primeiro caminho, "produzir animação é uma forma de arte e não apenas, um exercício de técnicas. A capacidade de contar uma história e expressar emoção através da animação é uma habilidade que leva anos para desenvolver" [KOC 87]. As pesquisas no sentido de gerar resultados artisticamente satisfatórios têm caminhado em



três direções. A primeira, concentra-se na automatização por computador, de técnicas de síntese de imagens e animação já existentes e que, no entanto, não podem ser largamente utilizadas, em função de diversos fatores, como custo e dificuldade de utilização. A técnica que melhor pode servir de exemplo neste caso talvez seja a Tela de Alfinetes (*pinscreen*) hoje praticamente desativada devido principalmente ao custo do equipamento [LOP 92] [NED 91]. Em segundo lugar, destacam-se as pesquisas nos modelos baseados em leis físicas. Isto se dá com o objetivo de gerar movimentos melhor sincronizados e mais harmônicos pois tal qual objetos em um mundo real, modelos dinâmicos tem uma atividade e uma resposta natural às forças sobre eles aplicadas. Com a incorporação da física na animação por computador, torna-se possível a "coreografia" de movimentos, através do controle de parâmetros físicos, condições iniciais e forças aplicadas [TER 88]. Por fim, têm avançado ainda as pesquisas no que se refere à definição de linguagens e/ou sistemas de animação voltados a artistas e não a programadores.

No que diz respeito à visualização científica, conforme [HEG 89] existe uma grande relação entre o controle de movimento na animação, na simulação e na visualização. Um problema atual de visualização científica é que os modelos originais usados para simulação não contêm necessariamente uma descrição geométrica adequada. Já no que tange à simulação, pode-se dizer que é intrinsecamente baseada em modelos dinâmicos, incluindo informações a respeito das propriedades físicas, como massa, forças e torques. Mais uma vez, portanto, tornam-se essenciais as pesquisas em modelos baseados em leis físicas, desta vez porém com o objetivo de auxiliar na obtenção de um modelo geométrico que possa ser visualizado no espaço temporal.

Nas seções seguintes são apresentados métodos de controle de movimento em animação tradicional, bem como uma introdução às técnicas de modelagem que envolvem o comportamento dos modelos. Este estudo foi feito com o objetivo de oferecer os subsídios necessário para o

desenvolvimento de um modelo próprio de simulação do comportamento de objetos deformáveis.

## 1.1 Métodos de Controle do Movimento

Diversos autores tem classificado a animação por computador de acordo com as suas aplicações, os métodos de controle do movimento , o modelo de animação utilizado, entre outros. Após um estudo das classificações sugeridas em [FOL 90] [KRÖ 89] [KOC 87] [ISA 87] [PUE 88], optou-se por apresentar aqui a classificação da animação quanto aos métodos de controle do movimento. Isto foi feito em função dos objetivos deste trabalho, pois a preocupação aqui é unicamente com a geração do movimento. Dentre os trabalhos estudados, optou-se por apresentar os métodos citados em [FOL 90] por ser esta a classificação que engloba o maior número de métodos de controle do movimento. São eles:

- Controle explícito;
- Controle procedural;
- Sistemas baseados em restrições;
- Modelos gravados;
- Uso de atores;
- Cinemática e dinâmica;
- Animação baseada em leis físicas.

### 1.1.1 Controle explícito

Controle explícito é o tipo mais simples de controle da animação. Neste método, também referenciado como sistema de quadros-chave (*key-frame*), o animador fornece ao sistema uma descrição de tudo o que ocorre

durante a animação. Isto pode ser feito através da especificação de translações, rotações e escalas dos objetos, ou através da indicação de quadros-chave e métodos de interpolação a serem usados entre estes quadros [FOL 90].

Em um sistema com controle explícito do movimento, o animador não precisa de conhecimento prévio de programação, uma vez que tudo que ele necessita saber é definir os parâmetros dos objetos com os dispositivos à sua disposição: *mouse*, *joystick*, *data glove* ou outro meio qualquer [FOL 90]. Existe ainda a possibilidade de interagir com o sistema, criando, excluindo ou alterando os quadros-chave.

Como desvantagem, há o fato de que é muito difícil modificar uma sequência de ações definida entre dois quadros-chave. Por isso existe a necessidade de especificar quadros mais próximos uns dos outros para situações onde se deseja um movimento mais complexo, a fim de preservar a coerência da animação. Por exemplo, considerando uma animação onde há o choque entre duas esferas em movimento, torna-se necessária a definição de diversos quadros-chave, a fim de que o detalhe da colisão seja bem representado.

Um exemplo bidimensional de controle explícito de movimento é a interpolação de imagens. Aqui, os quadros-chave são imagens e o movimento pode ser representado através da interpolação da cor (representada através das componentes R, G e B) de cada *pixel* entre os quadros [KRÖ 89].

### 1.1.2 Controle procedural

O controle procedural é intrínseco a modelos procedurais, pois os modelos, em si, contêm todas as informações necessárias para permitir o

controle do movimento. Nestes modelos, vários elementos se comunicam a fim de determinar suas propriedades, ou seja, cada elemento influencia o comportamento de seus elementos vizinhos.

Um exemplo de controle procedural de movimento são os sistemas de partículas, onde o comportamento de uma partícula (definido algoritmicamente) é afetado por aspectos de outros objetos da cena. Outro exemplo é a utilização de algoritmos na geração de formas naturais, como conchas, chifres e galhos de plantas, que podem ser representados por expressões matemáticas. No caso das conchas, estas são formadas através da torção e expansão de um modelo poliédrico básico. Para visualizar o processo de formação destes objetos, basta apresentar passo a passo o seu procedimento de geração [KOC 87]. Infelizmente, a utilização de controle procedural, pressupõe que o animador seja também um programador.

O controle procedural está diretamente envolvido em outros mecanismos de controle de movimento [FOL 90]. Em particular nos sistemas baseados em leis físicas, a posição de um objeto pode influenciar o movimento de outro. Por exemplo, um cubo colocado sobre uma superfície, pode alterar a trajetória de uma esfera após a colisão entre ambos.

### 1.1.3 Sistemas baseados em restrições

Restrições são condições atribuídas aos movimentos dos objetos, de forma que devam ser respeitadas durante a animação para que o efeito desejado pelo animador, seja mantido [SCH 92]. Por exemplo, dada uma esfera no universo e somente a força da gravidade atuando sobre esta esfera, pode-se dizer que a esfera cairia eternamente, a menos que colidisse com outro objeto, como o solo. Este movimento poderia ser facilmente controlado com o uso de restrições. Neste caso, a restrição seria a colisão, enquanto que a mudança na direção do movimento da esfera seria o resultado da satisfação dessa restrição.

Diversos exemplos de sistemas baseados em restrições podem ser citados, sendo sem dúvida o primeiro deles, o SKETCHPAD de Ivan Sutherland [SUT 63][FOL 90]. Neste sistema foram previstos três tipos de restrições: restrições de ponto, onde um objeto era livre para mover-se contanto que um de seus pontos permanecesse fixo; restrições de união, onde se garantia que, independente de seus movimentos, dois objetos deveriam estar sempre unidos; restrições de ângulo, onde se deveria garantir que, por exemplo, uma linha estivesse ligada a outra formando um ângulo de 90 graus.

Atualmente, o uso de restrições tem sido associado a sistemas de animação, mais especificamente a sistemas baseados em leis físicas (ver seção 1.5). Neste caso, as restrições são usadas para resolver problemas de colisão entre objetos, características estruturais dos objetos (como plasticidade dos materiais) e graus de liberdade em sistemas hierárquicos e de corpos articulados. No entanto, apesar do controle da animação ser facilmente definido através do uso de restrições, ele nem sempre é de fácil execução. Restrições isoladas são fáceis de controlar; porém, quando existem múltiplas restrições, seu controle pode ser uma tarefa difícil senão impossível. Pode ocorrer que ao satisfazer uma restrição, automaticamente outra é violada [KRÖ 89].

#### 1.1.4 Modelos gravados

Esta técnica consiste basicamente do uso de informações previamente gravadas do mundo real através de diferentes técnicas e dispositivos ou computadas por outros modelos [PUE 88].

Rotoscopia é a técnica de gravação de movimentos reais de uma pessoa ou um animal onde, posteriormente, é feita a correspondência de pontos-chave para um modelo animado. Uma das maneiras utilizadas para isso é a identificação, no modelo real, de pontos-chave (em geral identificados

através de pontos luminosos) em movimento. A seguir, é feito um processamento sobre as imagens geradas, a fim de obter o movimento destes pontos para mapeá-los posteriormente para o modelo virtual.

Como mecanismo de interação, pode-se citar a *data glove* onde é permitido ao usuário, o posicionamento e a movimentação de atores através do movimento de sua própria mão e dedos [FOL 90].

### 1.1.5 Uso de atores

O uso de atores é uma forma de controle procedural de alto nível. Um ator em uma animação pode ser visto como um pequeno programa invocado apenas uma vez a cada quadro, com o objetivo de determinar as características de um objeto na animação. Traçando um paralelo com a programação orientada a objetos, pode-se dizer que um ator corresponde a um objeto. Um ator, que corresponde portanto a um objeto na animação, pode enviar mensagens a outros atores, a fim de influenciar no seu comportamento [FOL 90].

### 1.1.6 Cinemática e dinâmica

O controle de movimento através do uso de modelos cinemáticos, produz animação a partir da definição de posições, velocidades e acelerações, enquanto que o controle através de modelos dinâmicos produz movimento através da parametrização de forças (que geram movimentos translacionais) e torques (que geram movimentos rotacionais), de onde derivam dados cinemáticos. Para gerar movimento através da dinâmica, é preciso que o objeto possua a definição mecânica de seus elementos. Por exemplo: materiais (massas) e juntas (membros e molas) [PUE 88].

Grande parte dos sistemas de animação por computador disponíveis atualmente, fazem uso de modelos cinemáticos por serem de utilização mais fácil e intuitiva, além de consumirem menos tempo computacional do que modelos dinâmicos. Por outro lado, o uso da dinâmica permite a geração de movimentos mais complexos, suaves e realísticos que muitas vezes são impossíveis de serem representados através da cinemática.

O uso de modelos dinâmicos, apesar de gerar movimentos bastante próximos dos reais, possui a desvantagem de não permitir que sejam parametrizadas as características do objeto no final da sequência de animação, não podendo desta forma existir um controle rígido dos atores por parte do animador. Para resolver este tipo de problema são usados os conceitos de dinâmica inversa e cinemática inversa. Assim, tanto as velocidades (no caso da cinemática) quanto as forças iniciais (no caso da dinâmica) podem ser definidas a partir da posição final do objeto. Considerando um modelo de objetos articulados, para definir o movimento de um ator levar a mão até a cabeça é mais fácil a partir da posição final (mão na cabeça) definir quais as forças aplicadas inicialmente para obter este movimento. Porém, dependendo da posição inicial da mão, pode-se ter um número muito grande de diferentes trajetórias para chegar ao mesmo resultado final. Estes problemas envolvendo corpos articulados têm sido objeto de estudo de diversos autores [CAL 91] [GIR 91] [WIL 87].

Outros estudos nesta área dedicam-se ao uso de restrições em modelos dinâmicos e controle e resolução de equações do movimento através do uso de métodos numéricos para a solução das equações diferenciais ordinárias [GRE 91].

### 1.1.7 Animação baseada em leis físicas

Os modelos baseados em leis físicas são obtidos através da simulação da evolução de sistemas físicos. O uso de princípios fundamentais

da física, permite que a animação apresente resultados coerentes com o comportamento dos objetos reais e que seja utilizada para simulação de fenômenos naturais. Isto é obtido através:

- (a) da modelagem de objetos levando em consideração algumas de suas características físicas (massa, centro de massa, momento de inércia, etc);
- (b) da geração de seus movimentos através da aplicação de forças e torques sobre os objetos;
- (c) do controle através do uso de restrições e condições iniciais;
- (d) da utilização de procedimentos numéricos que calculem valores para as variáveis que estão sendo simuladas [RIG 92].

A animação baseada em leis físicas utiliza-se da interação da cinemática, dinâmica, restrições, controle procedural e do uso de atores para obter movimentos realísticos de corpos articulados, objetos flexíveis e deformáveis ou simular fenômenos naturais.

## 1.2 Modelagem Algorítmica

A modelagem algorítmica, também denominada modelagem procedural, é utilizada para descrever objetos que podem interagir com eventos externos para modificarem-se a si próprios. Destaca-se como característica dos modelos algorítmicos, o fato de não existir um modelo final, estático. Portanto, pode-se classificá-los como sistemas dinâmicos. Nesta classe de modelagem, pode-se incluir os fractais, sistemas de partículas, modelos baseados em gramática, objetos deformáveis, etc [FOL 90] [CAR 92].



Os modelos algorítmicos vêm sendo utilizados já há bastante tempo. Em [MAG 85] é descrito um modelo algorítmico para a construção de pontes, onde são considerados mais de 100 parâmetros. Neste modelo, considera-se que uma ponte é formada por classes parametrizáveis de objetos e pela posição e orientação da mesma no universo. Na tabela 1.1 é apresentada a relação entre alguns parâmetros e as 4 classes existentes de objetos, sendo que são elas:

- **Via:** como uma ponte é utilizada para transpor um obstáculo, a via é o elemento principal do modelo, sendo que os demais são definidos com o objetivo de suportá-la;
- **Superestrutura:** consiste do conjunto de elementos utilizados para suportar a ponte, não devendo ser confundidos com os pilares de sustentação;
- **Pilares:** elementos que sustentam a ponte, fazendo a ligação com o solo;
- **Parapeito:** trata-se da estrutura de proteção lateral da ponte, usada especialmente para pedestres.

Tabela 1.1 - Classes de objetos e seus parâmetros na modelagem de pontes

Objeto	Parâmetros
Via	Largura, altura, comprimento, número de curvas
Pilar	Tipo, distância entre as colunas, altura, número de pilares
Superestrutura	Tipo, número de seções da ponte, tipo de desenho das seções
Parapeito	Distância dos suportes, altura em relação a ponte

Outro tipo de modelo algorítmico desenvolvido para uso em animação, é apresentado na forma de sub-atores e descrito em [MAG 85a].

Quando um sub-ator é definido, são definidas também diversas instâncias deste sub-ator, todas com os mesmos parâmetros. A diferença, portanto, está nas leis da animação aplicadas sobre cada uma destas instâncias, que podem ser completamente diversas umas das outras, gerando movimentos diferentes. Os sub-atores devem estar associados a atores, que podem ainda ser animados de forma global.

Uma técnica algorítmica que vem se tornando comum na modelagem computacional, é o uso de processos estocásticos na geração de objetos como: montanhas e nuvens [COL 89], árvores [OPP 86], plantas [SMI 84], ondas [GOS 90] e fogo [REE 83]. A modelagem estocástica faz uso da aplicação de perturbações randômicas ou pseudo-randômicas para criar imagens mais realísticas e/ou interessantes [KOC 87].

Oppenheimer, em [OPP 86], sugere que a geometria e a topologia do seu modelo são controladas por parâmetros numéricos. Parâmetros estes que, numa analogia com o corpo humano, corresponderiam ao DNA. Através da manipulação destes parâmetros genéticos, pode-se modificar a geometria do objeto em tempo real. Os efeitos randômicos, utilizados no mundo virtual, contribuem na produção de uma diversidade bastante grande de objetos e na geração de efeitos com um alto grau de realismo. Cabe salientar ainda que é possível a geração de "espécies" mais complexas e evoluídas, bastando para tal aumentar a quantidade de parâmetros significativos.

Um número bastante grande de técnicas de modelagem algorítmica está em desenvolvimento para reproduzir fenômenos naturais. Estas técnicas são baseadas no uso de sistemas formais (como gramáticas) ou algoritmos para gerar primitivas geométricas que descrevem a forma de um objeto [GRE 88]. Porém, pesquisas vêm sendo realizadas, em busca de uma técnica que permita especificar de forma fácil e genérica, tanto a forma como o movimento dos objetos enquadrados como modelos algorítmicos.

## 1.3 Modelos Baseados em Leis Físicas

Segundo [BAR 88], um modelo baseado em leis físicas consiste na representação matemática de um objeto e seu comportamento, considerando forças, torques e energias, além de outros atributos da física newtoniana. Com esta aproximação, torna-se possível simular movimentos realísticos tanto de corpos flexíveis como rígidos, sem no entanto especificar detalhes desnecessários.

Sob o aspecto matemático e algorítmico pode-se dizer que os modelos baseados em física exigem uma base maior do que os modelos convencionais que utilizam a cinemática. É necessária a utilização de métodos numéricos ou elementos finitos para discretizar os elementos do modelo, no tempo.

Os estudos deste tipo de modelagem vêm fornecendo uma aproximação inicial para a resolução de um problema genérico. Trata-se da simulação de diversos tipos de objetos, seus movimentos e materiais, bem como o controle da reação entre estes objetos e destes com o ambiente, tanto no espaço como no tempo. É preciso ressaltar ainda que os modelos baseados em física não pretendem reproduzir a realidade e sim produzir movimentos com base nos mesmos princípios utilizados na natureza.

### 1.3.1 Tipos de objetos e seus movimentos

Em geral, movimentos são descritos em termos do número de coordenadas independentes necessárias para especificar a posição de todos os componentes do sistema. A isso denomina-se graus de liberdade [WIL 87a]. Serão descritos a seguir, alguns tipos de objetos modelados para uso em animação e suas características no que diz respeito à movimentação.

### a) Partículas

Uma partícula pode ser descrita como um ponto no espaço euclidiano. Portanto, tanto para posicionar como para movimentar uma partícula, são necessárias 3 variáveis ( $x$ ,  $y$ ,  $z$ ). Considerando ainda que uma partícula, por consistir em um ponto, não possui movimento rotacional, pode-se dizer que a sua movimentação possui 3 graus de liberdade, conforme figura 1.1a.

### b) Corpos rígidos

Um corpo rígido é descrito por um conjunto de pontos, de forma que a distância entre eles não varie nunca. Estes pontos são utilizados para definir a forma do corpo rígido, ou como vértices de polígonos ou como pontos de controle de superfícies paramétricas. A movimentação de um corpo rígido, dá-se única e exclusivamente em relação a um sistema de coordenadas, normalmente o do universo. Portanto, o movimento de um corpo rígido é especificado por 6 graus de liberdade (figura 1.1b):  $x$ ,  $y$ ,  $z$  para translação (como nas partículas) e  $x$ ,  $y$ ,  $z$  para rotação.

### c) Corpos flexíveis

Teoricamente pode-se definir um corpo flexível como um conjunto infinito de pontos que se movimentam uns em relação aos outros [WIL 87a]. No entanto, na prática, considera-se um corpo flexível como um conjunto finito de pontos que são usados para definir vértices de polígonos ou como pontos de controle de superfícies paramétricas. Em contrapartida aos corpos rígidos, neste caso, tem-se como característica fundamental a distância variável entre os pontos que definem um corpo, no tempo.

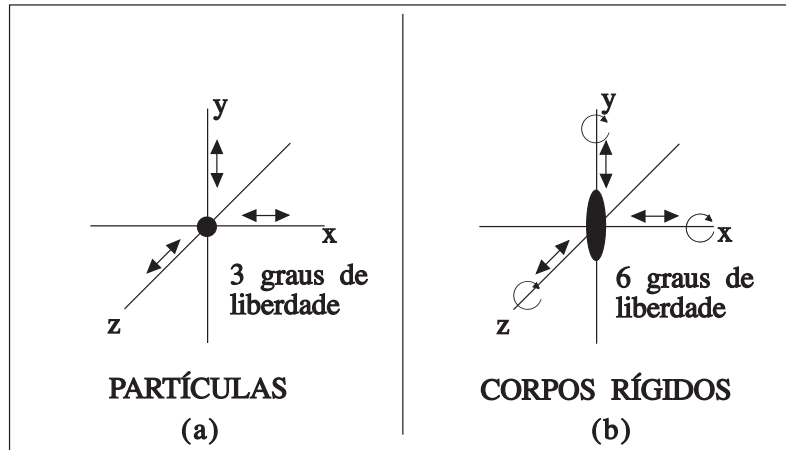


Figura 1.1 - Graus de liberdade e de movimento para (a) partículas e (b) corpos rígidos [WIL 87a].

Por movimentarem-se independentemente, pode-se considerar que cada ponto que compõe um corpo flexível comporta-se como uma partícula, possuindo portanto 3 graus de liberdade. Para definir a movimentação de um corpo flexível composto por  $n$  pontos, deve-se considerar  $n \times 3$  graus de liberdade, conforme ilustração na figura 1.2.

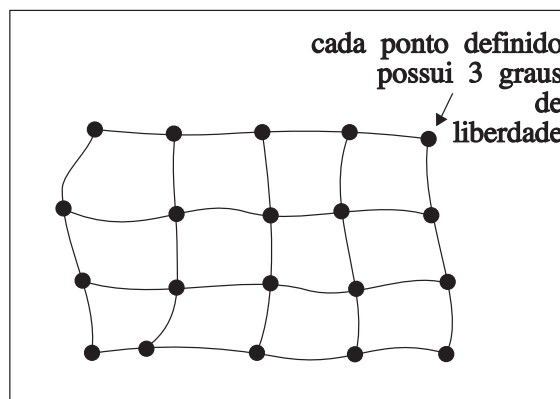


Figura 1.2 - Graus de liberdade de movimento para um corpo flexível [WIL 87a].

## d) Corpos articulados

Um corpo articulado é caracterizado por conter membros ligados por juntas, estabelecendo relações comportamentais. Estas relações consistem nos graus de liberdade de movimento [MUS 92]. Quanto aos membros, podem ser tanto corpos rígidos como corpos flexíveis, sendo que, na maioria dos sistemas de animação disponíveis, são rígidos [AMO 92][GAS 92] [ISA 88].

A movimentação de um membro de um corpo articulado é restringida pelos outros membros em que está conectado através das juntas, as quais podem possuir de 1 a 3 graus de liberdade. Portanto, o número total de graus de liberdade de um corpo articulado pode ser calculado como a soma entre os 6 graus de liberdade que o corpo possui em relação ao universo e o somatório dos graus de liberdade de todas as juntas do corpo. No caso do corpo humano, pode-se ter até 200 graus de liberdade. No entanto, se os membros forem considerados flexíveis, deve-se somar ao total de graus de liberdade do objeto, os graus correspondentes a cada membro isoladamente. Na figura 1.3, pode-se ver um exemplo de um corpo rígido articulado e os graus de liberdade de suas articulações.

### 1.3.2 Análise dinâmica

A análise dinâmica, refere-se ao estudo da relação entre forças, torques e a aceleração das massas [WIL 87]. Esta relação é expressa através das equações da dinâmica. Existe um número bastante grande de equações para representar a relação citada, porém todas elas derivam da mais simples delas, a Segunda Lei de Newton [RES 84]:

$$F = m.a$$

onde  $F$  é um vetor de força que atua sobre um ponto de massa  $m$  e  $a$  é a aceleração desta massa. Para um ponto de massa constante, esta equação pode ser resolvida de duas formas:

- Dada a força, encontra-se a aceleração;
- Dada a aceleração, encontra-se a força.

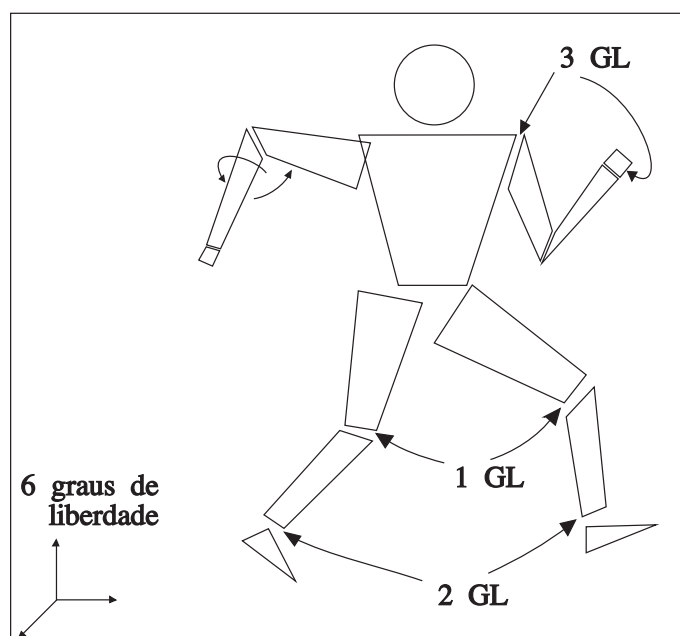


Figura 1.3 - Graus de liberdade de movimento para um corpo articulado.

Em aplicações envolvendo a robótica, o mais comum é fornecer a aceleração para encontrar a força necessária para permitir o movimento definido. Já em computação gráfica, o mais interessante tem sido fornecer as forças e torques externos e internos para obter a aceleração. Desta forma, de posse da aceleração (calculada), da posição e velocidade iniciais do objeto, pode-se traçar a trajetória percorrida, ou seja, a posição do objeto em cada instante de tempo. Para sistemas simples, com poucos graus de liberdade, pode ser utilizada integração analítica e o formalismo proposto por Newton.

No entanto, em sistemas mais complexos, torna-se necessário tanto a utilização de outros formalismos para representar o movimento, como o uso de técnicas de integração numérica [GRE 91].

Todas as formulações dinâmicas, expressam o comportamento do mundo modelado. No entanto, a escolha pela melhor opção depende do problema a ser tratado. A formulação de Newton é comum, mas outras formulações são muitas vezes mais apropriadas. Por exemplo, para corpos flexíveis é comum a aplicação de Lagrange [PLA 88] [WEL 69], enquanto que para corpos articulados utiliza-se a formulação matricial de Gibbs-Appel [WIL 91] ou ainda a recursividade desenvolvida por Armstrong [ARM 86].

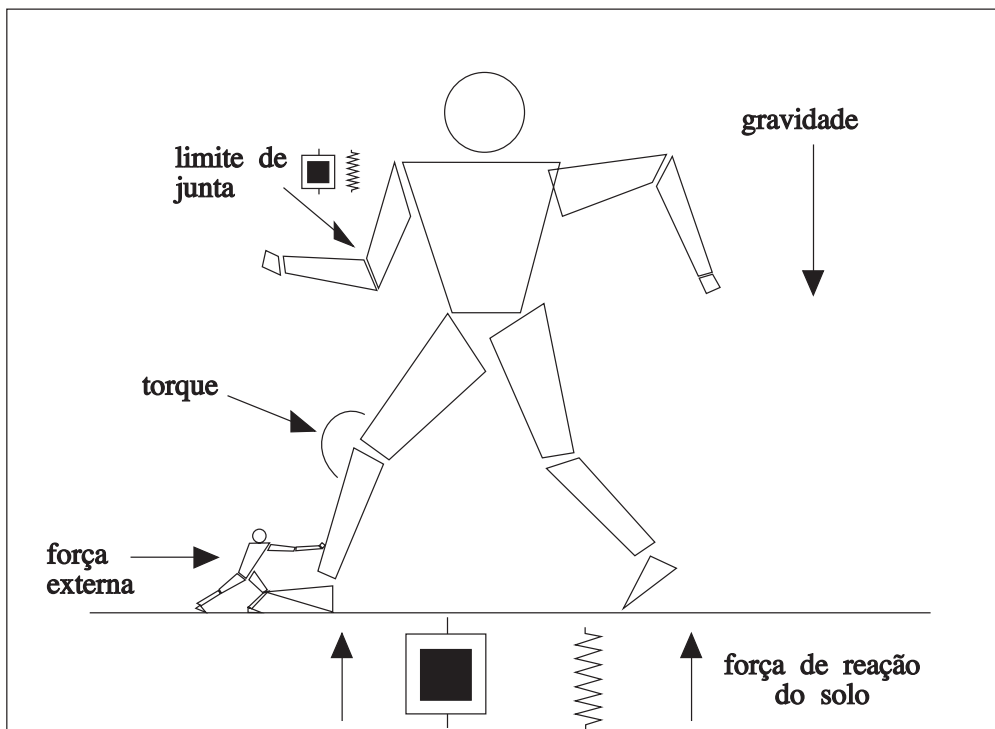


Figura 1.4 - Forças e torques que contribuem na geração do movimento.



Quanto às forças e aos torques que atuam sobre um corpo, pode-se dizer que algumas delas são automaticamente calculadas, outras podem ser simuladas através da utilização de molas e amortecedores, sendo que devem ser consideradas ainda, aquelas diretamente fornecidas pelo usuário [KOC 87]. Por exemplo, a força gravitacional pode ser automaticamente calculada; a interação entre o objeto e o solo, colisões em geral e os graus de liberdade das juntas (no caso de um corpo articulado), podem ser modelados como molas e amortecedores [ARN 89]; algumas forças internas, como as responsáveis pela contração e distensão dos músculos, são normalmente indicadas pelo usuário (ver figura 1.4).

### 1.3.3 Problemas envolvidos no uso da análise dinâmica

Os três problemas envolvidos na utilização da análise dinâmica são: custo computacional, instabilidade numérica e controle do movimento.

#### a) Custo computacional

O uso da análise dinâmica, principalmente a utilização de métodos lentos como o de Gibbs-Appell, dispense um tempo de processamento bastante grande. No intervalo de uma hora, consegue-se gerar apenas alguns segundos de animação usando um VAX-750, conforme [WIL 87]. Sendo que este tempo é dependente não só do método utilizado para cálculo do movimento, mas também da complexidade da cena.

Armstrong, Green e Lake propõe em [ARM 86] a utilização de processamento paralelo a fim de obter animações em uma velocidade bastante próxima do tempo real. Ainda, no que diz respeito ao tempo dispendido na geração das cenas de uma animação, é preciso salientar que o alto custo computacional não decorre apenas da dinâmica básica. Este custo é

decorrente também do grau de controle necessário, particularmente a detecção e a resposta às colisões.

## b) Instabilidade numérica

O segundo problema a ser considerado é a instabilidade numérica. As formulações dinâmicas utilizadas atualmente, envolvem um número considerável de amostragens a fim de garantir soluções coerentes para as equações. Isto é, as equações da dinâmica são solucionadas entre 50 e 300 vezes em um segundo, enquanto que a gravação em vídeo ocorre apenas 30 vezes por segundo [WIL 87].

O problema torna-se ainda mais sério quando o corpo possui muitos graus de liberdade e o movimento é muito controlado. Se a frequência em que são feitas as amostragens for insuficiente, a solução para as equações pode não ser encontrada, resultando em uma simulação errônea de movimento. Por exemplo, se o cálculo do movimento é efetuado a cada 0,1 segundo e ocorrer uma colisão entre 2 corpos no instante 4,05 segundos, pode acontecer desta colisão não ser detectada, não sendo por consequência respondida.

Com o objetivo de solucionar este tipo de problema, estudos vem sendo realizados no sentido de efetuar amostragens adaptativas [ARM 85]. Ainda utilizando o exemplo acima, para solucionar o problema colocado, o programa deveria poder identificar um intervalo onde haveria uma possível colisão, aumentando neste intervalo a frequência das amostragens.

### c) Controle do movimento

O último e mais grave problema envolvido na utilização da dinâmica é o controle do movimento. No que diz respeito às forças externas, pode-se dizer que as dificuldades são pequenas uma vez que são calculadas automaticamente, como por exemplo no caso da força gravitacional. O problema é mais flagrante na manipulação das forças internas, onde é necessário definir forças e torques como os realizados pelos músculos. É praticamente impossível quantificar a força muscular necessária para efetuar determinado movimento.

Analisando a questão de um prisma diferente, verifica-se ainda um outro tipo de problema: a dificuldade em controlar o movimento em um nível mais alto. Tomando como exemplo o caso dos corpos articulados, é bastante complexo o movimento de rotação do tronco para simular uma pessoa que se vira rapidamente para olhar para trás. Em primeiro lugar, porque o movimento do tronco depende da movimentação dos braços e do pescoço e em segundo porque as pessoas movimentam-se de forma diferente entre si. Outro exemplo é a simulação de uma pessoa caminhando. É um trabalho bastante exaustivo e experimental definir as forças e torques necessários para realizar uma animação com um grau razoável de realismo.

## 1.4 Modelagem Teleológica

Barr, descreve em [BAR 88a] uma nova concepção de modelagem, onde são criadas abstrações e representações matemáticas de objetos fisicamente realísticos e dependentes do tempo. Intuitivamente, um modelo teleológico pode ser considerado como um modelo orientado a objetivos, isto é, possui uma representação matemática que permite calcular o comportamento do objeto a partir de como este objeto "supostamente" será. Cada elemento inserido neste contexto é denominado um elemento teleológico

(do greco *teleos*, que significa objetivo, fim), pois tanto seu movimento como sua forma são construídos no sentido de alcançar um determinado objetivo.

A idéia central da modelagem teleológica é que a forma e o movimento dos objetos devem ser representados como um conjunto de objetivos, envolvendo força, posição, velocidade e demais variáveis físicas responsáveis por restringir o comportamento de um objeto. Os objetivos devem ser ordenados no tempo, a fim de produzir o que se denomina de uma "linha-de-tempo" referente ao comportamento. Além disso, deve-se salientar que a modelagem teleológica envolve explicitamente o parâmetro tempo, de forma a não haver distinção entre modelagem e animação.

Na figura 1.5 são apresentadas abstrações de um objeto e as conversões entre as respectivas representações. O nível mais baixo de abstração é quando um objeto é "modelado" como uma imagem. Neste caso, o objeto é representado através de primitivas 2D, consistindo de *pixels* de imagens ou ainda de vetores. A segunda abstração associa um objeto à sua forma no espaço 3D, ou seja, um objeto é representado por polígonos, superfícies paramétricas, etc. Num nível maior de abstração, representa-se um objeto através de seu comportamento físico. Neste caso, um objeto é modelado através das leis de Newton e apresentado como um corpo rígido ou como um corpo flexível. Por fim, a última abstração refere-se à teleológica. Um modelo teleológico incorpora "objetivos de comportamento" dependentes do tempo a fim de representar um objeto. Neste caso, pode-se considerar um objeto como um conjunto de objetivos dispostos no espaço temporal.

Foram definidas ainda, em [BAR 88a], maneiras de efetuar conversões entre as 4 abstrações acima citadas. Para representar um objeto através de seu comportamento físico, a partir do seu modelo teleológico, sugere-se a utilização do que denomina-se de métodos teleológicos, como por exemplo a dinâmica inversa. Para obter a representação do objeto através da sua forma no espaço é usada a simulação física para produzir posições e orientações do objeto. Por fim, com o objetivo de obter uma imagem que

sintetize um objeto, são usadas técnicas de *rendering*, como o *ray-tracing* para produzir uma imagem a partir de uma forma geométrica 3D.

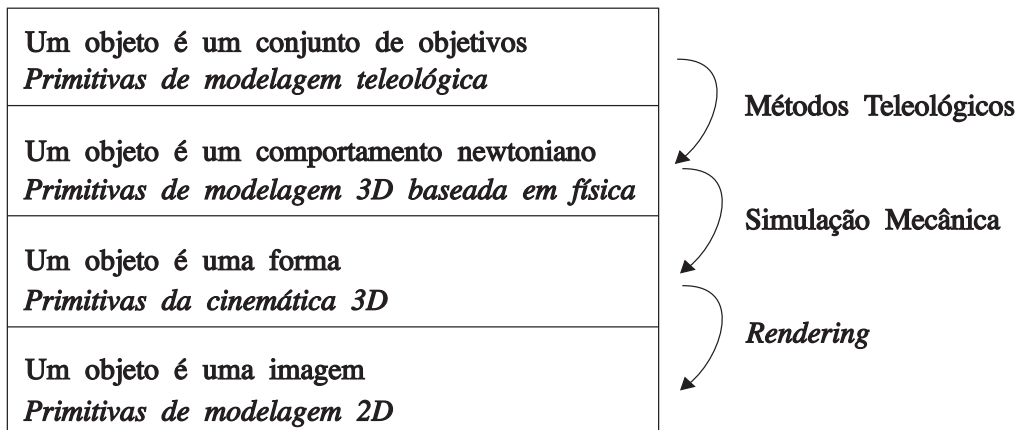


Figura 1.5 - Representações de objetos e conversões entre elas [BAR 88a].

## 1.5 Restrições em Sistemas de Simulação Dinâmica

Com o objetivo de facilitar a tarefa do animador de determinar o roteiro detalhado de uma animação, em um sistema de simulação dinâmica, foram desenvolvidos alguns métodos de controle do movimento. Estes métodos foram concebidos sob a premissa de não interferir na integridade física do sistema. São eles: restrições cinemáticas [ISA 87], restrições dinâmicas [BAZ 88] e restrições espaço-temporais [WIT 88].

### 1.5.1 Restrições cinemáticas

Bastante utilizadas em sistemas de simulação dinâmica, na geração do movimento, por serem simples e naturais de serem especificadas [ISA 87]. Trata-se de um conjunto de condições relacionadas com as

características cinemáticas do movimento, sendo que para cada condição estão associados métodos algorítmicos para realizar as operações necessárias para manter o sistema dentro das restrições previstas [SCH 92]. Como exemplo pode-se citar a restrição de parar um objeto quando este atingir a posição  $x,y,z$  ou ainda alterar a sua trajetória ao atingir a velocidade  $v$ .

### 1.5.2 Restrições dinâmicas

Trata-se das restrições que se baseiam nas leis físicas para manter o controle sobre o movimento. Em [BAZ 88] é apresentado um sistema no qual o comportamento de corpos rígidos é determinado através da simulação das leis de Newton. Com base nessas leis, são calculadas as forças de restrição dinâmica aplicadas sobre os corpos para que estes se comportem de acordo com as restrições geométricas especificadas pelo animador.

O sistema permite ao usuário a especificação dos movimentos através de 4 restrições geométricas:

- Ponto fixo no espaço: é fixado um ponto específico do objeto a uma determinada coordenada do espaço euclidiano, de forma que o objeto possa girar ou balançar;
- Ponto fixo em um ponto de outro corpo: dois corpos ficam unidos através dos pontos que estão em contato, podendo se mover livremente desde que os dois pontos em questão permaneçam juntos;
- Ponto fixo em um caminho: esta restrição permite determinar ao usuário a definição de um caminho no espaço 3D e um ponto no objeto, que deve seguir este caminho;
- Orientação de um ponto: permite que um corpo seja orientado no espaço, utilizando-se de transformações de rotação.

### 1.5.3 Restrições espaço-temporais

Conforme citado na seção 1.3.3, a maior dificuldade encontrada no uso de leis físicas para a geração de movimento é que o animador tem pouco controle sobre o movimento guiado pela Física. Com o objetivo de conciliar as vantagens da simulação dinâmica com a necessidade de um grau maior de controle, foi apresentada por Witkin em [WIT 88] uma nova abordagem para o uso de restrições, denominada **restrições espaço-temporais**. O princípio envolvido neste método é tratar as restrições cinemáticas como consequência de forças restritivas desconhecidas. Encontrar estas forças e adicioná-las à simulação de forma a controlar aquela componente de força que vai ao encontro das especificações das restrições. A abordagem espaço-temporal permite a imposição de restrições durante o movimento, com o efeito de propagar estas restrições para frente e para trás no tempo [SCH 92].

Como exemplo de restrições espaço-temporais, pode-se citar as posições iniciais, finais e intermediárias dos objetos, bem como as velocidades, influenciando diretamente na geração do movimento. Pode-se citar ainda outros tipos de restrições, como a limitação de forças, a prevenção de interpenetrações, entre outros.

As restrições impostas sobre os objetos, no sistema apresentado em [WIT 88], especificam as seguintes condições:

- O que um objeto deve fazer, no que diz respeito ao seu movimento;
- Como um movimento deve ser realizado;
- Qual a estrutura física do objeto, sua composição, peso, número de juntas, etc;

- Quais os recursos físicos disponíveis para realizar o movimento, por exemplo, os músculos do corpo ou os motores de um robô.

## 1.6 Objetivos e Organização do Trabalho

Animação é uma técnica que pode ser rapidamente definida como responsável por "fazer coisas modificarem-se ao longo do tempo" [CHA 89]. No entanto, em se tratando da animação de personagens, esta definição torna-se inadequada. O problema em questão não diz respeito ao modo como o animador descreve uma movimentação, mas sim em como ela é interpretada pelo personagem. Pode-se dizer que todos os elementos envolvidos na animação de um personagem devem influenciar para um melhor sincronismo de movimentos. Isto não quer dizer que devam ser usados procedimentos realísticos, mas que estes procedimentos devem ser cheios de expressividade, a fim de externar a personalidade do ator.

A animação de objetos gráficos, através do uso das técnicas de animação cinemática, freqüentemente requer a combinação de movimento entre múltiplas primitivas geométricas, cada uma envolvendo um número muito grande de variáveis, tais como posição, orientação, escala, etc. Para sintetizar movimentos de forma convincente, o animador deve especificar estas variáveis em cada instante de tempo da sequência de animação. Entretanto, muitas vezes os resultados obtidos não são satisfatórios; movimentos, especialmente quando se trata de objetos complexos, tendem a parecer pouco naturais. Em suma, criar animações que pareçam naturais, através da cinemática, exige muita paciência e conhecimento [TER 88a].

Como alternativa, tem-se utilizado atualmente a análise dinâmica, técnica que oferece resultados realistas, uma vez que se baseia nos princípios fundamentais da física. Mesmo usuários inexperientes conseguem criar movimentos realísticos através da aplicação de forças a modelos



dinâmicos, na simulação física de mundos. Procedimentos numéricos geram automaticamente valores que variam no tempo de acordo com as leis de Newton. Como objetos em um mundo real, os modelos dinâmicos tem uma atividade e uma resposta natural a forças e torques sobre ele aplicados. Este tipo de modelo leva os animadores a pensar mais como um tipo de "coreógrafos", não precisando se preocupar com os parâmetros e detalhes exigidos pela animação cinemática, devendo se concentrar em qualidades mais abstratas, como tempo, ritmo e estilo. Com a incorporação da física na animação por computador, é permitido que se "coreografe" movimentos, através do controle de parâmetros físicos, condições iniciais e forças aplicadas.

Na utilização dos modelos baseados em leis físicas para gerar animação, num primeiro momento, foram abordados apenas os objetos rígidos, passando-se mais tarde a considerar também os objetos articulados e por fim aqueles ditos deformáveis, ou ainda, flexíveis. A animação de personagens "deformáveis" exige que se desenvolva um modelo geométrico que permita a mudança de forma ao longo do tempo. Em computação gráfica, isto pode ser definido através de modelos físicos de curvas, superfícies ou mesmo sólidos deformáveis, governados através de leis mecânicas. Essas leis podem ser expressas, por exemplo, através de equações diferenciais dinâmicas, unificando deste modo a descrição de formas e movimentos. Com a resolução dessas equações numericamente, deve-se poder criar animações realísticas envolvendo a interação entre modelos deformáveis e a aplicação de várias forças, meio ambiente e obstáculos impenetráveis, em uma simulação física da realidade [TER 88].

Por outro lado, a resolução de um problema como, por exemplo, o de uma peça de roupa que voa ao vento e cai sobre um manequim dá-se puramente através da física, não sendo necessária nenhuma interferência da computação gráfica para resolver tal situação [FOL 90]. Daí, conclui-se que a real motivação para as pesquisas em modelos baseados em física não se

concentra na simulação de situações e sim na capacidade e vontade de visualizar resultados graficamente mais realísticos.

Este trabalho tem como objetivo um estudo dos modelos baseados em leis físicas, mais especificamente dos objetos com características elásticas (deformáveis).

No capítulo 2 é apresentada uma revisão bibliográfica dos métodos de deformação existentes, sendo que são abordadas não apenas as deformações por leis físicas, mas também as deformações ditas geométricas. São apresentados tanto métodos estáticos como dinâmicos de deformação, bem como são abordados trabalhos utilizando métodos baseados em restrições.

O capítulo 3 apresenta o modelo de representação de objetos flexíveis desenvolvido, salientando o modelo geométrico e o modelo físico utilizado para tal. São apresentadas em detalhe as forças aplicadas sobre o objeto e as restrições que podem ser impostas pelo mundo virtual no qual o mesmo está inserido. Dentre as forças descritas, pode-se destacar: força gravitacional, elasticidade, força de curvatura e torção, colisão e atrito.

O capítulo 4 aborda as características do modelo, no que diz respeito à simulação do movimento, apresentando o algoritmo utilizado para a geração do movimento e a integração numérica das equações diferenciais, no tempo.

O capítulo 5 descreve em detalhe o protótipo desenvolvido com o propósito de validar o modelo proposto. É apresentada a estratégia de detecção do choque utilizada, bem como a linguagem definida a fim de permitir a especificação da animação.

No capítulo 6 são apresentadas as conclusões e avaliados os resultados obtidos a partir do modelo. É dedicada especial atenção às perspectivas futuras deste trabalho.

## 2 MODELOS DEFORMÁVEIS

Atualmente, muitos estudos vêm sendo desenvolvidos no sentido de modelar objetos não-rígidos, ou seja, objetos que possuam características elásticas. Dentre estes estudos, tem-se pesquisado tanto modelos geométricos cujo movimento é controlado através da cinemática, como modelos baseados em outros atributos físicos como massas, forças e torques e com movimento controlado pela dinâmica. É preciso salientar ainda o uso de restrições na modelagem dos objetos deformáveis.

Com o objetivo de definir melhor o que se espera de um modelo flexível, torna-se necessário avaliar quais seriam suas propriedades desejáveis. Platt e Barr, em [PLA 88a], indicam 5 propriedades que devem ser incorporadas a um modelo, a fim de gerar movimentos suaves e convincentes. São elas:

- **Realismo físico:** O comportamento de um objeto flexível deve ser natural e intuitivo. Para tanto, sugere-se a utilização da teoria da elasticidade para controlar o movimento destes objetos;
- **Controlabilidade:** Um objeto flexível deve ser capaz de movimentar-se de acordo com um roteiro de animação. Deve ainda poder seguir um caminho pré-definido, ao mesmo tempo que se contorce e interage com outros objetos;
- **Não-interpenetração:** Um objeto flexível deve interagir consigo mesmo e com outros objetos, sem que haja interpenetrações. No entanto, este trabalho deve ser executado sem dispendir um tempo computacional muito elevado;
- **Limite de compressibilidade:** O modelo flexível deve permitir aos objetos que estes mantenham seu volume constante. A não ser em casos em que se deseje simular o

esmagamento de objetos sem que se esparramem para os lados, como por exemplo uma esponja;

- **Moldabilidade:** Um modelo flexível deve poder ser moldado através da aplicação de forças externas. Para isso, deve ser utilizada a teoria da plasticidade, que descreve materiais que não retornam a sua forma de repouso após uma deformação muito grande.

Este capítulo tem como objetivo apresentar uma revisão bibliográfica dos modelos deformáveis. São apresentadas algumas técnicas desenvolvidas com o objetivo de produzir tanto deformações estáticas como o movimento de objetos com características elásticas. Estas técnicas estão aqui classificadas em dois grandes grupos: os modelos geométricos e os modelos baseados em leis físicas.

## 2.1 Deformação Geométrica

Um crescente interesse vem sendo registrado no que diz respeito à modelagem e animação geométrica de objetos flexíveis e deformáveis [BEC 92]. Barr introduziu a modelagem geométrica de deformações em objetos sólidos, através da manipulação de operadores sobre dados abstratos [BAR 84]. Sederberg e Parry, em [SED 86], introduziram o conceito da deformação em formas-livres (*Free-Form Deformation - FFD*), o qual permite esculpir modelos sólidos [CHA 89]. O FFD é hoje um dos métodos mais usados na obtenção de deformações geométricas.

A técnica conhecida como deformação animada em formas-livres (*Animated Free-Form Deformation - AFFD*), desenvolvida por Coquillart e Jancène, foi concebida a partir do FFD [COQ 91]. É apresentada uma ferramenta de deformação, composta por duas grades 3D: a inicial e a final, interativamente deformada pelo usuário. Quando o objeto é inserido na grade final, ele se deforma de acordo com a mesma. A animação de um objeto 3D se

dá através da associação do mesmo com a ferramenta acima descrita [BEC 92].

Outras técnicas foram desenvolvidas com o objetivo de gerar deformação. A seguir serão descritas em detalhe algumas dessas técnicas, bem como aplicações de modelos deformáveis.

### 2.1.1 Deformação de primitivas sólidas

Em [BAR 84] são introduzidas algumas novas operações hierárquicas para uso na modelagem de sólidos. Estas operações têm a capacidade de simular torções, curvaturas, afunilamentos e outras transformações sobre objetos geométricos. O principal resultado dessas transformações, é que o vetor normal de uma superfície deformada pode ser calculado diretamente através do vetor normal da superfície antes da deformação e de uma matriz de transformação.

Através deste método, deformações são facilmente combinadas em uma estrutura hierárquica, criando objetos complexos a partir de outros simples. Os vetores de posição e os vetores normais ao objeto simples são usados para calcular os vetores correspondentes ao objeto na sua forma mais complexa. Cada nível na hierarquia da deformação requer uma matriz adicional para calcular o novo vetor normal.

Uma **deformação global** sobre um sólido 3D pode ser especificada como uma função matemática  $\underline{F}^1$ , que modifica explicitamente as coordenadas globais dos pontos que definem este sólido. Os pontos que definem o sólido antes da deformação são denominados  $\underline{x}$ , enquanto que os pontos do sólido já deformado são dados por  $\underline{X}$ . As componentes  $x, y, z$  do vetor  $\underline{x}$ , são designadas  $x_1, x_2, x_3$ , quando for conveniente. Da mesma forma ocorre

---

<sup>1</sup> Uma variável sublinhada indica que a mesma corresponde a um vetor de componentes  $x,y,z$ .

com as componentes do vetor  $\underline{X}$ , porém referenciadas através de letras maiúsculas. Matematicamente, uma transformação global é representada pela equação:

$$\underline{X} = \underline{F}(\underline{x})$$

Uma **deformação local** é especificada através da modificação do espaço tangente ao sólido. Os vetores normal e tangente à superfície não deformada podem ser transformados nos vetores normal e tangente à superfície deformada. Para tanto, é necessária a manipulação de regras de transformação envolvendo a multiplicação da função de transformação  $\underline{F}$  pela matriz jacobiano  $\underline{J}$ <sup>2</sup>.

A matriz jacobiano  $\underline{J}$  para a função de transformação  $\underline{X} = \underline{F}(\underline{x})$  é função de  $\underline{x}$  e pode ser calculada através das derivadas parciais de  $\underline{F}$  nas coordenadas  $x_1, x_2, x_3$ :

$$\underline{J}_i(\underline{x}) = \frac{\partial \underline{F}(\underline{x})}{\partial x_i}$$

Resumindo, a  $i$ ésima coluna de  $\underline{J}$  é dada pela derivada parcial de  $\underline{F}(\underline{x})$  em função de  $x_i$ .

Quando a superfície de um objeto é definida através de uma função paramétrica de duas variáveis  $u$  e  $v$ ,

$$\underline{x} = \underline{x}(u, v),$$

os vetores tangentes à superfície podem ser obtidos através de combinações lineares de derivadas parciais de  $\underline{x}$  em função de  $u$  e  $v$ . A direção do vetor

---

<sup>2</sup> Uma variável duplamente sublinhada corresponde a uma matriz de duas dimensões.

normal pode ser obtida pelo produto vetorial entre dois vetores tangentes a duas superfícies lineares independentes.

A regra de transformação do vetor tangente, pode ser expressa na forma matricial, como:

$$\frac{\partial X}{\partial u} = J \frac{\partial x}{\partial u}$$

que é equivalente a:

$$X_{i,u} = \sum_{j=1}^3 J_{ij} x_{j,u}$$

A fim de melhor ilustrar o método proposto, será apresentado um exemplo de transformação: o afunilamento global ao longo do eixo Z. A transformação de afunilamento é semelhante à escala. Trata-se da variação de duas componentes globais enquanto mantém-se uma terceira fixa. Na figura 2.1, a função  $f(z)$  é uma função linear que decresce conforme  $z$  aumenta. Quando a função de afunilamento  $f(z) = 1$ , a porção do objeto deformado não se altera; o objeto aumenta de tamanho como função de  $z$ , quando  $f'(z) > 0$  e diminui quando  $f'(z) < 0$  [BAR 84].

$$r = f(z),$$

$$X = rx,$$

$$Y = ry,$$

$$Z = z$$

A matriz de transformação tangente é dada por:



$$\underline{\underline{J}} = \begin{bmatrix} r & 0 & f'(z)x \\ 0 & r & f'(z)y \\ 0 & 0 & 1 \end{bmatrix}$$

A taxa de expansão volumétrica local, do determinante, é  $r^2$ . A matriz de transformação normal é dada por:

$$r^2 \underline{\underline{J}}^{-1T} = \begin{bmatrix} r & 0 & 0 \\ 0 & r & 0 \\ -rf'(z)x & -rf'(z)y & r^2 \end{bmatrix}$$

A transformação inversa é dada por:

$$\begin{aligned} r(Z) &= f(Z), \\ x &= X / r, \\ y &= Y / r, \\ z &= Z \end{aligned}$$

### 2.1.2 Deformação em formas livres (FFD)

A técnica apresentada em [SED 86], propõe a deformação de sólidos geométricos de maneira livre. Esta técnica pode ser usada em conjunto com diversos sistemas de modelagem geométrica, como CSG ou B-rep por exemplo. Quanto às superfícies a serem deformadas, podem ser planares, quádricas, paramétricas ou, ainda, implicitamente definidas. No que diz respeito à deformação propriamente dita, pode ser tanto global como local, sendo que também é possível deformar um sólido sem que seu volume seja alterado.



Figura 2.1 - Transformação de afinamento em uma faixa.

O FFD é referenciado como um método para esculpir modelos sólidos. Seu princípio de funcionamento baseia-se em um mapeamento  $R^3$  para  $R^3$ , através da aplicação dos polinômios de Bernstein. Portanto, pode-se dizer que não se trata da deformação de um objeto e sim do universo onde está inserido.

Para melhor entendimento da formulação envolvida no FFD, é conveniente fazer uma analogia física com um modelo conhecido. Considerando um paralelepípedo de material plástico transparente e flexível no qual é "mergulhado" um ou vários objetos também com características de flexibilidade. No momento em que o paralelepípedo for deformado, a forma dos objetos "mergulhados" no mesmo será também alterada, de maneira consistente com o movimento do paralelepípedo. A figura 2.2 ilustra a analogia proposta, num universo 2D.

Matematicamente, o FFD pode ser definido primeiramente através de um sistema local de coordenadas na região periférica do paralelepípedo, conforme figura 2.3. Cada ponto  $X$  possui coordenadas  $(s, t, u)$  neste sistema, de forma que:

$$X = X_0 + sS + tT + uU$$

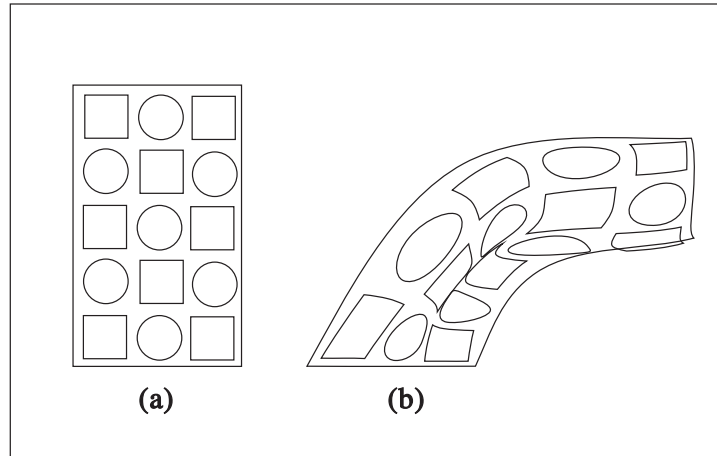


Figura 2.2 - (a) retângulo transparente contendo quadrados e círculos; (b) retângulo transparente após sofrer o processo de deformação.

As coordenadas  $(s,t,u)$  do ponto  $X$  podem ser encontradas através da aplicação da álgebra linear. Uma solução vetorial é

$$s = \frac{T \times U \cdot (X - X_0)}{T \times U \cdot S}, t = \frac{S \times U \cdot (X - X_0)}{S \times U \cdot T}, u = \frac{S \times T \cdot (X - X_0)}{S \times T \cdot U} \quad (2.1)$$

É preciso salientar ainda que para cada ponto interior ao paralelepípedo,  $0 < s < 1$ ,  $0 < t < 1$  e  $0 < u < 1$ .

Após a definição do sistema de coordenadas do paralelepípedo, é preciso definir uma grade através de pontos de controle  $P_{ijk}$ . Assim, tem-se  $l+1$  planos na direção  $S$ ,  $m+1$  na direção  $T$  e  $n+1$  na direção  $U$  [SED 86]. Estes pontos de controle têm sua localização definida do seguinte modo:

$$P_{ijk} = X_0 + \frac{i}{l}S + \frac{j}{m}T + \frac{k}{n}U$$

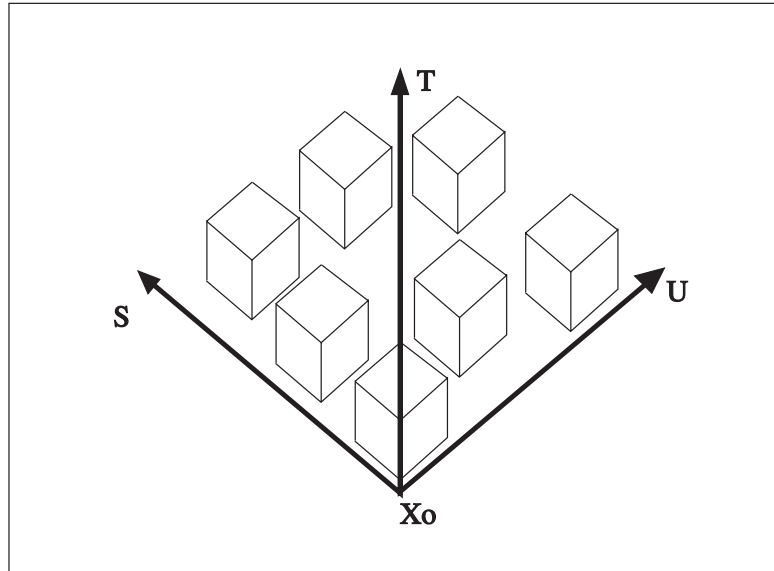


Figura 2.3 - Sistema de coordenadas  $(s, t, u)$  [SED 86].

A deformação é especificada através da movimentação dos pontos de controle  $P_{ijk}$  e de uma função de deformação definida pelo produto tensorial de polinômio de Bernstein. A posição de um ponto  $X$  após a deformação, é dada primeiramente através do cálculo das coordenadas  $(s, t, u)$ , na equação 2.1, sendo a seguir calculado  $X_{ffd}$  através do polinômio de Bernstein, como um vetor de coordenadas cartesianas:

$$X_{ffd} = \sum_{i=0}^t \binom{t}{i} (1-s)^{t-i} s^i \left[ \sum_{j=0}^m \binom{m}{j} (1-t)^{m-j} t^j \left[ \sum_{k=0}^n \binom{n}{k} (1-u)^{n-k} u^k P_{ijk} \right] \right]$$

O vetor  $X_{ffd}$  representa a posição do ponto  $X$  após a deformação. Neste momento, o ponto  $P_{ijk}$  corresponde aos coeficientes do polinômio de Bernstein. Na figura 2.4, verifica-se os pontos  $P_{ijk}$  antes e depois da deformação.

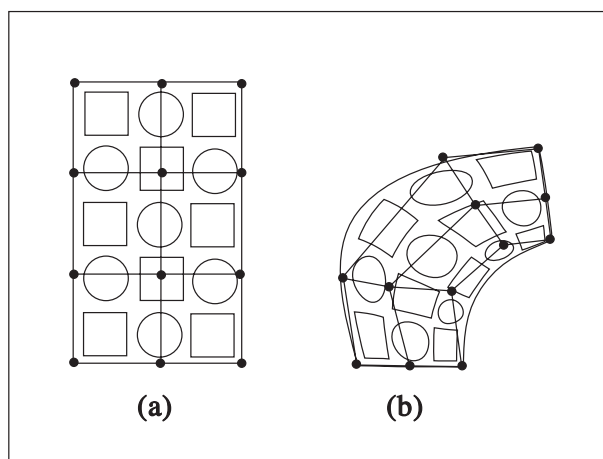


Figura 2.4 - (a) pontos de controle sobre o retângulo; (b) posição dos pontos de controle, após a deformação.

A técnica de deformação acima descrita tem dado origem a diversas pesquisas. Dentre estas é preciso destacar o trabalho realizado por Coquillart e Jancène e apresentado em [COQ 91], que propõe a deformação animada em formas livres, como um novo modelo. Não pode deixar de ser citado também o recente trabalho de Hsu, Hughes e Kaufman em [HSU 92], que aborda a manipulação direta de formas livremente deformadas.

### 2.1.3 Método baseado em restrições

Em [WIT 87], é apresentado um método baseado em restrições para animação de objetos geometricamente deformáveis. O modelo geométrico utilizado constitui-se por primitivas geométricas (cubos, cilindros, esferas, cones, etc) e por operadores (translações, rotações, escalas, torções e deformações), formando um modelo hierárquico. A construção de um modelo deste tipo pode ser dividida em duas partes: a montagem da estrutura de árvore que representa a hierarquia e a definição dos parâmetros internos tanto das primitivas como dos operadores.

O método proposto objetiva expressar e solucionar restrições em modelos hierárquicos parametrizáveis. As restrições são formuladas em forma de funções de energia no que se refere aos parâmetros do modelo, sendo que uma restrição é considerada satisfeita quando a função de energia (aqui denominada de energia de restrição) for igual a zero. Para obter a energia de restrição total do sistema, basta efetuar o somatório de todas as energias de restrição. Fundamentalmente, o método baseia-se na minimização desta energia, congregando as facilidades expostas abaixo.

- **Montagem Automática de Modelos:** a partir de uma configuração inicial, o objeto pode se mover e deformar-se parametricamente, a fim de satisfazer determinadas restrições;
- **Modelos Animados:** uma vez construído, um objeto pode se movimentar de acordo com restrições que variam no tempo, sem no entanto deixar de atender às restrições estáticas;
- **Generalidade e Modularidade:** o modelo permite que sejam formuladas novas restrições, para aplicação em um modelo arbitrário, sem que seja necessário um conhecimento prévio das restrições existentes bem como das primitivas geométricas e operadores utilizados. Da mesma forma, novas primitivas e operadores podem ser criados e inseridos no sistema;
- **Adição:** a energia de restrição total do sistema é composta através do processo de adição. Portanto, a solução de um sistema de restrições é dada através da solução de uma única equação, a soma dos termos-energia;
- **Controle Interativo:** uma vez que o processo de satisfação das restrições dá-se através de modificações no espaço paramétrico, objetivando a minimização da energia, isto pode ser feito no tempo. Como vantagem permite-se a intervenção do usuário, a fim de resolver problemas de ambiguidade.

Este método permite a definição de restrições bastante úteis, tais como: manter um determinado ponto do objeto fixo a um ponto no espaço 3D; manter 2 pontos de diferentes objetos em contato; manter um ponto específico de um objeto em contato com um ponto qualquer de um segundo objeto; manter um ponto de um objeto sobre uma linha no espaço; prevenir a colisão entre 2 objetos, evitando a interpenetração; permitir restrições diretas sobre parâmetros, com o objetivo de estabelecer valores padrão ou ainda definir relações entre parâmetros. Como exemplo, pode-se forçar uma rotação sobre o eixo  $x$  para ser sempre de  $30^\circ$ .

Concluindo, pode-se citar a generalidade como vantagem fundamental do método. Em contra partida, é preciso ressaltar que é necessário um processamento numérico intensivo, o que envolve um gasto bastante grande em termos de tempo computacional.

#### 2.1.4 Aplicações

A deformação de modelos geométricos, possui diversos campos de aplicação. Pesquisas têm sido realizadas no sentido de apresentar novos modelos, desenvolvidos com vistas a objetivos específicos.

Em [CHA 89], [MAG 88] e [MAG 91] são descritos métodos de deformação para simular a musculatura humana, sendo que em [MAG 88] e [MAG 91] é abordada exclusivamente a musculatura da mão. Quando se trata de simular músculos, o problema a ser analisado não diz respeito apenas à deformação. São estudados, em separado, o esqueleto e todos os problemas envolvidos na simulação de corpos articulados, assim como a superfície que o recobre, responsável por simular a musculatura. No entanto, trata-se de um trabalho bastante inter-relacionado, uma vez que a contração (gerando normalmente a dilatação) e distensão dos músculos é dependente do movimento do esqueleto.

Outro trabalho que deve ainda ser lembrado é o apresentado em [MIL 91]. É descrita uma metodologia para extrair a topologia de um modelo geométrico fechado, no que diz respeito aos dados referentes a volume. A técnica, denominada modelos geometricamente deformados (GDM), inicia com um modelo simples e topologicamente fechado. Em seguida, o modelo é deformado com base em um conjunto de restrições, como por exemplo o crescimento do objeto, resultando no aumento de volume.

## 2.2 Deformação Baseada em Leis Físicas

Uma nova área da computação gráfica tem se preocupado com os estudos envolvendo a simulação das propriedades físicas de objetos modelados. Tanto o movimento como a deformação destes objetos podem ser calculados através da aplicação de leis físicas [CHA 89].

O primeiro modelo de objetos deformáveis baseado em física foi desenvolvido objetivando uma simulação estática (sem animação) da realidade. Weil, em [WEI 86] propõe a modelagem de roupas, através da interpolação de superfícies entre curvas catenárias, a fim de simular roupas penduradas por pontos [TER 88].

Barr, Terzopoulos, Platt e Fleischer usaram componentes moleculares discretos, para modelar o comportamento elástico dos objetos [PLA 88]. Estes componentes podem ser vistos como pontos de massa interconectados através de molas com atributos de dureza e amortecimento baseados nas propriedades físicas dos objetos. Terzopoulos e Fleischer ainda estenderam seu modelo, para incluir componentes flexíveis e rígidos [TER 88a], bem como para permitir a simulação de comportamento inelástico [TER 88b].



Uma deformação é dita "elástica", quando após removidas todas as forças externas, o objeto retorna completamente à sua forma inicial. Para simular um objeto com características elásticas, pode-se utilizar uma aproximação energética. Como uma generalização de uma mola ideal, um modelo elástico possui energia potencial durante a deformação. As deformações "inelásticas" são genericamente como aquelas que não obedecem às leis da elasticidade clássica. Deformações inelásticas ocorrem em materiais reais, quando estes são submetidos a temperaturas ou forças que excedem certos valores limites alterando irreversivelmente a sua forma [TER 88].

Outros autores têm abordado a deformação de objetos através da física. Dentre os modelos desenvolvidos, pode-se salientar os métodos baseados em restrições, apresentados primeiramente por Platt e Barr, em [PLA 88a], e mais recentemente por Metaxas e Terzopoulos, em [MET 92]. A simulação de tecidos foi abordada ainda por Dias e Rebordão em [DIA 91] onde é apresentado um modelo energético para simulação estática e por Carignan, Yang, Magnenat-Thalmann e Thalmann em [CAI 92], com o desenvolvimento de um modelo dinâmico para simular tecidos complexos. Outros trabalhos são ainda apresentados em [BAF 92], [CEL 91], [PEN 89], [SCL 91], [WIT 90] e em [HOU 92] e [SZE 92], onde são abordados modelos baseados em sistemas de partículas.

A seguir, serão apresentados em detalhe alguns métodos de deformação julgados importantes no desenvolvimento das pesquisas em sistemas baseados em leis físicas.

### 2.2.1 Deformação estática de superfícies por curvas catenárias

Este método, apresentado por Weil em [WEI 86], aborda uma solução para um problema bastante específico. Uma peça de tecido retangular num espaço 3D é fixada a pontos no universo através de pontos de restrição.

O problema consiste em determinar uma possível solução para simular a forma do tecido quando "pendurado" por estes pontos, sendo que não é suficiente determinar uma superfície lisa, uma vez que um tecido pode apresentar dobras.

O tecido é representado como uma grade e modelado como uma matriz bidimensional na qual cada célula armazena as coordenadas de um ponto da superfície, no universo. Para aumentar a resolução da grade, basta aumentar a densidade da matriz. Quanto à deformação, esta é feita em 2 estágios: aproximação da superfície e relaxação.

Inicialmente, apenas os pontos de restrição (determinados pelo usuário) estão corretamente posicionados no espaço 3D. Em seguida, são traçadas curvas catenárias entre estes pontos, de forma que os pontos da grade entre os pontos de restrição são também posicionados no espaço. Analisando-se a grade que representa o tecido, pode-se verificar (conforme figura 2.5) que as curvas catenárias formam triângulos. É escolhido o maior lado de cada triângulo e este é subdividido formando outros dois. São traçadas mais curvas catenárias e assim sucessivamente até todos os pontos da malha estarem posicionados.

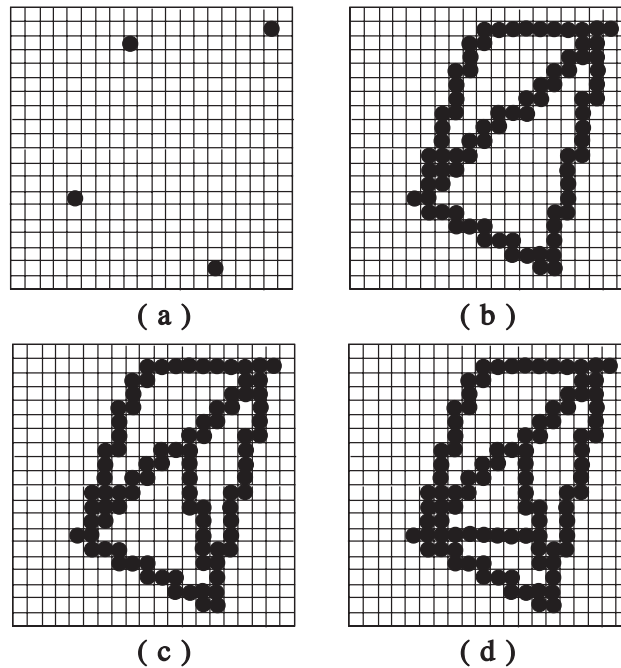


Figura 2.5 - (a) Pontos de restrição; (b) Triângulos formados a partir da conexão dos pontos; (c) e (d) Subdivisão dos triângulos [WEI 86].

A fase de relaxação trata-se de um processo iterativo, onde a cada iteração são reposicionados os pontos da grade. Para achar a distância de equilíbrio de um ponto em relação aos seus 4 vizinhos, basta somar os vetores do ponto até cada um dos seus vizinhos (ver figura 2.6).

Alguns materiais são mais flexíveis do que outros, o que faz com que o objeto se curve e torça com mais intensidade. Esta propriedade pode ser facilmente incorporada ao método apresentado. São considerados 3 pontos consecutivos na grade e é controlado o ângulo formado entre estes pontos (inicialmente igual a  $180^\circ$ ). Admitindo um ângulo mínimo a ser formado entre os pontos, controla-se o nível de flexibilidade do tecido.

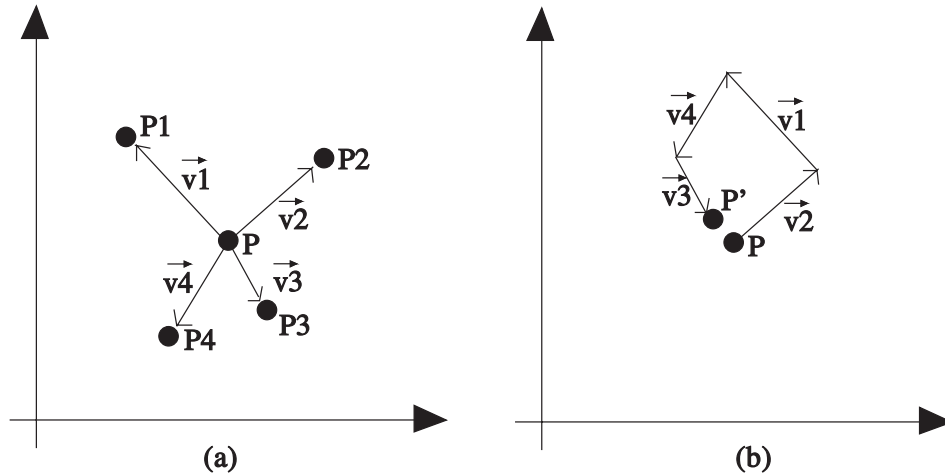


Figura 2.6 - (a) Ponto de restrição  $P$ , seus 4 vizinhos ( $P_1, P_2, P_3, P_4$ ) e os vetores  $\vec{v}_1, \vec{v}_2, \vec{v}_3, \vec{v}_4$ ; (b) Soma dos vetores ( $\vec{v}_1, \vec{v}_2, \vec{v}_3, \vec{v}_4$ ) a fim de encontrar a distância de repouso e conseqüentemente a nova posição de  $P$  ( $P'$ ).

O estágio de aproximação da superfície pode ser completamente eliminado do processo, uma vez que é utilizado apenas para obter uma aproximação inicial, reduzindo o número de iterações necessárias durante o estágio de relaxação. Um exemplo é o da animação, onde os pontos de restrição são gradualmente movimentados a cada quadro. Neste caso, uma aproximação da superfície pode ser considerada o quadro anterior ao que está sendo calculado.

### 2.2.2 Modelo energético para deformação estática de superfícies

O modelo apresentado por Dias e Rebordão em [DIA 91] baseia-se na caracterização global da energia potencial associada à geometria da superfície, mais especificamente nas suas componentes gravitacional, elástica e de curvatura. Admite-se um modelo simplificado no qual a superfície

deformável é apresentada por uma grade retangular de pontos onde cada um desses pontos possui massa e é representado pelas coordenadas  $(x,y,z)$ .

A elasticidade da superfície segue o modelo das molas elásticas de Hooke, da seguinte forma: dados os 8 vizinhos de um ponto  $P$  da grade, o modelo utiliza 2 molas longitudinais de constante elástica  $kl$ , 2 molas transversais de constante  $kt$  e 4 molas diagonais com constante  $kd$ , conforme figura 2.7.

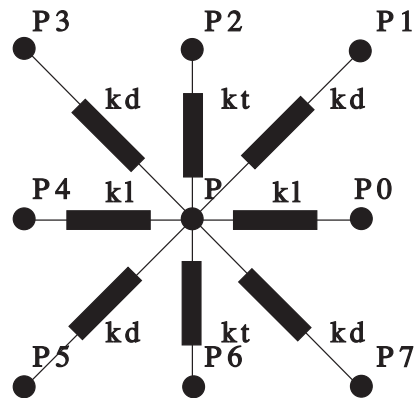


Figura 2.7 - Componente elástica do modelo.

Para caracterizar a componente de curvatura e torção da superfície, o modelo baseia-se no cálculo dos seguintes ângulos entre o ponto  $P$  e os seus 8 vizinhos:  $P_1PP_5, P_4PP_0, P_2PP_6, P_3PP_7$ . Considera-se a superfície deformada quando algum dos ângulos medidos for diferente de  $180^\circ$ .

Quanto às restrições impostas pelo universo onde está inserida a superfície, considerou-se apenas o campo gravitacional e a existência de pontos imóveis. Isto permite a simulação de um tecido "pendurado" em alguns pontos, sob a ação da gravidade.

Considerando a representação geométrica de uma superfície deformável através de uma matriz bidimensional  $\mathbf{M}$  e as características físicas já citadas, pode-se determinar o valor da energia potencial total associada à superfície. Este valor é obtido pela aplicação da função potencial a toda a matriz:

$$E_{POTENCIAL}(\mathbf{M}) = E_{GRAV}(\mathbf{M}) + E_{ELAST}(\mathbf{M}) + E_{CURV}(\mathbf{M}),$$

onde  $E_{CURV}$  determina o grau com que a superfície dobra e torce,  $E_{ELAST}$ , o quanto se distende e comprime, e  $E_{GRAV}$ , o quanto cai sob o efeito da gravidade. Variando o peso relativo destas funções no cálculo da energia total, obtêm-se superfícies deformáveis com características físicas diferentes, como borracha, papel, pano, etc.

No que diz respeito ao método de simulação estática utilizado, pode-se dizer que utiliza-se um processo iterativo para a obtenção de uma forma final que represente a superfície. Em cada iteração  $k$ , aplica-se a todos os pontos da grade um método de relaxação: cada ponto é deslocado no espaço de um valor  $\Delta$ , de tal modo que a energia potencial global da superfície diminua. Os pontos são varridos e as novas posições somente são atualizadas no fim de um varrimento completo. Assim, após cada varrimento, cada ponto  $P_{i,j}$  recebe um novo valor da energia. A determinação deste novo valor de  $E_K$ , tal que  $E_K < E_{K-1}$ , corresponde a um problema de minimização não-linear, da função de energia dependente das variáveis  $(x,y,z)$ . O processo iterativo termina quando  $\Delta E = |E_K - E_{K-1}|$  for desprezível ou quando a variação nas coordenadas dos pontos for insignificante.

Finalmente, o método prevê o tratamento da colisão com corpos rígidos. Para tal, utiliza-se um método de penalização no qual adiciona-se um termo à expressão da energia potencial da superfície, tornando-a proibitiva quando se viola uma determinada restrição, introduzindo assim uma penalização. Por exemplo, conhecendo-se para cada sólido uma função

interior  $I$ , tal que  $I(P_{i,j}) = 0$  para pontos  $P_{i,j}$  da superfície situados no bordo do sólido,  $I(P_{i,j}) > 0$  para pontos fora do sólido e  $I(P_{i,j}) < 0$  para pontos no seu interior, a função de penalização seria:

$$E_{PENAL} = \exp(-k \cdot I(P_{i,j})),$$

sendo  $k$  uma constante elevada. Desta forma, a energia potencial viria:

$$E_{POTENCIAL} = E_{GRAV} + E_{ELAST} + E_{CURV} + E_{PENAL}$$

### 2.2.3 Modelo dinâmico de deformação

Após uma análise dos métodos apresentados para simular dinamicamente curvas, superfícies e sólidos, optou-se por abordar aqui o modelo desenvolvido por Terzopoulos. Esta escolha foi feita por ser este o modelo mais genérico dentre os consultados, além de ser fortemente referenciado pelos demais autores.

Num primeiro momento, desenvolveram-se modelos deformáveis baseados na teoria da elasticidade. As equações de Lagrange para movimento de corpos não-rígidos são expressas como funções de posições no espaço euclidiano. Estas funções - paramétricas em coordenadas intrínsecas do objeto - posicionam explicitamente cada ponto do objeto no espaço, em função do tempo. As equações diferenciais parciais de movimento incluem uma força elástica não-linear associada com o corpo deformável. Esta força foi designada invariante no que diz respeito ao movimento do corpo como um todo.

O modelo a ser apresentado aqui, denominado modelo híbrido [TER 88a], leva em consideração dois tipos de movimento: o movimento das partículas que compõem o corpo deformável, em relação ao sistema de referência do objeto e o movimento do objeto em relação ao sistema de

coordenadas do universo, como se este fosse um corpo rígido. O movimento completo do objeto no espaço euclidiano dá-se através da soma dos dois movimentos acima citados.

Quanto à representação geométrica de um corpo deformável, tem-se  $u = (u_1, u_2)$  (no caso de superfícies deformáveis) como as coordenadas de um ponto no corpo  $\Omega$ . A posição de um ponto do corpo, em relação ao sistema de coordenadas do universo ( $\Phi$ ), é dada por<sup>3</sup>:

$$\mathbf{x}(u, t) = [x_1(u, t), x_2(u, t), x_3(u, t)]^T$$

Um corpo deformável é representado como a soma entre a componente de referência:

$$\mathbf{r}(u) = [r_1(u), r_2(u), r_3(u)]^T$$

e a componente de deslocamento:

$$\mathbf{e}(u, t) = [e_1(u, t), e_2(u, t), e_3(u, t)]^T$$

Ambas as componentes acima descritas são expressas em termos do sistema de referência do objeto ( $\phi$ ), cuja origem coincide com o centro de massa do objeto (ver figura 2.8):

$$\mathbf{c}(t) = \mu(u)\mathbf{x}(u, t)du,$$

onde  $\mu(u)$  é a densidade de massa do corpo deformável. Portanto, a posição dos pontos de massa que constituem um corpo, em relação ao sistema de referência  $\phi$ , é:

---

<sup>3</sup> Para melhor entendimento, os vetores no espaço serão representados em negrito.



$$\mathbf{q}(u,t) = \mathbf{r}(u) + \mathbf{e}(u,t)$$

A movimentação do corpo como um todo, dá-se através da movimentação do sistema de referência  $\phi$  em relação a  $\Phi$ . Assim, considera-se as velocidades linear e angular de  $\phi$  em relação a  $\Phi$ , como:

$$\mathbf{v}(t) = \frac{d\mathbf{c}}{dt}; \mathbf{w}(t) = \frac{d\theta}{dt},$$

onde  $d\theta$  é o ângulo infinitesimal de rotação de  $\phi$  em função de  $\Phi$ . Desta forma, a velocidade dos elementos de massa do objeto em relação a  $\Phi$ , dada as velocidades  $\mathbf{e}(u,t) = \partial \mathbf{e} / \partial t$  em relação a  $\phi$ , é<sup>4</sup>:

$$\mathbf{x}(u,t) = \mathbf{v}(t) + \mathbf{w}(t) \times \mathbf{q}(u,t) + \mathbf{e}(u,t)$$

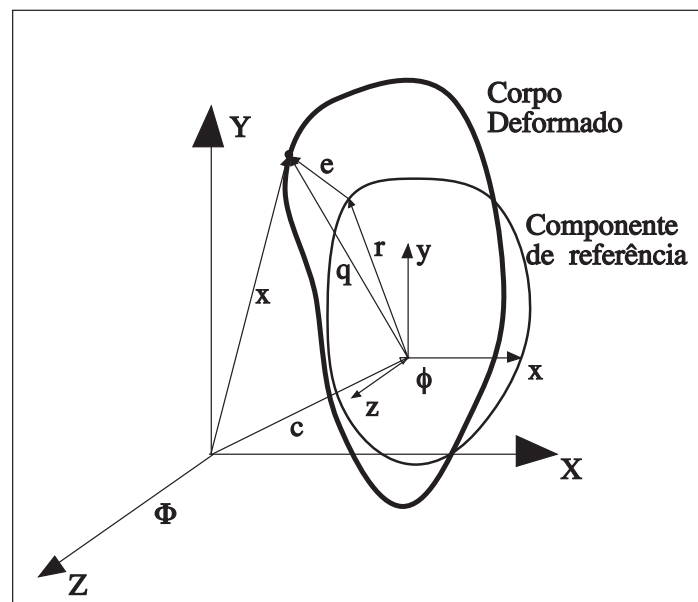


Figura 2.8 - Representação geométrica de um corpo deformável [TER 88a].

<sup>4</sup>  $\mathbf{x}$  denota a derivada em função do tempo  $\partial \mathbf{x} / \partial t$ .

Portanto, um modelo deformável é descrito completamente através das posições  $\mathbf{x}(u,t)$ , das velocidades  $\dot{\mathbf{x}}(u,t)$  e das acelerações  $\ddot{\mathbf{x}}(u,t)$ . Assim, a equação de Lagrange que define o movimento, pode ser expressa de forma relativamente simples

$$\mu\ddot{\mathbf{x}} + \gamma\dot{\mathbf{x}} + \delta\mathbf{x} = \mathbf{f},$$

onde o primeiro termo representa a força inercial dependente da massa, o segundo, a velocidade dependente da força de amortecimento e o terceiro, a força de elasticidade.  $\mathbf{f}(\mathbf{x},t)$  representa o vetor de forças externas.

As equações de movimento utilizadas para resolução do método são contínuas tanto no espaço como no tempo. Portanto, para simular numericamente um modelo deformável, é necessário discretizar as equações utilizando o método de aproximação dos elementos finitos ou das diferenças finitas. Em seguida, é preciso integrar as equações discretizadas através do tempo, simulando assim a dinâmica dos corpos deformáveis. Para isto, utilizou-se um procedimento de integração semi-implícito.

Posteriormente, o modelo apresentado foi ampliado a fim de permitir também um comportamento inelástico aos corpos deformáveis. Foram consideradas então: a plasticidade do material, a viscoelasticidade e a fratura, conforme [TER 88b].

#### 2.2.4 Método baseado em restrições

Este método, apresentado em [PLA 88a], descreve como utilizar restrições matemáticas baseadas na física e na teoria da otimização para criar animações controladas e realísticas de modelos flexíveis baseados na física. São abordados 2 tipos de restrições: restrições de reação (RCs) e restrições lagrangeanas (ALCs = Augmented Lagrangian Constraints). As

primeiras permitem o cálculo rápido de colisões entre corpos flexíveis e outros modelos poligonais. Por outro lado, as ALCs criam efeitos de animação como por exemplo manter o volume de um objeto estável após uma colisão e moldar substâncias com características semelhantes à argila.

A técnica desenvolvida por Platt e Barr utiliza o conceito de métodos de restrição baseados em força, o qual adiciona forças externas a sistemas físicos para produzir movimentos físicos realísticos. Existem pelo menos 3 métodos de restrição baseados na física (ver figura 2.9), para permitir a criação de modelos flexíveis com as propriedades de realismo físico, controlabilidade, não-interpenetração, limite de compressibilidade e moldabilidade. São eles:

- **Restrições dinâmicas:** são fáceis de usar em sistemas de dinâmica simples. No entanto, os modelos flexíveis caracterizam-se por possuírem muitas variáveis de estado, o que dificulta o uso da dinâmica inversa (utilizada nas restrições dinâmicas). Por este motivo, não é abordado aqui este tipo de restrição;
- **Restrições de reação:** utilizam um método de projeção modificado para permitir restrições simples como guiar um modelo deformável por um caminho ou ainda prevenir que um modelo flexível interpenetre um polígono qualquer. As RCs geram forças de reação que cancelam as outras forças que levariam à violação da restrição. As RCs são fáceis e simples de usar e não requerem a adição de equações diferenciais adicionais ao sistema físico. No entanto, pode ser aplicada apenas uma RC, por vez, a um ponto de massa;
- **Restrições de otimização:** usam a filosofia da teoria da otimização para restringir sistemas físicos.

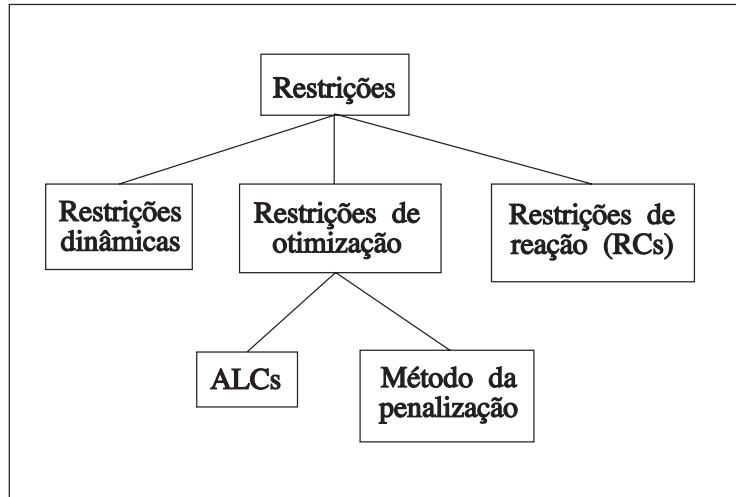


Figura 2.9 - Hierarquia dos métodos de restrição [PLA 88a].

Existem 2 tipos de restrições de otimização. O tipo mais simples, denominado **método de penalização**, adiciona uma energia extra ao sistema, a fim de penalizar comportamentos incorretos. Este método funciona de maneira análoga à colocação de elásticos para atrair o sistema físico para os pontos de restrição. O segundo tipo de restrição, denominado ALC, consiste de um método de restrição de otimização, que adiciona equações diferenciais para o cálculo dos multiplicadores de Lagrange aos sistemas físicos. Essas equações diferenciais levam o sistema a eventualmente satisfazer múltiplas restrições, inclusive na presença de forças externas. As ALCs consistem fundamentalmente de uma versão diferencial do método dos multiplicadores da teoria da otimização. São utilizados para restringir modelos flexíveis a fim de manter suas características de incompressibilidade e moldabilidade.

Algumas substâncias naturais, como por exemplo a argila, não retornam à sua forma natural após uma deformação grande. A utilização de modelos elásticos puros é inadequada para este tipo de material. Portanto, o uso de ALCs para simular substâncias que podem ser moldadas é uma aproximação razoável. Além do mais, aplicando forças externas a estes objetos, pode-se moldar formas interessantes sem especificar numericamente a sua forma de repouso.

## 2.2.5 Modelo baseado em sistemas de partículas

Uma outra abordagem utilizada na modelagem de objetos flexíveis, é a apresentada em [HOU 92], onde propõe-se a simulação de superfícies flexíveis através de sistemas de partículas. Neste caso, apresenta-se um método para simular o comportamento de um sistema de partículas através da dinâmica. A dinâmica de partículas utilizada inclui tanto a interação entre as partículas como a das partículas com o mundo. O algoritmo proposto utiliza uma aproximação dividida em 2 fases para simular uma superfície deformável. Esta aproximação permite a especificação das restrições entre as partículas usando funções de energia, além de prover a aplicação de métodos baseados em forças para gerar um movimento dinâmico convincente.

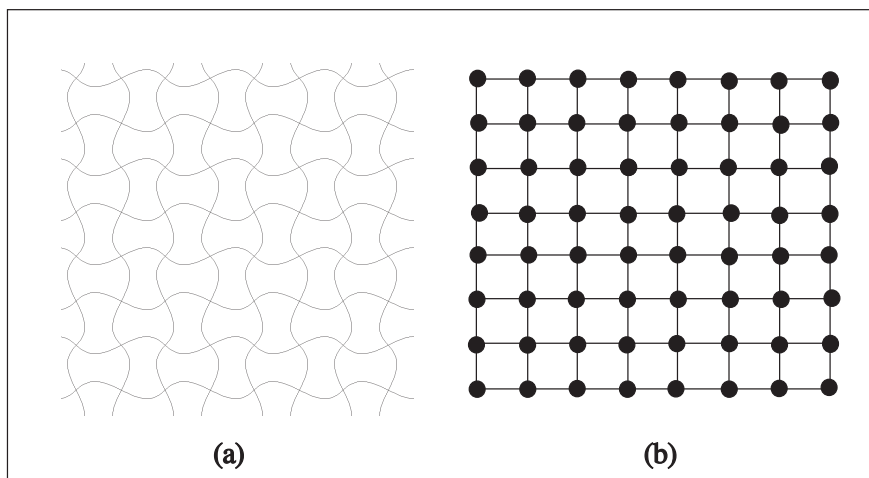


Figura 2.10 - (a) Malha que representa a superfície deformável;  
(b) Representação geométrica através de uma grade de partículas [HOU 92].

O método utilizado para modelar tecidos trata uma superfície como sendo um conjunto de pontos. Estes pontos correspondem às posições onde os fios que formam a malha do tecido se cruzam, conforme mostra a

figura 2.10. O modelo apresenta ainda várias restrições e interações a nível da malha do tecido, através do uso de funções energéticas de relações geométricas simples entre as partículas e seus 4 vizinhos próximos. Estas funções têm por objetivo englobar 4 interações mecânicas básicas. São elas: colisão das fibras do tecido, espichamento, curvatura e torção, além da torção da malha em torno de um dos pontos de cruzamento do tecido. As funções energéticas que representam estas interações, podem ser vistas graficamente na figura 2.11.

Analisando as funções energéticas descritas, pode-se concluir facilmente que tanto as funções de colisão como de espichamento descrevem uma relação de distância entre as partículas. Por outro lado, as funções de curvatura e torção da malha atuam como relações de variação angular entre as partículas.

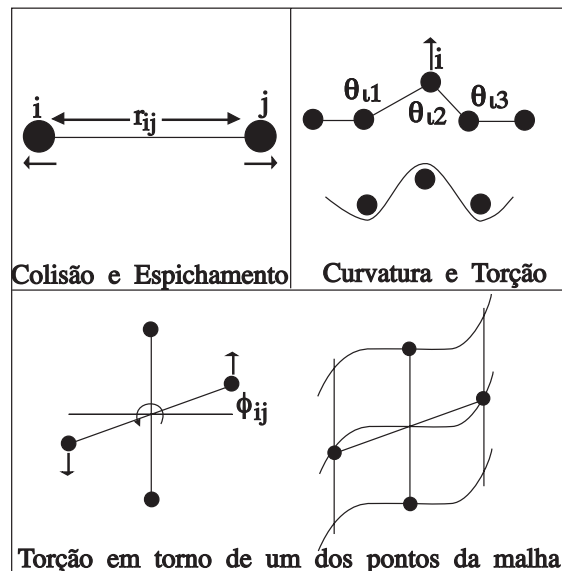


Figura 2.11 - Funções energéticas do modelo de partículas [HOU 92].

As funções apresentadas descrevem interações locais a nível microscópico. No entanto, a aplicação dessas funções a todas as partículas que compõem o tecido, produz um comportamento macroscópico da superfície como um todo. Para simular dobraduras no tecido, é necessário ainda aplicar a força gravitacional sobre as partículas do modelo e rastrear seu comportamento dinâmico, ao mesmo tempo em que se trata das colisões com outros objetos geométricos que compõem a cena.

## 2.2.6 Aplicações

Diversos estudos têm sido realizados, no que diz respeito às aplicações de modelos deformáveis. Talvez o mais comum dentre estes estudos seja a procura de modelos para simular tecidos. Conforme demonstrado nos ítems acima, diversas técnicas têm sido desenvolvidas com este propósito. Dentre elas, pode-se salientar o trabalho realizado por Dias e apresentado em [DIA 91]. Este trabalho busca inclusive a definição dos parâmetros físicos de materiais conhecidos, a fim de simular realisticamente o comportamento destes materiais. O autor busca a utilização de seu modelo na indústria de confecção, mais especificamente em CAD para tecidos. Para tanto, são pesquisadas ainda maneiras para obter tecidos não retangulares, a simulação de costuras entre dois panos, além da interação das superfícies deformáveis com objetos sólidos.

Outra aplicação interessante para superfícies deformáveis é a apresentada por Lundin em [LUN 87] que consiste na simulação das velas de uma escuna, através de um modelo dinâmico. As velas são representadas por malhas flexíveis, as quais são submetidas a um processo iterativo de relaxação. É calculada a força resultante em cada vértice da malha. A força do vento é representada como um vetor de força externa que atua na direção da superfície normal de cada vértice. Em cada iteração, cada vértice da malha é reposicionado com o objetivo de encontrar o ponto de equilíbrio em relação aos seus vizinhos, sendo que após sucessivas iterações, a vela toda é posta em

equilíbrio. Várias condições limites são aplicadas ao modelo como, por exemplo, fixar uma linha da malha a um mastro, uma verga ou ainda a um dos cabos de sustentação do mastro. A forma das velas pode ainda ser alterada dinamicamente, bastando para isto que haja uma alteração na força do vento.

Miller em [MLL 88] e [MLL 91] descreve um sistema massa-mola para simular o movimento de cobras e minhocas. Cada segmento da criatura é modelado como um cubo com massas nas extremidades e molas unindo essas massas. Para cada intervalo de tempo, o comprimento das molas e as suas velocidades são usadas para computar as forças exercidas pelas mesmas sobre os pontos de massa. Entretanto, para gerar o movimento das cobras como um todo, é necessário ainda que sejam exercidas forças sobre as massas. Isto é feito globalmente através da aplicação da força da gravidade e localmente, aplicando-se vetores de força sobre pontos de massa de forma individual.



### 3 MODELO DE REPRESENTAÇÃO DE OBJETOS FLEXÍVEIS

Como mencionado anteriormente, o objetivo fundamental deste trabalho consiste no desenvolvimento de um método para simular o comportamento de objetos flexíveis. Após um estudo detalhado dos métodos já existentes para simular este tipo de objeto, conforme apresentado no capítulo anterior, foram tomadas algumas decisões fundamentais, a fim de dar prosseguimento às pesquisas.

Em primeiro lugar, optou-se pela aplicação de um modelo baseado em leis físicas, decisão esta tomada juntamente com a de apresentar os resultados da deformação, no tempo. Esta abordagem foi escolhida devido ao fato de possibilitar a futura utilização deste trabalho em simulações, além, é claro, de ser a que apresenta melhores resultados no que diz respeito à qualidade tanto na forma dos objetos como no seu movimento suave.

Outra importante decisão tomada foi a de modelar apenas superfícies elasticamente deformáveis. O fato de não utilizar sólidos deformáveis, veio em decorrência da complexidade em fazê-lo e, conseqüentemente, do tempo disponível para tal. Esse mesmo motivo levou à modelagem apenas das características elásticas dos objetos.

O desenvolvimento de um modelo próprio de deformação deu-se a partir de estudos em física, mais especificamente em dinâmica. Os motivos que levaram a isto, foram:

- a falta de informação sobre o assunto, uma vez que os artigos escritos nesta área não abordam claramente alguns aspectos que, apesar de não serem contribuições importantes para a comunidade, são fundamentais para o desenvolvimento dos modelos;

- a falta de experiência na área, devido ao fato da autora não possuir formação aprofundada em física, para abstrair certos detalhes omitidos nos trabalhos já publicados;
- a maneira intuitiva de trabalhar, proporcionada pela física. Neste caso em específico, trata-se tipicamente de experimentação. A partir de um conjunto de leis (fornecidas pela física) e de um conjunto de idéias (proporcionadas pelos trabalhos já existentes), a elaboração de um modelo para simular superfícies deformáveis dá-se através de um processo de experimentação.

Quanto ao método desenvolvido para representar superfícies deformáveis tanto na sua forma estática como em movimento, utilizou-se um modelo contínuo. Para representar graficamente este modelo, é necessária a discretização do mesmo tanto no aspecto espacial como temporal. Nos itens abaixo, é apresentado o modelo discreto de representação utilizado a fim de permitir a simulação geométrica e física de uma superfície deformável.

### 3.1 Modelo Geométrico

A discretização da superfície deformável no espaço dá-se no sentido de que é necessário que a mesma seja graficamente representável. Para tanto, parte-se do princípio que a superfície é definida através de um conjunto de pontos no espaço euclidiano.

Como o objetivo deste trabalho não reside na apresentação de modelos geometricamente complexos, optou-se por restringir a forma das superfícies deformáveis a retângulos. Portanto, o modelo geométrico considerado consiste de uma malha retangular representada por uma matriz de pontos  $M$ , de dimensões  $n \times m$ . Cada elemento  $M_{i,j}$  desta matriz, contém um terno  $(x,y,z)$  de coordenadas, que corresponde a um ponto da superfície no

espaço 3D. Considera-se ainda  $r$  como sendo a distância horizontal e vertical entre os elementos da matriz  $M$  (conforme figura 3.1).

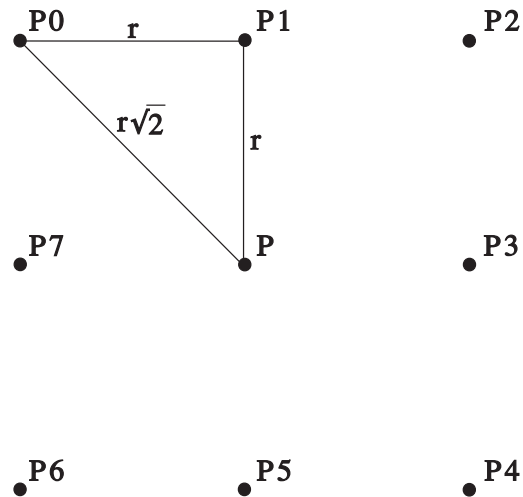


Figura 3.1- Modelo geométrico de uma malha deformável.

A representação ideal de uma superfície deformável ocorre quando a quantidade de pontos que a representa tende ao infinito. No entanto, é evidente que trata-se de uma abordagem impraticável. Como alternativa, é permitida a parametrização da matriz  $M$ , conforme descrito acima. Portanto, para aumentar a densidade de uma superfície, basta aumentar o seu número de linhas e/ou colunas na mesma razão em que se diminui a distância  $r$ .

### 3.2 Modelo Físico

De modo a simular fisicamente uma malha flexível utilizou-se as leis da dinâmica de partículas uma vez que, segundo [RES 84], o movimento de uma partícula é determinado pela natureza e pela disposição de outros corpos, que constituem sua vizinhança. Neste caso específico, considera-se que uma superfície deformável é formada por partículas com densidade de

massa  $m$ , cujo comportamento é definido através da sua interação com o universo e com as demais partículas que formam a superfície. Fazendo a correspondência com o modelo geométrico sugerido, tem-se cada célula da matriz  $M$  como uma partícula.

O modelo físico baseia-se fundamentalmente na aplicação de forças sobre os diversos pontos de massa da malha, gerando novas posições para os mesmos. Através da soma de todas as forças aplicadas, obtém-se uma força resultante para cada ponto da superfície deformável.

Com o objetivo de simplificação, considerou-se aqui a aplicação de 7 forças distintas, sendo elas: força da gravidade, força externa, força de elasticidade, força de curvatura e torção, força de colisão, força de restrição e força de atrito. Desta forma, pode-se dizer que a força resultante em cada ponto da malha, em um instante de tempo  $t$  pode ser calculada da seguinte forma:

$$F_{RESULT} = F_{GRAV} + F_{EXT} + F_{ELAST} + F_{CURV} + F_{COLIS} + F_{RESTRI} + F_{ATRITO}$$

Abaixo, são apresentadas em detalhe as componentes envolvidas no cálculo da força resultante sobre um ponto, além dos detalhes envolvidos na sua definição e cálculo.

### 3.2.1 Força da gravidade

A força da gravidade é calculada para cada ponto de massa da superfície, da seguinte forma:

$$F_{GRAV} = g.m,$$

onde  $g$  representa a aceleração da gravidade e  $m$ , a densidade de massa do ponto.

### 3.2.2 Vetor de força externa

Trata-se de um vetor de força aplicado sobre todos os pontos de massa da superfície com o objetivo de simular uma força qualquer existente no ambiente. Este vetor de força é fornecido diretamente ao sistema e foi incluído no modelo a fim de permitir a simulação de forças como a exercida pelo vento, por exemplo.

### 3.2.3 Elasticidade

Para simular a elasticidade resultante em um ponto  $P_{ij}$  da matriz  $M$ , utilizou-se os conceitos envolvidos na Teoria da Elasticidade, mais precisamente das molas de Hooke. Supôs-se a conexão de cada ponto de massa com seus 8 pontos vizinhos, através de molas (figura 3.2).

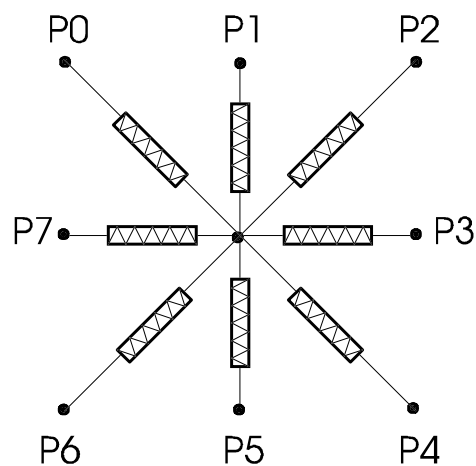


Figura 3.2 - Modelo elástico da superfície.

Sabendo-se que a força exercida por uma mola de Hooke sobre um ponto de massa é

$$F_{MOLA} = -km.(P_f - P_r), \quad (3.1)$$

onde  $km$  é a constante que indica o grau de elasticidade da mola,  $P_f$ , o ponto que representa a sua extremidade fixa e  $P_r$ , o ponto de repouso da mola. Pode-se, por extensão, definir a força de elasticidade como sendo o somatório das forças exercidas pelas molas, no ponto  $P$ :

$$F_{ELAST} = \sum_{i=0}^7 F_{MOLA_i}$$

Fazendo uma relação das equações acima descritas com o modelo elástico exibido na figura 3.2, pode-se dizer que:

- considerando a mola que une o ponto  $P_1$  ao ponto  $P$ , o cálculo da força exercida pela mola sobre o ponto  $P$  dá-se através da aplicação da equação 3.1. Assim, tem-se o ponto  $P_f$  representado por  $P_1$ . Quanto ao ponto de repouso da mola ( $P_r$ ), pode ser calculado da seguinte forma: sabendo-se que a distância vertical e horizontal entre os pontos que compõem a malha flexível é representada pela variável  $r$  (que indica a distância de repouso da mola), o ponto  $P_r$  pode ser encontrado tomando-se o vetor com origem em  $P_1$ , comprimento  $r$  e direção e sentido indicados pelo vetor  $\overline{P_1P}$ ;
- no caso das molas diagonais, a distância de repouso da mola não pode ser indicada pela variável  $r$ . Isto pode ser observado através da disposição física dos pontos. Para determinar uma distância diagonal de repouso ( $r_d$ ), utilizou-se o teorema de Pitágoras, chegando-se a  $r_d = r\sqrt{2}$ .

No que se refere à simulação de materiais reais, pode-se dizer que as suas características elásticas são determinadas através da parametrização das constantes das molas. Por exemplo, para simular uma folha de papel, deve-se utilizar constantes de elasticidade muito grandes para as molas. Já no caso de simular um tecido bastante elástico, as constantes devem possuir valores muito baixos.

O modelo elástico prevê ainda que as molas podem estar divididas em 3 classes: horizontais, verticais e diagonais. A diferenciação permitida entre elas diz respeito unicamente às suas constantes de elasticidade. A título de ilustração, pode-se citar como exemplo a simulação de uma folha de papel crepom, que apesar de ter muito pouca elasticidade no sentido vertical, possui grande capacidade de deformação no sentido horizontal.

### 3.2.4 Força de curvatura e torção

A força aqui descrita tem como objetivo determinar o grau com que uma superfície se curva e torce, formando dobras na mesma. Tal qual a força da elasticidade, esta força também é calculada para cada ponto de massa da malha, em função da posição dos seus 8 pontos vizinhos.

A simulação física deste efeito, foi concebida através da colocação do que se denominou "**molas angulares**". Estas molas foram colocadas entre os vizinhos de um dado ponto  $P$ , da seguinte forma: uma mola no ângulo formado entre os vértices  $P_0, P$  e  $P_4$  (conforme pode ser visto na figura 3.3); uma entre  $P_1, P$  e  $P_5$ ; outra entre  $P_2, P$  e  $P_6$  e uma última entre  $P_3, P$  e  $P_7$ .

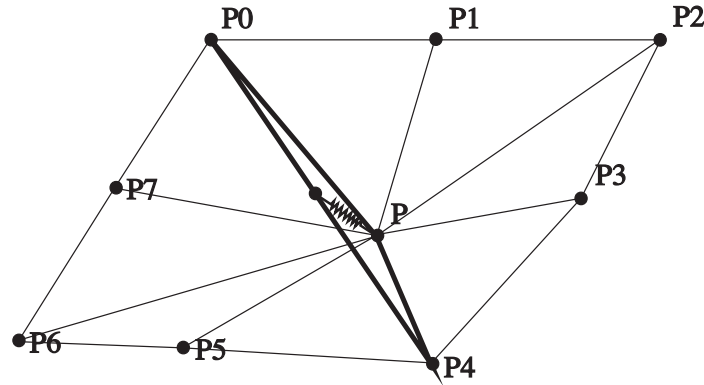


Figura 3.3 - Mola angular entre os pontos  $P_0, P$  e  $P_4$ , com ângulo inferior a  $180^\circ$ .

A implementação dessas molas, dá-se da seguinte forma (supondo o exemplo da mola no ângulo formado entre  $P_0, P$  e  $P_4$ ):

- são traçados 2 vetores com origem no ponto  $P$ , direção e sentido equivalentes aos vetores  $\overline{PP_0}$  e  $\overline{PP_4}$  e comprimento igual à distância inicial entre os pontos da malha (repouso das molas), definida na seção 3.1 pela variável  $r$ ;
- a partir das 2 extremidades dos vetores traçados, traça-se um novo segmento de reta, unindo-as;
- por fim, coloca-se uma mola unindo o ponto  $P$  ao ponto médio do novo segmento. Assume-se que esta "mola angular" está em repouso quando a distância entre as suas extremidades é igual a 0 (zero), ou seja, a mola está em repouso quando o ângulo formado pelos pontos  $P_0, P$  e  $P_4$  é igual a  $180^\circ$ .

Pode-se concluir então, que a definição das características materiais de uma determinada superfície, no que diz respeito a sua capacidade de se curvar e torcer, é determinada através das constantes de elasticidade das "molas angulares". Portanto, para simular uma superfície com um baixo grau de curvatura como, por exemplo, uma folha de papel grosso, deve-se parametrizar as constantes das molas angulares com um



valor bastante alto. Por outro lado, para se obter uma superfície com alto grau de curvatura e torção, como na simulação de uma rede de pescador, onde o ângulo formado entre um determinado ponto  $P$  e seus vizinhos pode tender a zero, é necessário atribuir valores tendendo a zero para as constantes das molas angulares.

### 3.2.5 Colisão

Inicialmente, procurou-se um método que solucionasse o problema da colisão das superfícies deformáveis com outro sólido. A opção foi feita pela aplicação de uma função  $f(i,j)$  sobre cada ponto que compõe a malha, conforme descrito em [NED 92]. Esta função tem como objetivo verificar a situação de um ponto da malha em relação a um sólido: dentro, fora ou na superfície. No caso de estar fora ou na superfície de um sólido, não existe colisão: a força de colisão é nula. No entanto, se for detectado que um ponto da superfície deformável está dentro de um objeto sólido, deve ser aplicada uma força tal que leve o ponto para fora do sólido, percorrendo o mesmo caminho através do qual entrou, porém no sentido oposto. A magnitude desta força é calculada através da utilização dos conceitos da dinâmica inversa. Uma vez que se determina a posição em que o ponto deve ficar, já fora do sólido, torna-se fácil definir qual a força necessária para permitir este movimento.

Com o uso deste tipo de técnica para resolver colisões pode-se gerar efeitos bastante interessantes, como, por exemplo, a colocação de uma toalha sobre uma mesa. No entanto, a técnica acima descrita não permite simular um tipo de situação na qual uma superfície colide com outra. Assim, desenvolveu-se um novo método para calcular a força decorrente da colisão entre os pontos de massa da superfície com ela própria, com sólidos ou ainda com outras superfícies deformáveis ou não.

Considerando que a descrição dos objetos que compõem o mundo virtual dá-se através de faces (no caso das superfícies deformáveis, admite-se que cada 3 pontos de massa, definem um triângulo), a detecção da colisão é executada através de uma função  $f(i,j)$  que verifica para cada vértice da malha, se existe contato com uma face qualquer do universo. Em caso negativo, a força de colisão é nula. Entretanto, uma vez detectada a colisão, procede-se com o seu tratamento.

Neste estudo são consideradas apenas as colisões elásticas, ou seja, aquelas em que há conservação da energia cinética. Para melhor entendimento da metodologia usada para resolver a colisão entre dois corpos, julgou-se adequado fazer uma rápida revisão no capítulo da física que trata a respeito do choque elástico entre partículas, conforme [RES 84].

Considerando-se a colisão entre as partículas de massa  $m_1$  e  $m_2$ , cujas velocidades antes do choque são definidas por  $v_{1i}$  e  $v_{2i}$  e velocidades após o choque determinadas por  $v_{1f}$  e  $v_{2f}$ . Pela conservação do momento linear durante a colisão, tem-se que:

$$m_1 v_{1i} + m_2 v_{2i} = m_1 v_{1f} + m_2 v_{2f} \quad (3.2)$$

Uma vez que o choque é elástico, a energia cinética se conserva e, portanto, pode-se dizer que:

$$\frac{1}{2} m_1 v_{1i}^2 + \frac{1}{2} m_2 v_{2i}^2 = \frac{1}{2} m_1 v_{1f}^2 + \frac{1}{2} m_2 v_{2f}^2 \quad (3.3)$$

Conhecidas as massas e as velocidades iniciais, pode-se facilmente obter as velocidades finais  $v_{1f}$  e  $v_{2f}$ . Assim, reescrevendo as equações 3.2 e 3.3, tem-se:

$$m_1(v_{1i} - v_{1f}) = m_2(v_{2f} - v_{2i}) \quad (3.4)$$

e

$$m_1(v_{1i}^2 - v_{1f}^2) = m_2(v_{2f}^2 - v_{2i}^2) \quad (3.5)$$

Dividindo a equação 3.5 pela 3.4 e supondo  $v_{2f} \neq v_{2i}$  e  $v_{1f} \neq v_{1i}$ ,  
obtem-se

$$v_{1i} + v_{1f} = v_{2f} + v_{2i},$$

ou ainda,

$$v_{1i} - v_{2i} = v_{2f} - v_{1f} \quad (3.6)$$

Para encontrar as velocidades  $v_{1f}$  e  $v_{2f}$ , após o choque, começa-se por isolar a variável  $v_{2f}$  na equação 3.6. Assim,

$$v_{2f} = v_{1i} + v_{1f} - v_{2i}$$

Substituindo na equação do momento (equação 3.4), obtém-se:

$$v_{1f} = \frac{m_1 - m_2}{m_1 + m_2} v_{1i} + \frac{2m_2}{m_1 + m_2} v_{2i}$$

Da mesma forma, levando o valor  $v_{1f} = v_{2f} + v_{2i} - v_{1i}$  (da equação 3.6), na equação 3.4 e resolvendo para  $v_{2f}$ , tem-se:

$$v_{2f} = \frac{2m_1}{m_1 + m_2} v_{1i} + \frac{m_2 - m_1}{m_1 + m_2} v_{2i}$$

A partir da análise destas 2 últimas equações, conclui-se que:

- quando as partículas em colisão têm a mesma massa, ou seja,  $m_1 = m_2$  (caso da colisão de uma superfície consigo mesma), as duas equações anteriores se tornam simples:  $v_{1f} = v_{2i}$  e  $v_{2f} = v_{1i}$ . Isto quer dizer que, em uma colisão elástica entre duas partículas de massas iguais, as partículas simplesmente trocam suas velocidades durante a colisão;
- quando  $m_2$  está em repouso e é muito maior que  $m_1$  (caso previsto para a colisão com sólidos), então  $v_{1f} \cong -v_{1i}$  e  $v_{2f} \cong 0$ . Isto significa que quando uma partícula leve se choca com outra mais pesada, em repouso, a velocidade da partícula leve é praticamente invertida, enquanto que a partícula pesada permanece praticamente em repouso.

Considerando o modelo geométrico definido na seção 3.1, no qual uma malha deformável é representada através de um conjunto de pontos, é preciso admitir uma situação na qual um ponto da malha, na colisão com ela própria, não colide exatamente com outro ponto e sim com uma posição qualquer implícita no modelo.

A fim de contornar esta restrição imposta pelo modelo, foram estudadas 3 diferentes abordagens para a solução deste problema, sendo que nos 3 casos considera-se a malha descrita em termos de triângulos cujos vértices consistem em pontos de massa. Ao detectar a colisão, opta-se por uma das 3 abordagens sugeridas, sendo que são elas:

1. elege-se o vértice do triângulo que está mais próximo do ponto de colisão e assume-se este como sendo o novo ponto de colisão;
2. os 3 vértices do triângulo atingido são considerados como pontos de colisão;

3. conecta-se o ponto de colisão com os vértices do triângulo através de molas em repouso, conforme figura 3.4. Assim, após este sofrer as consequências da colisão, afetará também diretamente os vértices do triângulo.

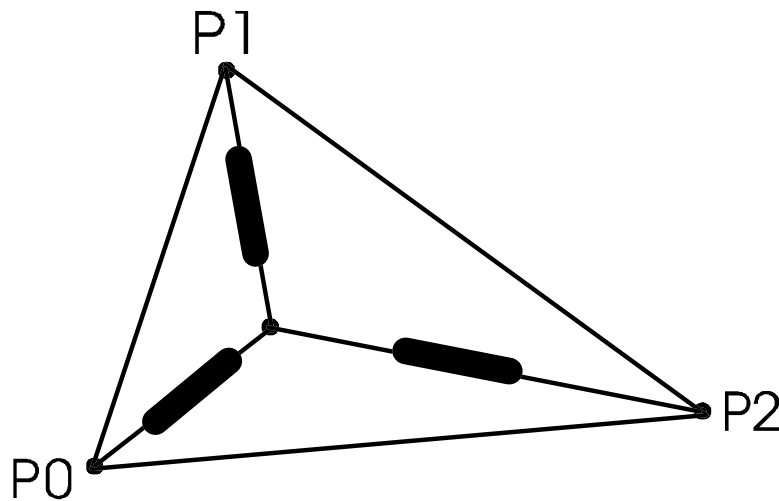


Figura 3.4 - Colisão de uma partícula com um triângulo.

### 3.2.6 Restrições

Com o objetivo de facilitar o controle das formas e, principalmente, do movimento de superfícies deformáveis, foram impostas algumas restrições geométricas ao modelo. São elas:

- a existência de pontos imóveis na grade que representa a superfície, isto é, alguns pontos de massa são fixados a coordenadas no espaço euclidiano;
- a determinação de um ponto do objeto flexível em movimento tornar-se fixo ao atingir determinada posição  $(x,y,z)$ ;
- a fixação de pontos do objeto flexível em pontos de outro corpo. Desta forma, pode-se fixar superfícies deformáveis a corpos

sólidos e simular, por exemplo, uma pessoa vestida e caminhando;

- o movimento cinemático de pontos. Através desta restrição, pode-se transladar um ponto ou um conjunto deles, fazê-los percorrer um determinado caminho, ou ainda rotacioná-los em torno de uma determinada posição no espaço.

A implementação destas funções é feita através da utilização da dinâmica inversa. No caso de ter sido especificada qualquer restrição sobre um dado ponto, é calculada uma força tal que leve o ponto a obedecer a restrição em questão. Entretanto, a metodologia utilizada para atendimento das restrições permite facilmente a inclusão de novas restrições, sem que, no entanto, isto acarrete modificações no modelo físico existente.

Quanto às restrições atualmente previstas, pode-se dizer que foram concebidas com o objetivo de simular situações em que a superfície mantém-se "pendurada" por um ou mais pontos ou movimenta-se em determinada direção. Esta restrição pode ser usada, por exemplo, na simulação de uma toalha pendurada em um prego; uma cortina se fechando; ou ainda, uma bandeira presa a um mastro e tremulando ao vento.

### 3.2.7 Força de atrito

Denomina-se força de atrito a força responsável pelo retardo de um movimento e que atua sobre a força resultante em cada ponto da superfície deformável. A aceleração resultante da força de atrito atua sempre no sentido oposto ao sentido do movimento.

Para definir a força de atrito, é fornecido ao sistema um coeficiente que indica o percentual da força resultante que é anulado pelo

atrito. Neste modelo, assumiu-se como atrito a resistência do ar e principalmente a fricção entre as partes das molas.

## 4 SIMULAÇÃO DO MOVIMENTO

A movimentação de objetos segundo o modelo proposto no capítulo 3 é realizada através da aplicação dos conceitos da dinâmica de partículas. No entanto, para produzir animação baseada em leis físicas, devem ser considerados alguns aspectos bastante importantes. Entre eles, serão abordados aqui, além das leis que regem o movimento, também alguns métodos numéricos utilizados para solucionar essas equações.

### 4.1 Equação do Movimento de uma Partícula

Conforme apresentado anteriormente, o modelo utilizado para simular uma superfície deformável baseia-se em um sistema de partículas que se relacionam com o meio externo e entre si, através de molas. Portanto, a simulação do movimento de uma superfície dá-se através da aplicação das equações de movimento sobre cada uma das partículas que formam o modelo.

O objeto mais simples a ser considerado pela física é a partícula, que corresponde a um ponto de massa. Considerando uma partícula como um ponto, torna-se óbvio que o único movimento relevante a ser considerado é o translacional, sendo o rotacional ignorado. A posição de uma partícula no universo, é indicada pelo vetor  $x$ , sua velocidade pelo vetor  $v$  e a aceleração pelo vetor  $a$ . A densidade de massa de um partícula é indicada pela variável escalar  $m$  e as forças que atuam sobre ela, pelo vetor  $F_i$ . Assim, pode-se definir a equação de movimento para uma partícula, como:

$$ma = m \frac{dv}{dt} = \sum_i F_i$$

$$\frac{dx}{dt} = v$$



As equações de movimento constituem equações diferenciais ordinárias (EDO) de segunda ordem, podendo entretanto serem descritas como equações diferenciais de primeira ordem. Assim, podem ser solucionadas por métodos numéricos, como um sistema de equações.

## 4.2 Métodos Numéricos para EDO de Primeira Ordem

Dada a importância dos problemas envolvidos na solução numérica de equações diferenciais ordinárias, um grande número de métodos têm sido desenvolvidos, conforme pode ser verificado em [CLA 89], [GEA 71], [GRE 91], [PLA 88b] e [PRE 88]. Nesta seção serão apresentados alguns dos métodos comumente utilizados para solucionar sistemas de equações diferenciais de primeira ordem. Com o objetivo de simplificação, os métodos serão descritos em termos de uma equação apenas. No entanto, para solucionar um sistema de equações, basta aplicar o método sobre cada uma das equações envolvidas, em paralelo.

Os métodos descritos propõem a solução de um problema a partir da posse de um valor inicial conhecido:

$$\begin{aligned}\frac{dx}{dt} &= f(x,t) \\ x(t_0) &= x_0\end{aligned}$$

Uma vez que a solução exata para este problema é representada pela função  $g(t)$ , o que se pretende calcular é uma aproximação para esta função no intervalo de tempo  $[t_0, t_n]$ , sendo que este intervalo é ainda subdividido em  $n$  subintervalos. Para cada um desses subintervalos é calculada uma aproximação para a função  $g(t)$ . O tamanho dos subintervalos (representados pela variável  $h$ ) é dado por:

$$h = \frac{(t_n - t_0)}{n}$$

Os métodos começam a funcionar a partir de um valor conhecido para  $x$  quando  $t = t_0$ , gerando uma aproximação  $x_1$  que solucione a equação no tempo  $t = t_0 + h$ . O valor calculado para  $x_1$  é então utilizado no cálculo de  $x_2$  e, assim, sucessivamente, até obter o valor para  $x_n$ . Pode-se, portanto, observar que trata-se de uma aproximação discreta para encontrar  $g(t)$ , uma vez que só são calculados valores para  $g(t)$  a cada intervalo de tempo  $h$  [GRE 91].

#### 4.2.1 Método de Euler

Trata-se do mais antigo e mais conhecido método numérico para solução de equações diferenciais ordinárias. Este método é baseado na série de Taylor e calcula, a partir de um valor  $x_i$  conhecido, uma solução para  $x_{i+1}$ , de acordo com a seguinte equação:

$$x_{i+1} = x_i + h \cdot f(x_i, t_i)$$

No que diz respeito à precisão do método de Euler, pode-se dizer que é inversamente proporcional ao tamanho do intervalo de tempo  $h$ . Quando o espaço de tempo  $h$  for pequeno, o método pode ser considerado instável; no entanto, com  $h$  alto, os valores das variáveis envolvidas na equação crescem exponencialmente [PLA 88b].

Uma das mais importantes técnicas em métodos numéricos para EDO é o controle automático do espaço de tempo  $h$ . Esta técnica pode ser ilustrada considerando o método de Euler, onde o terceiro termo da série de Taylor é:

$$\frac{h^2}{2} f'(g(t_i), t_i)$$

Este termo pode ser usado como uma medida do erro no cálculo de  $x_{i+1}$ . Pode-se aproximar  $f'$  numericamente, com base no valor calculado de  $x_{i+1}$  (usando diferenças numéricas) [GRE 91]. Então, o erro estimado pode ser comparado com o erro máximo admitido em cada intervalo de tempo. A diferença entre estes dois valores pode ser usada para calcular o máximo espaço de tempo  $h$ , que irá produzir um erro aceitável. Obviamente, o máximo espaço de tempo calculado, deve ser inversamente proporcional ao valor indicado por  $f'$ . Assim, o cálculo de  $h$ , considerando  $max\_erro$  como o erro máximo admitido, é calculado através da equação:

$$h = \sqrt{\frac{2 \cdot max\_erro}{f'}}$$

A cada passo do algoritmo, esta expressão é calculada a fim de determinar o tamanho do próximo espaço de tempo. Entretanto, se o erro no passo corrente é maior do que  $max\_erro$ , o passo pode ser recalculado com um  $h$  menor.

## 4.2.2 Método de Runge-Kutta

Trata-se de um dos métodos mais populares na resolução de EDOs. Sua popularidade deve-se, em geral, a sua aplicabilidade e não à sua eficiência, segundo [GRE 91]. A aproximação utilizada neste método consiste em calcular diversos valores para  $f$  em cada espaço de tempo  $h$ . Enquanto no método de Euler,  $f$  é calculado apenas no início de cada intervalo de tempo  $h$ , no Runge-Kutta, é calculado no mínimo duas vezes.

O método mais comum de Runge Kutta, também conhecido como método clássico [GEA 71], consiste em calcular para cada intervalo de tempo:

$$\begin{aligned}k_1 &= h \cdot f(x_i, t_i) \\k_2 &= h \cdot f\left(x_i + \frac{k_1}{2}, t_i + \frac{h}{2}\right) \\k_3 &= h \cdot f\left(x_i + \frac{k_2}{2}, t_i + \frac{h}{2}\right) \\k_4 &= h \cdot f(x_i + k_3, t_i + h) \\x_{i+1} &= x_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)\end{aligned}$$

Da mesma forma que no método de Euler, em Runge Kutta não é obrigatório que o intervalo de tempo  $h$  seja constante. Em [PRE 88], descreve-se um algoritmo para controle adaptativo do intervalo de tempo.

### 4.2.3 Métodos de múltiplos passos

Nos métodos anteriormente descritos, para determinar  $x_{i+1}$  considerava-se apenas o valor de  $f$  no ponto  $x_i$ , ou seja, trabalhava-se com métodos de um só passo. Na realidade, para calcular  $x_{i+1}$  pode-se levar em conta não só  $x_i$ , mas também os pontos anteriores,  $x_{i-1}, x_{i-2}, x_{i-3}, \dots$ . Estes métodos são denominados de métodos de múltiplos passos [CLÁ 89].

Um dos mais conhecidos métodos de múltiplos passos é o de Adams-Bashforth, cuja forma é descrita abaixo:

$$\begin{aligned}x_{i+1} &= x_i + \frac{h}{24}(55f_i - 59f_{i-1} + 37f_{i-2} - 9f_{i-3}) \\f_i &= f(x_i, t_i)\end{aligned}$$

Como se pode verificar, a utilização deste método implica em que deve-se armazenar sempre 3 valores passados para  $f$ . Em cada passo do

algoritmo, é calculado um novo valor para  $f$ , sendo o último dentre os armazenados, descartado.

Um dos problemas identificados neste método é o custo computacional para armazenar os valores passados de  $f$ . Outro problema é que mesmo para calcular o primeiro valor para  $f$ , necessita-se de valores anteriores. Neste caso, pode-se utilizar outro método como, por exemplo, o Runge-Kutta para calcular os 3 primeiros valores para  $f$  [GRE 91].

#### 4.2.4 Métodos preditores-corretores

Para melhor entender o funcionamento dos métodos preditores-corretores é preciso, em primeiro lugar, salientar a diferença entre um método explícito e um método implícito. Os métodos explícitos são aqueles que utilizam as informações contidas no ponto anterior para calcular o novo ponto  $x_{i+1}$ . Como exemplo, pode-se citar os métodos apresentados nas seções 4.2.1, 4.2.2 e 4.2.3. Já os métodos implícitos, podem ser considerados bastante mais estáveis uma vez que para calcular o ponto  $x_{i+1}$ , utiliza as informações do próprio ponto [PLA 88b]. Por exemplo, a equação que calcula o valor de  $x_{i+1}$  através do método de Euler implícito, é dada por:

$$x_{i+1} = x_i + h \cdot f(x_{i+1}, t_{i+1})$$

Como vantagens dos métodos implícitos, pode-se destacar que: usam menos informações do que os métodos explícitos; os coeficientes de  $f_i$  são menores e, portanto, são fórmulas menos sensíveis ao arredondamento e os limites do erro de truncamento são menores [CLA 89].

Apesar de possuírem todas essas vantagens, os métodos implícitos possuem um grande problema. Para calcular  $x_{i+1}$ , precisa-se do valor de  $f_{i+1}$ , o que torna o método numericamente inaplicável.

Os métodos preditores-corretores são métodos de múltiplos passos, que atuam através da combinação de um método explícito com um implícito, objetivando obter as vantagens de cada método e evitando suas desvantagens [CLA 89]. Seu funcionamento baseia-se nos seguintes passos:

- calcular o valor de  $x_{i+1}$  através de um método explícito (trata-se do método preditor);
- de posse do valor previsto, calcular  $f_{i+1}$ ;
- recalculer  $x_{i+1}$  utilizando um método implícito (método corretor).

Um dos métodos mais utilizados como corretor é a fórmula de Adams-Moulton, apresentada abaixo:

$$x_{i+1} = x_i + \frac{h}{24}(9f_{i+1} + 19f_i - 5f_{i-1} + f_{i-2})$$

### 4.3 O Problema da Descontinuidade

Muitas das técnicas de controle utilizadas na animação por computador introduzem descontinuidades nas equações de movimento. Como exemplo, pode-se citar o caso das colisões ou, ainda, a utilização de molas para simular a elasticidade. No caso da colisão, a descontinuidade dá-se através da mudança rápida de forças e torques aplicados sobre um objeto, após a colisão. Já no caso da utilização de molas, pode-se verificar a descontinuidade a cada vez em que uma mola totalmente distendida ou contraída inverte o sentido de seu movimento. O problema da descontinuidade tem sido objeto de estudo [GEA 84] devido à sua importância, pois, em geral, as descontinuidades tornam os métodos numéricos instáveis.

Existem dois problemas importantes associados às descontinuidades em equações diferenciais. O primeiro é na sua detecção, ou seja, como detectar onde e quando ela acontecerá. Quanto ao segundo problema, trata-se de como resolver numericamente uma descontinuidade. Normalmente, opta-se por integrar a equação até o ponto onde se detecta a descontinuidade e, em seguida, recomeçar a integração após este ponto [GRE 91].

Entretanto, o tratamento da descontinuidade envolve um estudo bastante aprofundado dos métodos de integração numérica. Por não ser o principal objetivo deste trabalho, não foi previsto no cálculo do movimento.

#### 4.4 Algoritmo de Geração do Movimento

Esta seção tem como objetivo descrever o procedimento usado para gerar o movimento das partículas que compõem as superfícies deformáveis. O método utilizado para integrar numericamente as equações diferenciais ordinárias descritas na seção 4.1 foi o método de Euler explícito com espaço de tempo  $h$  constante. A opção por este método foi feita em vista de sua simplicidade e facilidade de implementação.

Abaixo, descreve-se o algoritmo utilizado na geração das posições dos pontos de massa da malha, após cada intervalo de tempo especificado ( $h$ ).

- Para um tempo  $t$ , inicialmente igual a  $0$  e variando de  $h$ :
  - calcular a força resultante em cada ponto da malha flexível, através dos métodos descritos no capítulo 3 (os detalhes deste passo estão descritos na seção 5.3.3);
  - conhecidas as forças resultantes e a densidade de massa de cada ponto, através da aplicação das leis de Newton ( $F=m.a$ ,

onde  $m$  é a densidade de massa e  $a$  a aceleração), calcular a aceleração em cada ponto;

- com base nesta aceleração, calcular a velocidade dos pontos através da fórmula:

$$v_{i+1} = v_i + \frac{d^2x_i}{dt^2}, \text{ onde } v_i \text{ representa a velocidade no momento anterior e } \frac{d^2x}{dt^2}, \text{ a aceleração;}$$

- dada a velocidade, encontra-se a nova posição do ponto, da seguinte forma:

$$x_{i+1} = x_i + \frac{dx_i}{dt}, \text{ onde } x_i \text{ é a posição anterior do ponto e } \frac{dx_i}{dt}, \text{ a sua velocidade.}$$

Quanto ao problema de instabilidade numérica do método de Euler, agravada neste caso pela grande descontinuidade gerada pela utilização de molas no modelo, a solução proposta foi a redução do espaço de tempo  $h$ , sendo que para cada nova cena da animação gerada, são calculadas diversas posições  $x_i$ . Uma vez que o método torna-se estável quando  $h$  tende a zero, julgou-se apropriado apresentar alguns resultados obtidos. Desta forma pode-se verificar visualmente os erros inseridos no cálculo das novas posições  $x_i$  a cada passo do algoritmo.

Foi utilizada uma mola longitudinalmente posicionada sobre o eixo  $x$ . Assumiu-se um sistema massa-mola com as seguintes características: uma das extremidades da mola fixa na posição  $x = 0$ ; na outra extremidade existe uma partícula com massa igual a 1; a mola está na sua posição de repouso quando a partícula citada está na posição  $x = 2$ ; a constante de elasticidade da mola é igual a 1. A fim de demonstrar a simulação do movimento deste sistema considerou-se a partícula posicionada em  $x = 4$  (ou seja, a mola distendida) e com velocidade inicial igual a 0.



Na tabela 4.1 são apresentadas simulações do movimento do sistema massa-mola, considerando-se um tempo total igual a 9 unidades. Na primeira coluna apresenta-se o instante de tempo  $t_i$ ; na segunda, a posição da partícula no momento  $t_i$ , considerando-se um espaço de tempo  $h = 1$  para o cálculo da nova posição; nas demais, apresentam-se os resultados, considerando-se  $h = 0.5$ ,  $h = 0.05$ ,  $h = 0.005$ ,  $h = 0.0005$  e  $h = 0.00005$ .

Analisando os resultados na tabela 4.1 e na figura 4.1, que mostra graficamente os valores apresentados na tabela, pode-se verificar facilmente que quanto menor o intervalo  $h$ , menor será o erro numérico introduzido pelo método.

Tabela 4.1 - Comportamento do algoritmo relativo a um sistema massa-mola em diversas condições diferentes.

$t_i$	$P_i$					
	$h=1.0$	$h=0.5$	$h=0.05$	$h=0.005$	$h=0.0005$	$h=0.00005$
0	4.0000	4.0000	4.0000	4.0000	4.0000	4.0000
1	4.0000	3.5000	3.1093	3.0833	3.0808	3.0806
2	2.0000	1.1250	1.1282	1.1635	1.1672	1.1676
3	-2.0000	-1.6562	-0.1332	0.0051	0.0185	0.0198
4	-6.0000	-2.1171	0.5498	0.6795	0.6914	0.6925
5	-6.0000	1.5371	2.6337	2.5743	2.5680	2.5673
6	2.0000	7.7387	4.2274	3.9493	3.9232	3.9206
7	18.0000	11.3314	3.8048	3.5345	3.5104	3.5080
8	34.0000	7.0303	1.6607	1.7032	1.7084	1.7089
9	34.0000	-5.0348	-0.2736	0.1363	0.1736	0.1773
10	2.0000	-16.4121	-0.1656	0.2792	0.3176	0.3214

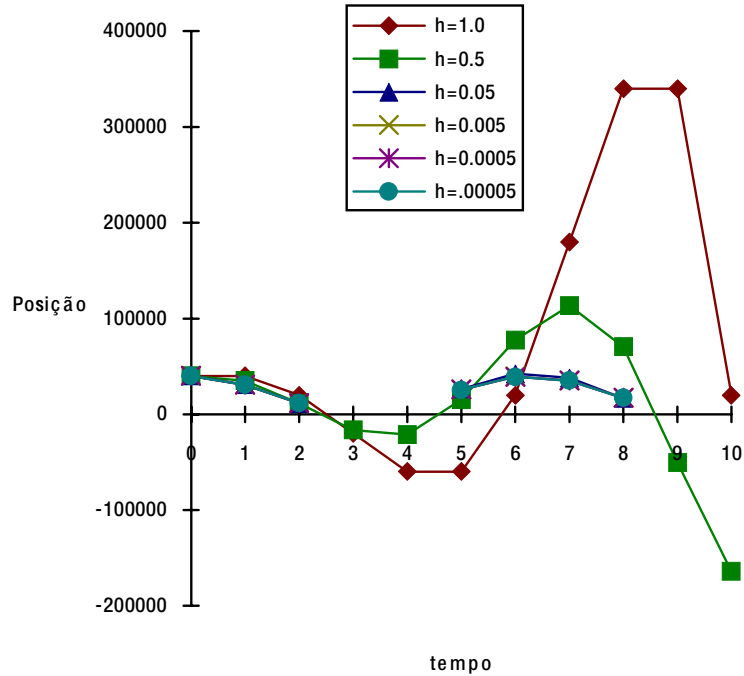


Figura 4.1 - Gráfico posição x tempo para simulação do movimento em um sistema massa-mola.

Na figura 4.2 são apresentados alguns resultados de simulações geradas a partir de uma malha deformável com as mesmas características paramétricas, inseridas dentro de universos com características idênticas sendo que a diferença fundamental entre as sequências consiste no tamanho do intervalo de tempo  $h$  utilizado na sua geração. Com o objetivo de ilustrar o problema da precisão numérica acima discutido, utilizou-se o quadro da animação de número 12 do primeiro exemplo apresentado no anexo A-2.

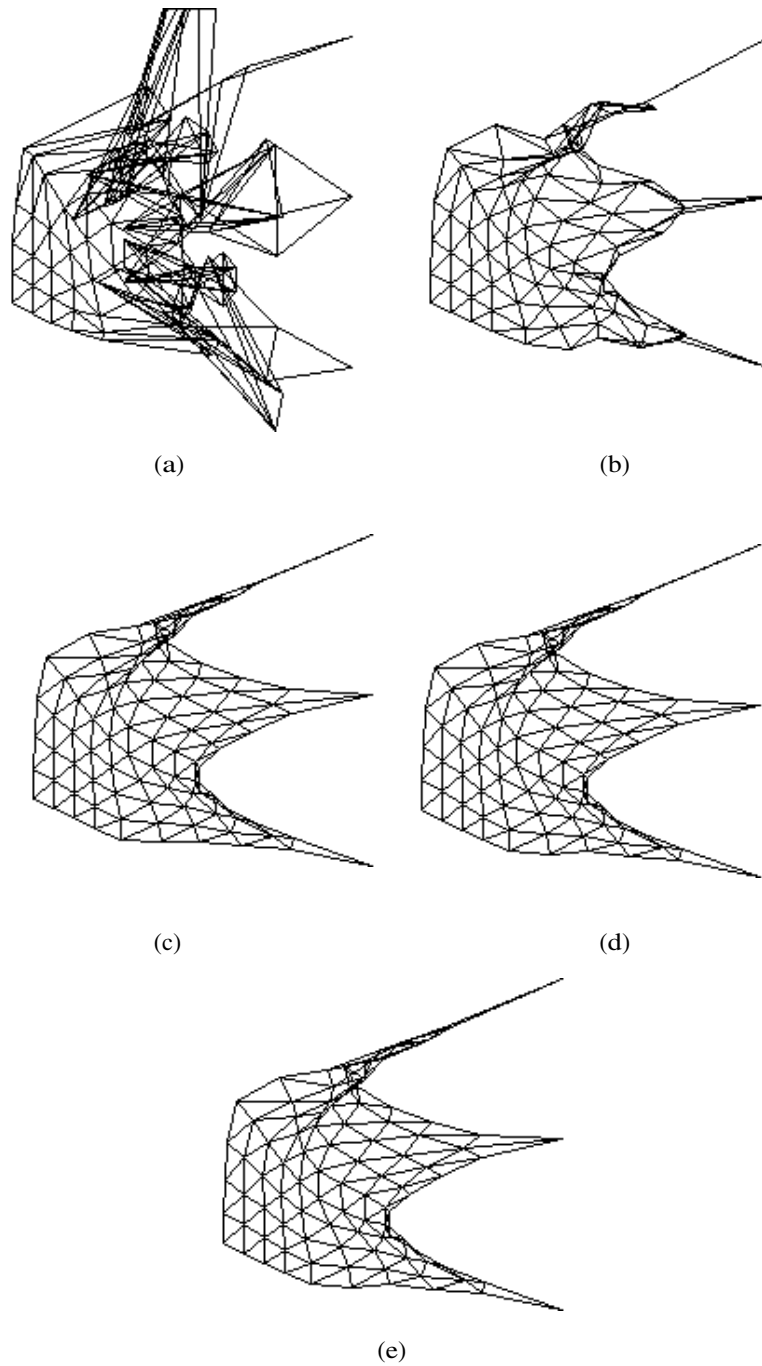


Figura 4.2 - Resultado da animação de uma malha com parâmetros idênticos e  $h$  diferente. (a) quadro gerado considerando  $h = 0.1$  ; (b)  $h = 0.05$ ; (c)  $h = 0.01$ ; (d)  $h = 0.005$ ; (e)  $h = 0.001$ .

Na figura 4.2a pode-se perceber facilmente que o erro introduzido na animação através do cálculo numérico é bastante grande, uma vez que não se consegue perceber a forma real do objeto. Já na figura 4.2b, apesar do erro ainda ser grande, já é possível vislumbrar a forma aproximada do objeto. Observando as figuras 4.2c, 4.2d e 4.2d verifica-se que existe uma diferença visual muito pequena entre elas. Assim, pode-se concluir que, para este quadro da animação, o método utilizado na geração do movimento é aceitável para  $h = 0.01$ .

## 5 IMPLEMENTAÇÃO DO PROTÓTIPO FLEX3D

Neste capítulo é apresentado o protótipo FLEX3D desenvolvido com o objetivo de verificar e validar o modelo desenvolvido e apresentado nos capítulos 3 e 4. Serão abordados, em primeiro lugar, a organização do protótipo e, em seguida, alguns detalhes bastante significativos no que diz respeito à sua implementação. Dentre estes, será descrita a estratégia utilizada para detectar uma colisão e a linguagem de descrição da animação desenvolvida com o objetivo de parametrizar tanto a forma de uma superfície deformável, como o seu movimento.

### 5.1 Organização do Protótipo

O protótipo FLEX3D (modelagem e animação de superfícies FLEXíveis no espaço 3D) tem como objetivo, a partir de um arquivo contendo os parâmetros necessários para modelar e animar uma dada superfície, simular o comportamento de uma superfície deformável através do uso da análise dinâmica. A apresentação dos resultados é feita através de arquivos contendo as informações geométricas dos objetos de uma cena a cada intervalo de tempo.

Portanto, pode-se dizer que sua execução é dividida em três etapas (figura 5.1):

- Modelagem das superfícies deformáveis;
- Geração do movimento das superfícies;
- Geração dos arquivos de saída.

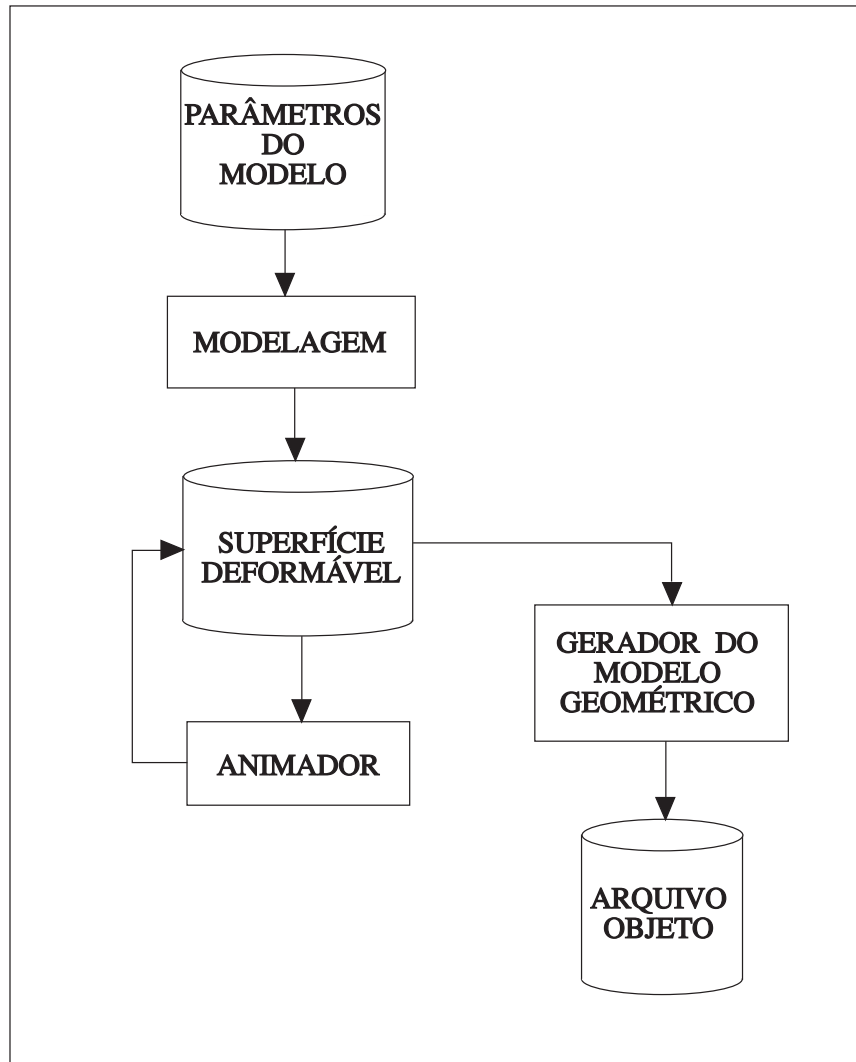


Figura 5.1 - Etapas do protótipo FLEX3D.

## 5.2 Parametrização do Modelo

No que diz respeito à entrada de dados do FLEX3D, desenvolveu-se uma linguagem com o objetivo de parametrizar o comportamento das superfícies deformáveis, bem como as características do universo no qual estão inseridas. Trata-se, em parte, do roteiro da animação. Abaixo, serão apresentadas sem muito rigorismo formal a sintaxe e a

semântica desta linguagem, respeitando algumas convenções: trechos entre colchetes e seguidos de asterisco podem ser repetidos inúmeras vezes; trechos entre < > devem ser substituídos por parâmetros escolhidos pelo animador.

### 5.2.1 Sintaxe

DURAÇÃO = <tempo da animação em segundos>  
 QUADROS\_SEG = <número de quadros por segundo>  
 DELTA\_t = <intervalo de tempo para cálculo da cena>  
 FORÇA\_EXTERNA (<fx> <fy> <fz>)  
 ATRITO = <coeficiente de atrito>  
 OBJETOS\_NA\_CENA = <número de objetos na cena>  
 [<arquivo\_obt>]\*  
 NUM\_SUPERFÍCIES = <número de superfícies modeladas>  
 SUPERFÍCIE <nome da superfície>  
     MALHA (<m> <n>)  
     DENSIDADE = <distância inicial entre os pontos>  
     POSIC\_NO\_SRU (<x> <y> <z>)  
     VETOR\_NORMAL (<x> <y> <z>)  
     MASSA = <densidade de massa dos pontos>  
     CONST\_ELASTICIDADE (<kh> <kv> <kd>)  
     CONST\_CURVATURA = <kc>  
     RESTRIÇÕES  
         PONTOS\_FIXOS = <número de pontos fixos>  
         [<i> <j>]\*  
         TRANSLAÇÃO (<tx> <ty> <tz>)  
         INI\_TRANSL = <tempo inicial de translação>  
         FIM\_TRANSL = <tempo final de translação>  
     FIM\_RESTRIÇÕES  
 FIM\_SUPERFÍCIE



## 5.2.2 Semântica

- **DURAÇÃO:** Indica o tempo total da animação, em segundos;
- **QUADROS\_SEG:** Indica o número de quadros a serem apresentados em cada segundo da animação. A existência deste parâmetro tem como objetivo a visualização em tempo real de uma animação. Por exemplo, dependendo das características da superfície animada, pode-se visualizá-la em tempo real através da apresentação de 10 quadros por segundo. Por outro lado, para efetuar sua gravação em vídeo é necessária a produção de 30 quadros por segundo;
- **DELTA\_t:** Trata-se do intervalo de tempo considerado na produção de cada cena. Por exemplo, dada uma animação na qual definiu-se a apresentação de 2 quadros por segundo. Neste caso, é preciso gerar uma cena a cada meio segundo. No entanto, o intervalo de tempo de 0,5 segundos para o cálculo de cada cena, é muito grande e resultaria em uma superfície com um número muito grande de erros, devido ao método numérico utilizado no seu cálculo, conforme já foi visto no capítulo 4. Portanto, é permitido ao usuário definir separadamente o número de cenas a serem calculadas e o número de cenas que devem ser apresentadas. Isto significa que pode-se calcular milhares de cenas, enquanto apenas algumas dezenas delas são apresentadas;
- **FORÇA\_EXTERNA:** Indica, através das coordenadas ( $f_x$ ,  $f_y$ ,  $f_z$ ), um vetor de força externa existente no universo onde ocorrerá a deformação. Usada normalmente para simular a força do vento;
- **ATRITO:** Indica o coeficiente de atrito do universo. Funciona como um percentual sobre o vetor de força resultante, como pode ser visto na seção 3.2.7;

- **OBJETOS\_NA\_CENA:** Através deste parâmetro pode-se indicar ao sistema, o número de objetos externos presentes na animação, bem como os nomes dos seus arquivos de descrição (apresentados no anexo A-1). Entende-se por objetos externos aqueles cujo movimento não é modelado pelo FLEX3D. Uma restrição atual do sistema, é que não é permitida a movimentação destes objetos;
- **NUM\_SUPERFÍCIES:** Indica o número de superfícies que serão animadas;
- **SUPERFÍCIE:** Este parâmetro indica a presença dos parâmetros referentes a uma superfície deformável. Primeiramente, deve ser associado um nome à superfície, nome este que será utilizado na geração dos arquivos que descrevem a mesma. Feito isto, são apresentados os demais parâmetros que definem a superfície. Os dados referentes a uma mesma superfície são delimitados inicialmente pela palavra reservada **SUPERFÍCIE** e no final por **FIM\_SUPERFÍCIE**;
- **MALHA:** Indica o tamanho da matriz onde será armazenada a malha deformável. Possui  $m$  linhas por  $n$  colunas;
- **DENSIDADE:** Define a distância existente entre os pontos da malha, no instante inicial da animação;
- **POSIC\_NO\_SRU:** Indica a posição da malha no sistema de referência do universo através das coordenadas  $(x,y,z)$ ;
- **VETOR NORMAL:** Tem como objetivo indicar o vetor normal à superfície. Dessa forma, pode-se indicar qual a orientação da superfície em seu estado inicial;
- **MASSA:** Define a densidade de massa dos pontos que formam a superfície;
- **CONST\_ELASTICIDADE:** Define as constantes de elasticidade das molas apresentadas na seção 3.2.3. Considera-

se o parâmetro  $kh$  como a constante das molas horizontais,  $k_v$ , das verticais e  $kd$ , das diagonais;

- **CONST\_CURVATURA**: Indica, através do parâmetro  $kc$ , a constante de elasticidade das molas angulares descritas na seção 3.2.4;
- **RESTRICÇÕES**: Esta palavra reservada, bem como a palavra **FIM\_RESTRICÇÕES** são usadas como delimitadores. Entre estes delimitadores são indicadas todas as restrições definidas sobre a superfície;
- **PONTOS\_FIXOS**: Informa quais os pontos da malha que devem ficar fixos, ou seja, quais os pontos que não se movem de acordo com as leis físicas;
- **TRANSLAÇÃO**: Indica através do vetor  $(tx,ty,tz)$  o quanto os pontos fixos já determinados devem se translacionar entre cada quadro da animação. Quando o vetor  $(tx,ty,tz)$  corresponder a  $(0,0,0)$ , então não há translação dos pontos;
- **INI\_TRANSL**: Indica o momento da animação em que o movimento de translação deve iniciar. Da mesma forma, a palavra reservada **FIM\_TRANSL** indica o momento em que os pontos devem parar de ser translacionados.

Através da observação da sintaxe e da semântica da linguagem desenvolvida, pode-se verificar que esta não prevê todas as restrições definidas na seção 3.2.6. Da mesma forma, pode-se constatar que poderiam ser exploradas mais capacidades do que as previstas atualmente. No entanto, a linguagem disponível permite a geração de movimentos suficientes para validar o modelo apresentado. No anexo A-2 são apresentados alguns exemplos de roteiros de animação, bem como algumas cenas correspondentes aos mesmos.

### 5.3 Estratégia de Detecção do Choque

Diversos métodos têm sido estudados para detectar a colisão entre dois corpos em movimento. Este processo não é muito simples uma vez que, normalmente, a modelagem do movimento de objetos dá-se através da discretização de suas posições intermediárias, no tempo. Desta forma, a cada intervalo de tempo  $\Delta t$  tem-se uma nova posição para cada um dos objetos envolvidos na animação. O que ocorre é que a colisão pode acontecer no meio de um intervalo  $\Delta t$ , ou seja, num momento em que não existe uma posição discretizada para os objetos em questão. Dentre os estudos realizados, pode-se salientar os apresentados em [BAF 90], [BAF 92] e [MOO 88].

Para detectar colisões de superfícies flexíveis no FLEX3D, admite-se que as superfícies são modeladas através de uma grade de pontos, os quais são conectados formando triângulos. Da mesma forma, outros objetos que compõem a cena devem ser descritos através de triângulos. As colisões são detectadas testando-se a intersecção entre cada vértice da superfície deformável com todos os triângulos da cena dos quais o vértice em questão não faz parte.

Inicialmente, assume-se que não existem colisões de nenhuma espécie. Em seguida, para cada intervalo de tempo  $\Delta t$ , as posições de um ponto no início e no final do intervalo de tempo são comparadas com os pontos internos de um triângulo a fim de verificar se alguma das posições do ponto em questão, em algum instante durante o intervalo  $\Delta t$ , pertence ao triângulo analisado. Desta forma verifica-se a ocorrência ou não de uma colisão, dentro deste intervalo de tempo.

O algoritmo utilizado na detecção de colisão é descrito em [BAD 90] e largamente utilizado nas técnicas de *ray tracing*. Trata-se de um algoritmo que detecta o ponto de intersecção entre um raio e um polígono. Abstraindo para o problema em questão, tem-se os polígonos como sendo os

triângulos anteriormente mencionados e o raio definido pelas posições do ponto antes e depois do intervalo de tempo  $\Delta t$ . O algoritmo é dividido em duas etapas, onde a primeira determina se um raio intercepta ou não um polígono. Em caso afirmativo, passa-se para a segunda etapa onde se determina qual o ponto de intersecção.

## 5.4 Estruturas de Dados

O protótipo FLEX3D conta com duas estruturas de dados fundamentais. A primeira delas consiste de uma tabela (representada na figura 5.2) responsável pelo armazenamento de todos os triângulos que compõem os objetos externos ao sistema, descritos na seção 5.2.2. A estrutura de dados possui informações do número de triângulos existentes, do número de vértices, além de uma lista contendo referência a estes triângulos e outra, com as coordenadas  $(x,y,z)$  dos vértices utilizados para defini-los.

<b>Número de triângulos</b>	
<b>Número de vértices</b>	
<b>TRIÂNGULOS</b>	v1, v2, v3
<b>VÉRTICES</b>	x, y, z

Figura 5.2 - Lista de triângulos.

A segunda estrutura de dados consiste de uma tabela contendo os dados referentes a cada uma das superfícies que estão sendo animadas (figura 5.3). Contém a densidade dos pontos na malha, a posição do centro da superfície em relação ao sistema de coordenadas do universo, o vetor normal à superfície, a densidade de massa dos pontos da superfície, as constantes de

elasticidade das molas, os dados a respeito do movimento de translação dos pontos de restrição, o comprimento e a largura da malha, a lista dos vértices que compõem a malha, a lista das velocidades de cada um dos pontos e uma lista contendo os pontos de restrição da malha.

<b>S U P E R F Í C I E S</b>	densidade	
	coordenadas do centro da superfície	
	vetor normal à superfície	
	massa	
	constantes $k_h$ , $k_v$ , $k_d$	
	constante $k_c$	
	passo da translação	
	ini_transl	fim_transl
	comprimento	largura
	lista de vértices da malha no instante $t$	
	lista de velocidades dos pontos no instante $t$	
	lista dos pontos de restrição	
	lista de vértices da malha no instante $t + \Delta t$	
	lista de velocidades no instante $t + \Delta t$	

Figura 5.3 - Dados referentes às superfícies.

Os arquivos de dados manipulados pelo FLEX3D são de dois tipos. O primeiro deles possui a extensão .PAR e armazena o roteiro da animação, ou seja, os parâmetros que definem as superfícies deformáveis. Já o segundo tipo possui extensão .OBT e apresenta a descrição geométrica de

um objeto, para ser lida e apresentada por um sistema de visualização compatível com este formato de arquivo.

## 5.5 Modelagem das Superfícies

Na primeira etapa do FLEX3D é feita a modelagem das superfícies deformáveis em seu estado inicial. Primeiramente, é efetuada a leitura do roteiro de animação (descrito em um arquivo com extensão .PAR). Em seguida, em função dos parâmetros lidos, são construídas as superfícies. Este processo pode ser melhor descrito através do algoritmo abaixo:

- Leitura do roteiro da animação;
- Para cada superfície descrita no roteiro:
  - Através dos parâmetros que indicam as dimensões da malha, a densidade dos pontos e o vetor normal à superfície, encontrar as coordenadas iniciais dos vértices que definem uma superfície. Supor o centro da malha nas coordenadas (0,0,0) do objeto;
  - Para cada vértice da malha, considerar velocidade inicial igual a zero;
  - Para cada ponto da malha identificado como ponto-fixa, inicializar uma variável indicando a existência da restrição.

## 5.6 Geração do Movimento

O procedimento envolvido na geração do movimento, envolve o processamento iterativo sobre a estrutura de dados que armazena os dados referentes à superfície deformável, como pode-se verificar na figura 5.1. A cada intervalo de tempo  $\Delta t$  é gerada uma nova cena, de acordo com o seguinte algoritmo:

- Com base no tempo total da animação e no intervalo de tempo definido por  $\Delta t$ , calcular o número total de cenas a serem produzidas.
- Para cada cena da animação:
  - Para cada superfície descrita:
    - Para cada ponto de uma superfície:
      - Calcular a força da elasticidade;
      - Calcular a força de curvatura e torção;
      - Calcular a força resultante da colisão;
      - Calcular a força de restrição;
      - Calcular o atrito;
      - Calcular o vetor de força resultante;
      - Calcular a aceleração;
      - Calcular a velocidade;
      - Calcular a nova posição do ponto;

## 5.7 Geração dos Arquivos de Saída

A geração dos arquivos de saída é feita de acordo com os parâmetros referentes aos tempos. Através dos parâmetros que indicam o tempo total da animação, o número de quadros por segundo e o intervalo de tempo existente entre cada cálculo de uma nova cena, pode-se calcular de quanto em quanto tempo devem-se gravar os arquivos contendo a descrição das superfícies.

É importante descrever ainda, a lei de formação dos nomes dos arquivos de saída. Conforme já foi descrito anteriormente, o sufixo dos arquivos é dado pela extensão .OBT. No que diz respeito ao nome, é composto



pelo identificador da superfície, seguido pelo caracter ( ) e finalmente pelo número de ordem da cena na animação. Por exemplo, o arquivo que descreve a geometria da superfície denominada PANO no quadro de número 124 da animação, denomina-se PANO\_124.OBT.

## 5.8 Linguagem de Programação e Equipamentos Utilizados

A linguagem de programação utilizada na implementação do protótipo foi "C", principalmente por ser uma linguagem com grande capacidade de portabilidade.

Inicialmente, desenvolveu-se um protótipo, denominado **FLEX2D**, em equipamento IBM-PC 386 cujo objetivo era modelar e animar uma superfície deformável em um universo bi-dimensional. O **FLEX2D** foi desenvolvido unicamente com o objetivo de validar os estudos feitos no que diz respeito ao modelo elástico descrito na seção 3.2.3. Por este motivo, não foi anteriormente mencionado.

Quanto ao **FLEX3D**, foi implementado primeiramente em uma estação de trabalho Silicon Graphics - Indigo, por ser, dentre os equipamentos disponíveis na época em que o trabalho começou a ser feito, o que apresentava melhor desempenho no que diz respeito a velocidade de processamento gráfico. Em seguida, com o objetivo de compatibilizar o **FLEX3D** com as outras ferramentas de animação desenvolvidas no Instituto de Informática da UFRGS, foi portado para estações de trabalho SUN.

Por fim, objetivando finalmente obter resultados bastante rápidos, portou-se o **FLEX3D** para o supercomputador CRAY, no Centro de Supercomputação da UFRGS.

## 5.9 Apresentação dos Resultados

A apresentação dos resultados é feita através da visualização dos arquivos com a descrição geométrica dos objetos modelados e animados. O FLEX3D, conforme citado anteriormente, não possui um módulo para permitir esta visualização. Entretanto, gera arquivos com descrição geométrica em formato reconhecido pelo sistema de animação ANIMAKER [LOS 92], desenvolvido na UFRGS em estações de trabalho SUN. Atualmente, está em desenvolvimento uma versão do ANIMAKER executável em equipamentos Silicon Graphics. No anexo A-1 é apresentado o formato do arquivo de descrição geométrica de objetos utilizado.

## 6 CONCLUSÕES

### 6.1 Análise do Trabalho

O trabalho apresentou um modelo que permite a deformação de objetos, com base em leis físicas. Este modelo foi construído a partir do estudo da dinâmica de partículas, bem como das molas ideais de Hooke. Sintetizando, trata-se de um modelo baseado em forças tais como a exercida pelo campo gravitacional e as determinadas pelas características físicas do objeto (como por exemplo a elasticidade e o grau de curvatura e torção). Utilizou-se um método baseado em forças para determinar dinamicamente a configuração geométrica de superfícies com características elásticas.

Admitiu-se ainda a presença de restrições geométricas no modelo, como a existência de pontos fixos na superfície e a movimentação destes pontos através da cinemática. No que diz respeito à interação dessas superfícies com o mundo externo, admitiu-se a sua colisão com outros objetos descritos através de polígonos. Desenvolveu-se ainda uma linguagem simplificada de animação que permite tanto a parametrização das características das superfícies, como do mundo no qual estão inseridas.

O motivo fundamental que impulsionou este trabalho foi o interesse por parte da autora em obter efeitos de animação ainda não disponíveis nos principais sistemas de animação existentes. Trata-se de um trabalho pioneiro a nível nacional, que objetiva constituir um passo inicial da UFRGS nas pesquisas na área de deformação de objetos através da física. Assim sendo, considerou-se que os resultados obtidos foram bastante bons no que diz respeito ao apelo visual. Como pode ser observado na seção 6.2, as superfícies apresentadas possuem um grau de realismo considerável no que diz respeito à sua aparência, sendo que seu movimento é bastante suave e natural.

Atualmente, o protótipo desenvolvido e apresentado no capítulo 5 vem sendo utilizado pelo Grupo de Computação Gráfica da UFRGS na produção de animações. Pode-se salientar a sua utilização na produção do filme "CG com Filtro", apresentado em [MUS 92a]. No filme, o FLEX3D é utilizado na simulação de um papel que se rasga e de cortinas que se fecham e depois abrem novamente.

## 6.2 Validação do Modelo

A fim de validar o modelo desenvolvido, são apresentadas nesta seção diversas imagens obtidas através do uso do protótipo FLEX3D. No anexo A-2, são apresentados ainda outros exemplos de sequências animadas.

Na figura 6.1 visualiza-se uma superfície, descrita através do roteiro da figura 6.2, em duas condições diferentes. Na figura 6.1a é apresentada a superfície em seu estado inicial, ou seja, antes de ser aplicada sobre ela qualquer tipo de força. Já na figura 6.1b é mostrado o quadro número 25 da animação. Este quadro corresponde ao momento no qual, aplicadas as devidas forças, a superfície encontra-se em repouso, ou seja, a partir deste quadro, o deslocamento dos pontos que definem a superfície é muito pequeno, sendo considerado insignificante.

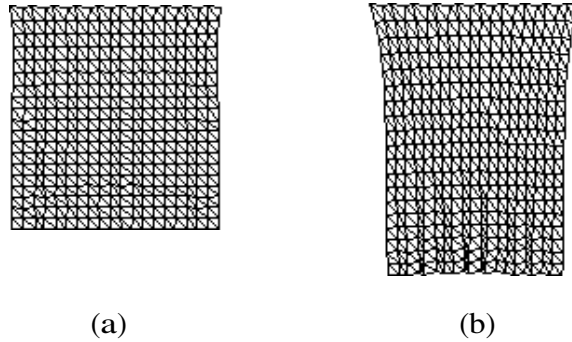


Figura 6.1 - Animação de uma superfície a partir de uma forma inicial, até a sua situação de repouso. (a) Superfície inicial; (b) superfície em repouso (quadro 25).

```

DURAÇÃO = 25
QUADROS_SEG = 1
DELTA_t = 0.001
FORÇA_EXTERNA = (0.0 0.0 0.0)
ATRITO = 0.5
OBJETOS_NA_CENA = 0
NUM_SUPERFÍCIES = 1
SUPERFÍCIE cortina
    MALHA (20 20)
    DENSIDADE = 1.85
    POSIC_NO_SRU (-55.0 40.0 20.0)
    VETOR_NORMAL (0.0 0.0 1.0)
    MASSA = 0.1
    CONST_ELASTICIDADE = (10.0 10.0 10.0)
    CONST_CURVATURA = (0.0 0.0 0.0)
    RESTRIÇÕES
        PONTOS_FIXOS = 20
        0 0
        0 1
        0 2
        0 3
        0 4
        0 5
        0 6
        0 7
        0 8
        0 9
        0 10
        0 11
        0 12
        0 13

```

```
0 14
0 15
0 16
0 17
0 18
0 19
TRANSLAÇÃO (0.0 0.0 0.0)
INI_TRANSL = 0
FIM_TRANSL = 0
FIM_RESTRIÇÕES
FIM_SUPERFÍCIE
```

Figura 6.2 - Roteiro de animação usado na geração da animação apresentada na figura 6.1.

Na figura 6.3 são apresentadas algumas cenas, resultantes de uma animação, realizada com o objetivo de simular uma superfície que se projeta de uma outra superfície planar e fica pendurada por alguns pontos. Esta animação foi realizada para compôr uma das sequências do filme "CG com Filtro" [MUS 92a] onde simula uma folha de papel que se rasga de um *outdoor*, ficando pendurada apenas por uma de suas extremidades. Para simular este efeito assumiu-se a força gravitacional nula e um vetor de força externa parametrizado em  $y$  e  $z$ , como pode-se verificar no roteiro da animação apresentado na figura 6.4.

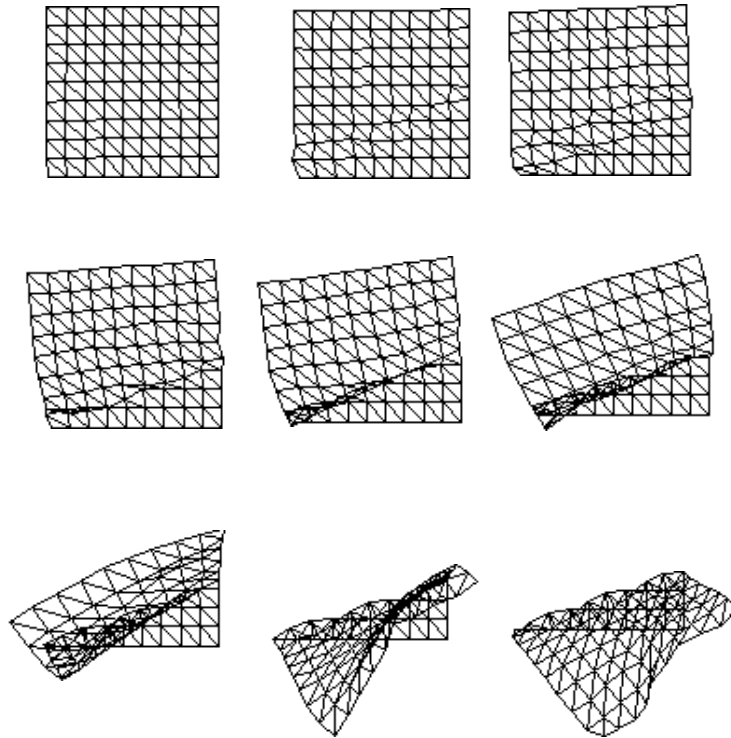


Figura 6.3 - Animação de uma superfície que se projeta de um plano. São apresentados os quadros da animação de número 3, 5, 7, 9, 11, 13, 15, 17 e 19 da esquerda para a direita e de cima para baixo.

```

DURAÇÃO = 20
QUADROS_SEG = 1
DELTA_t = 0.001
FORÇA_EXTERNA = (0.0 -0.39 0.2)
ATRITO = 0.18
OBJETOS_NA_CENA = 0
NUM_SUPERFÍCIES = 1
SUPERFÍCIE papel
    MALHA (10 10)
    DENSIDADE = 2.0
    POSIC_NO_SRU (0.0 40.0 0.0)
    VETOR_NORMAL (0.0 0.0 1.0)
    MASSA = 0.5
    CONST_ELASTICIDADE = (10.0 10.0 10.0)
    CONST_CURVATURA = (0.0 0.0 0.0)

```

```
RESTRICÇÕES
  PONTOS_FIXOS = 22
    90
    91
    92
    93
    94
    95
    96
    97
    98
    99
    83
    84
    85
    86
    87
    88
    89
    76
    77
    78
    79
    69
  TRANSLAÇÃO (0.0 0.0 0.0)
  INL_TRANSL = 0
  FIM_TRANSL = 0
  FIM_RESTRICÇÕES
FIM_SUPERFÍCIE
```

Figura 6.4 - Parâmetros da superfície apresentada na figura 6.3.

## 6.3 Perspectivas Futuras

Após o desenvolvimento deste modelo inicial, verificaram-se diversas opções para dar continuidade a este trabalho, sendo algumas delas expostas aqui.

Como primeira opção, apresenta-se a possibilidade de estender o modelo de representação de superfícies para sólidos deformáveis. Neste caso propõe-se que o objeto deve ser simulado através de uma grade tridimensional onde os vértices que a formam são unidos através de molas.



Modelos semelhantes já foram anteriormente desenvolvidos sendo que pode-se salientar o apresentado por Deussen e Koons no filme "Breaking the Utah Teapot" [DEU nn] e apresentado na figura 6.5.

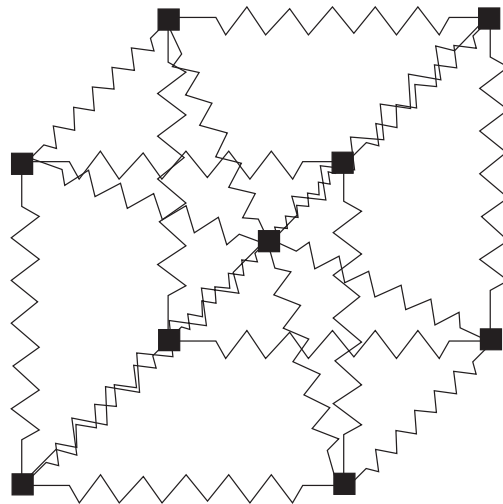


Figura 6.5 - Modelo de representação de um objeto 3D.

Existe, no entanto, muito trabalho a ser feito no que diz respeito às próprias superfícies deformáveis. Uma perspectiva de trabalho é quanto a definição da forma inicial das superfícies, atualmente considerada como uma malha retangular. Poderiam ser estudados mecanismos para modelar uma superfície deformável antes de começar a animação. Para isso, imaginou-se a criação de um modelador que gerasse superfícies de formatos diferentes, os quais respeitassem as seguintes restrições: a superfície deveria poder ser representada em uma matriz  $M_{i,j}$ ; as molas que compõem o sistema deveriam estar todas em repouso, no estado inicial.

Em relação a este último aspecto - restrições, cabe salientar que o presente trabalho se insere no contexto de desenvolvimento de sistemas baseados em restrições e, sob este ponto de vista, uma alternativa viável no prosseguimento da pesquisa é a estruturação do sistema de modelagem e animação segundo a abordagem orientada a objetos.

Quanto à apresentação dos resultados, também poderiam ser estudadas novas propostas. Atualmente, dá-se através da triangularização da superfície. Propõe-se o estudo de técnicas que permitam a apresentação de uma malha deformável através da utilização de superfícies paramétricas, como, por exemplo, as superfícies *splines*. Assim, os pontos que representam a malha funcionariam como pontos de controle para definir a superfície paramétrica. Esta aproximação possibilitaria a criação de superfícies com aparência mais suave e, portanto, mais realística.

Outra opção seria o refinamento do método a fim de utilizá-lo na indústria do vestuário, mais especificamente em CAD para roupas. Para isto, seria necessário estudar as características físicas de diversos tipos de materiais, a fim de simular as suas propriedades através da parametrização das constantes das molas e da densidade de massa dos pontos que compõem uma superfície.

Outra área a ser explorada é a de geração do movimento, onde seria útil um estudo comparativo entre a utilização de diferentes métodos numéricos, com o objetivo de escolher o que melhor se adapta à situação.

Por fim, propõe-se o refinamento da linguagem utilizada na produção do roteiro de animação. O objetivo deste trabalho seria fornecer mais possibilidades de movimento, além de possibilitar ao usuário maior controle sobre a animação. Dentro deste espírito, sugere-se também o desenvolvimento de uma ferramenta que possibilite a definição gráfica e interativa do roteiro.

## ANEXO A-1 FORMATO DO ARQUIVO DE DESCRIÇÃO GEOMÉTRICA DE OBJETOS

O formato de arquivo de descrição de objetos apresentado abaixo foi desenvolvido pelo Grupo de Computação Gráfica da UFRGS com o objetivo de padronizar, tanto a saída como a entrada de dados nos diversos programas existentes. Dessa forma, tornou-se possível a comunicação entre os sistemas desenvolvidos de modo eficiente e prático. No que diz respeito a forma dos objetos, pode-se dizer que são representados através de faces planas.

A seguir será apresentada, sem muito rigorismo, a sintaxe destes arquivos, cuja extensão equivale a .OBT. Em seguida, serão feitos alguns comentários a respeito da semântica deste formato de arquivos.

### A1.1 Sintaxe

```
$OBJETO <nome> <x> <y> <z> <raio>
$VÉRTICES <num_vert>
[<x> <y> <z>]*
$FACES <num_faces>
[<nv> <v1> <v2> ... <vn> <ind_tab_desc>]*
$DESCRIÇÃO
[#<n> <R> <G> <B> <coef_dif> <espec> <amb>]*
$END_OBJ
```

### A1.2 Semântica

Na primeira linha do arquivo, deve ser colocada a palavra reservada \$OBJETO, seguida das características globais de um objeto. É indicado o nome deste objeto, as coordenadas ( $x$ ,  $y$ ,  $z$ ) do seu centro e o raio do envelope esférico que o envolve.

Em seguida, passa-se a descrição dos vértices que definem o objeto. É colocada a palavra \$VÉRTICES seguida do número total de vértices do objeto. Seguindo, são listadas as coordenadas  $(x,y,z)$  desses vértices. Feito isso são apresentadas as faces do objeto. Esta operação é indicada através da palavra \$FACES e do número total de faces existentes. A indicação das faces que compõem o sistema dá-se através dos seguintes parâmetros: número de vértices que a definem; índices que indicam a posição do vértice na tabela de vértices e um índice para uma tabela de descrição. Nesta tabela são descritos os dados referentes às cores das faces.

A definição da tabela de descrição dá-se de forma análoga às citadas acima. Após a palavra reservada \$DESCRIÇÃO, começa-se a listar as condições desejadas. Atrêla-se a cada cor um número precedido do caracter #, suas componentes R, G e B, o coeficiente de luz difusa, especular e ambiental. Por fim, para finalizar a descrição de um objeto, utiliza-se a palavra reservada \$END\_OBJ.

## **ANEXO A-2 EXEMPLOS DE SEQUÊNCIAS ANIMADAS E CENAS GERADAS PELO FLEX3D**

Neste anexo, serão apresentados alguns exemplos de utilização do protótipo FLEX3D na geração de modelos deformáveis. No primeiro exemplo, anteriormente referenciado no capítulo 4, apresenta-se a simulação de uma superfície quadrada que possui 3 de suas 4 extremidades, fixas. Esta superfície está inserida em um universo onde inexistente a força gravitacional. No entanto, ela sofre a força realizada pelo vento e representada através do vetor de força externa. Na figura A2.1 pode-se ver algumas cenas que compõem esta animação, nomeadamente os quadros de número 3, 5, 7, 9, 11, 13, 15, 17 e 19. Na figura A2.2, apresenta-se o roteiro utilizado na parametrização das características geométricas e físicas da malha.

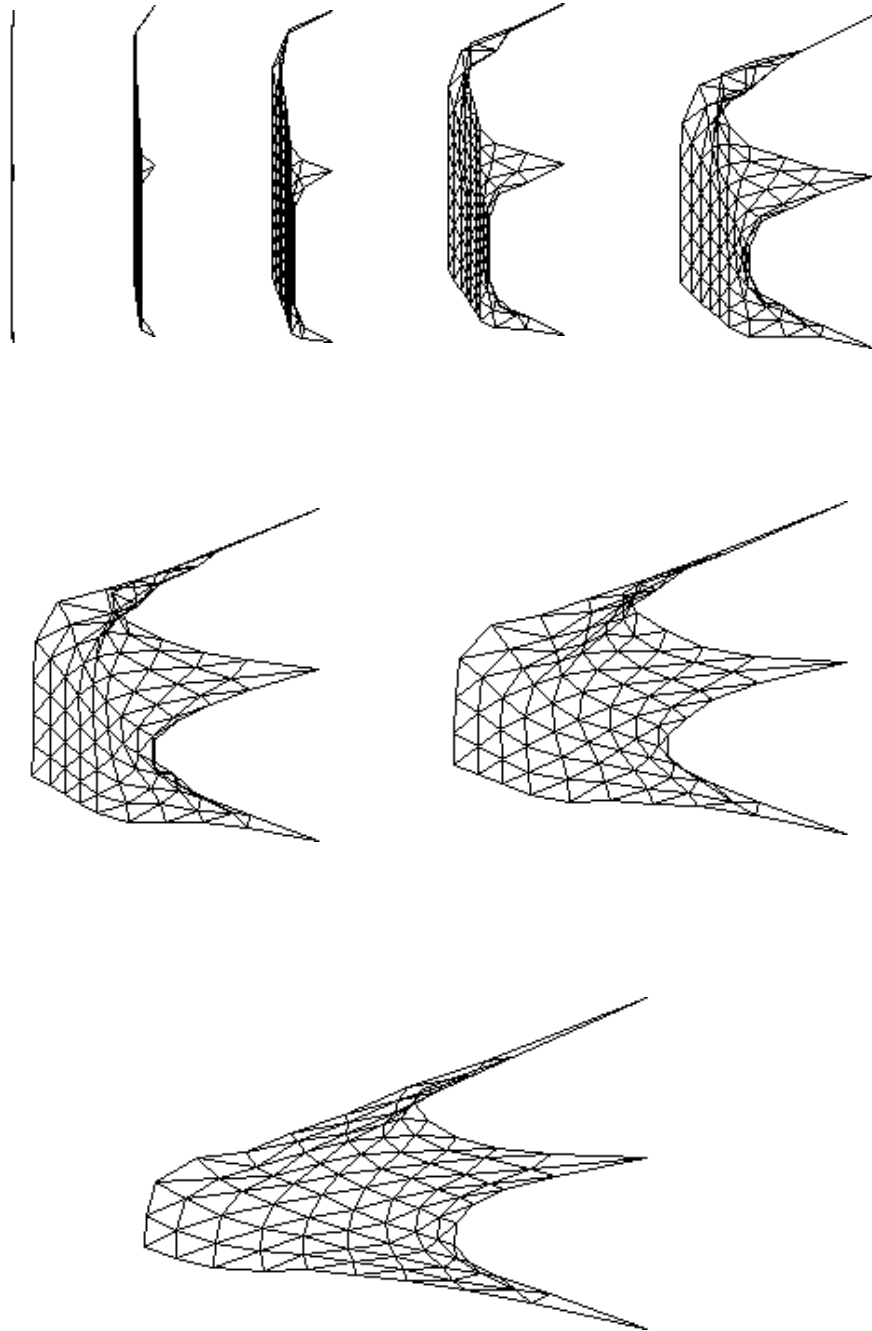


Figura A2.1 - Animação de uma superfície ao vento.

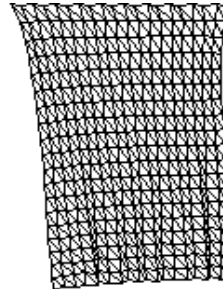
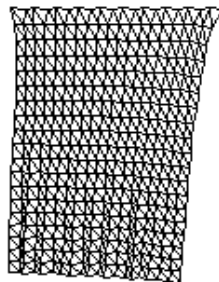
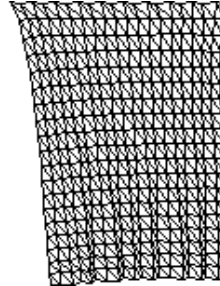
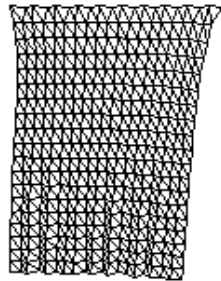
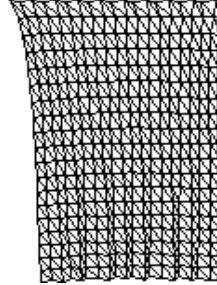
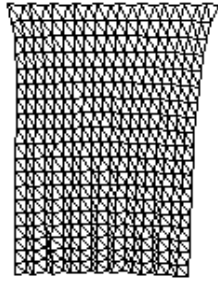
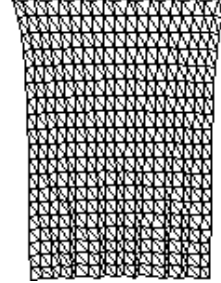
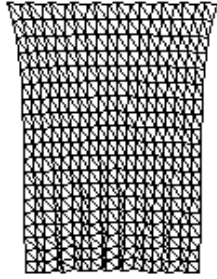
```

DURAÇÃO = 4
QUADROS_SEG = 5
DELTA_t = 0.001
FORÇA_EXTERNA = (0.0 0.0 9.8)
ATRITO = 0.18
OBJETOS_NA_CENA = 0
NUM_SUPERFÍCIES = 1
SUPERFÍCIE
    MALHA (10 10)
    DENSIDADE = 2.0
    POSIC_NO_SRU (0.0 40.0 0.0)
    VETOR_NORMAL (0.0 0.0 1.0)
    MASSA = 0.5
    CONST_ELASTICIDADE = (10.0 10.0 10.0)
    CONST_CURVATURA = (0.0 0.0 0.0)
    RESTRIÇÕES
        PONTOS_FIXOS = 3
        0 0
        0 9
        9 0
        TRANSLAÇÃO (0.0 0.0 0.0)
        INI_TRANSL = 0
        FIM_TRANSL = 0
    FIM_RESTRIÇÕES
FIM_SUPERFÍCIE

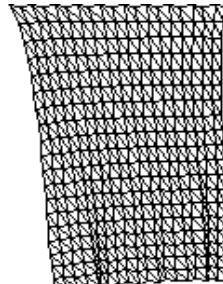
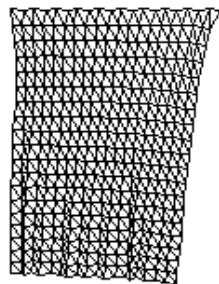
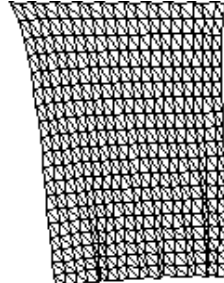
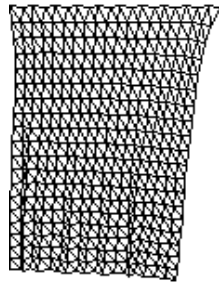
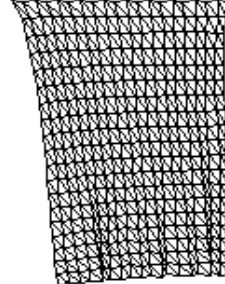
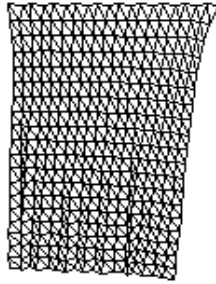
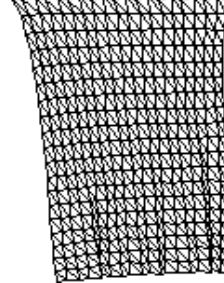
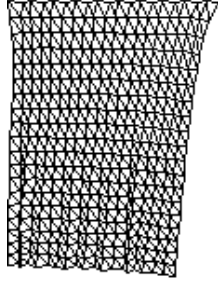
```

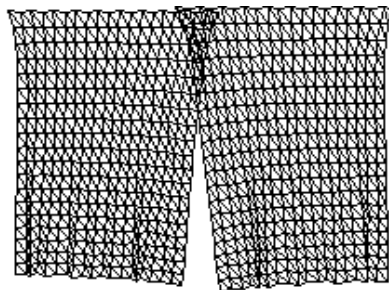
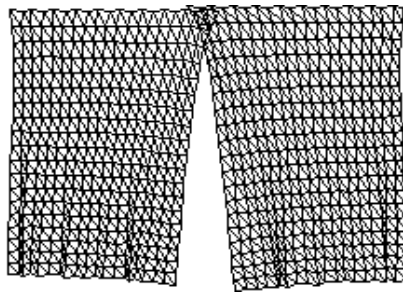
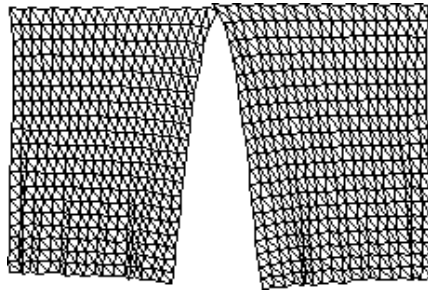
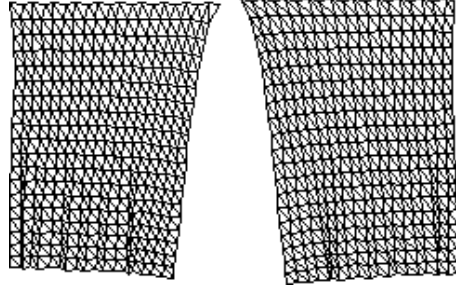
Figura A2.2 - Roteiro para parametrização da superfície exposta ao vento.

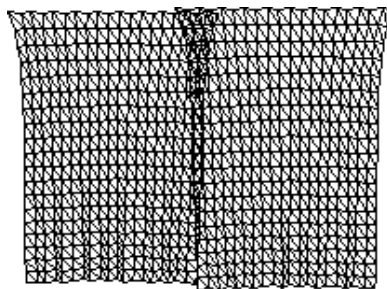
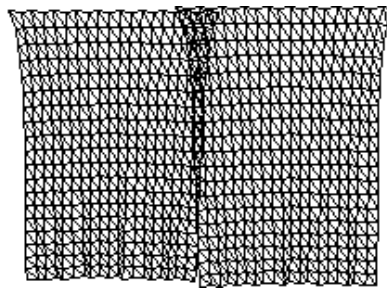
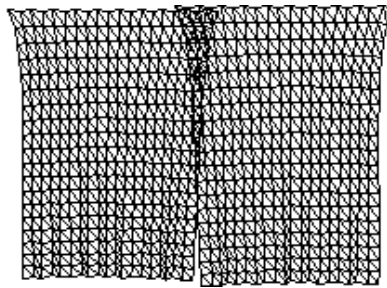
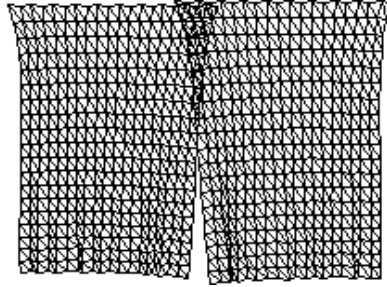
A segunda animação apresentada, tal qual a figura 6.3, exhibe uma sequência desenvolvida para o filme "CG com Filtro". Esta sequência tem como objetivo simular o fechamento e a re-abertura das cortinas de um teatro. Na figura A2.3 será apresentada uma sequência de 20 imagens geradas pelo FLEX3D e escolhidas de um total de 145, com o objetivo de ilustrar a simulação efetuada. Para permitir a movimentação de uma superfície em dois sentidos opostos (abre e fecha cortina) foi necessário efetuar uma pequena modificação na linguagem de parametrização do modelo. Para tanto, assumiu-se a possibilidade de existir mais de um movimento de translação no mesmo roteiro, conforme pode ser verificado na figura A2.4 onde são apresentados os parâmetros do modelo.











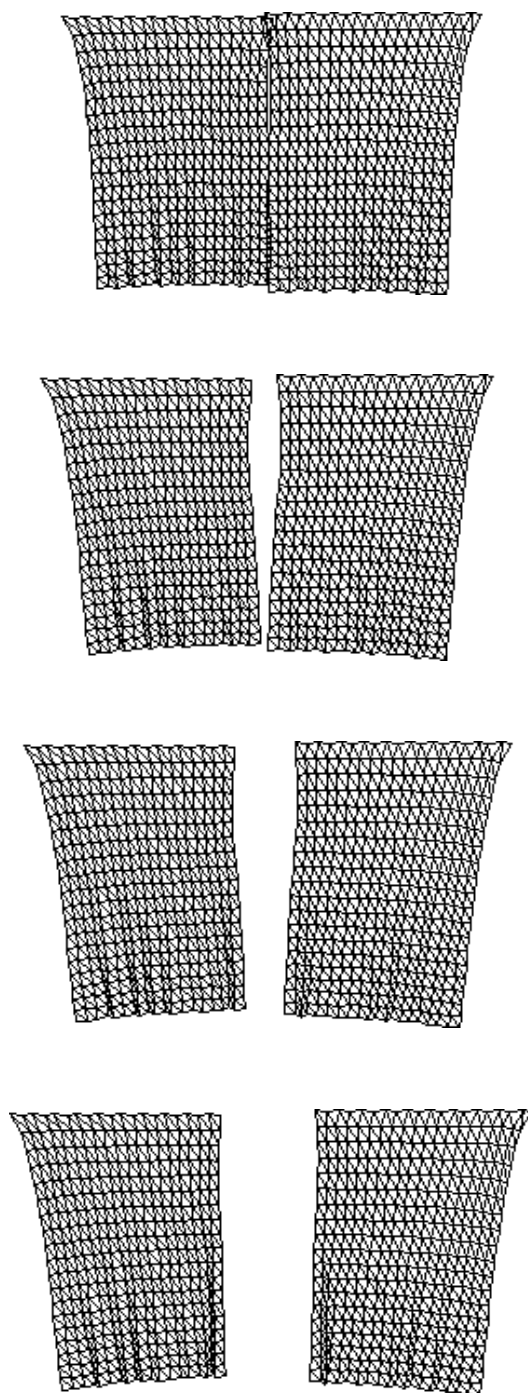


Figura A2.3 - Simulação do movimento das cortinas de um teatro.

DURAÇÃO = 145  
QUADROS\_SEG = 1  
DELTA\_t = 0.001  
FORÇA\_EXTERNA = (0.0 0.0 0.0)  
ATRITO = 0.5  
OBJETOS\_NA\_CENA = 0  
NUM\_SUPERFÍCIES = 2  
SUPERFÍCIE corte  
MALHA (20 20)  
DENSIDADE = 1.85  
POSIC\_NO\_SRU (-55.0 40.0 20.0)  
VETOR\_NORMAL (0.0 0.0 1.0)  
MASSA = 0.1  
CONST\_ELASTICIDADE = (10.0 10.0 10.0)  
CONST\_CURVATURA = (0.0 0.0 0.0)  
RESTRICÇÕES  
PONTOS\_FIXOS = 20  
0 0  
0 1  
0 2  
0 3  
0 4  
0 5  
0 6  
0 7  
0 8  
0 9  
0 10  
0 11  
0 12  
0 13  
0 14  
0 15  
0 16  
0 17  
0 18  
0 19  
TRANSLAÇÃO (0.43 0.0 0.0)  
INI\_TRANSL = 25  
FIM\_TRANSL = 78  
TRANSLAÇÃO (-0.57 0.0 0.0)  
INI\_TRANSL = 100  
FIM\_TRANSL = 140  
FIM\_RESTRICÇÕES  
FIM\_SUPERFÍCIE

```

SUPERFÍCIE cortd
  MALHA (20 20)
  DENSIDADE = 1.85
  POSIC_NO_SRU (18.0 40.0 23.0)
  VETOR_NORMAL (0.0 0.0 1.0)
  MASSA = 0.1
  CONST_ELASTICIDADE = (10.0 10.0 10.0)
  CONST_CURVATURA = (0.0 0.0 0.0)
  RESTRICÇÕES
    PONTOS_FIXOS = 20
    0 0
    0 1
    0 2
    0 3
    0 4
    0 5
    0 6
    0 7
    0 8
    0 9
    0 10
    0 11
    0 12
    0 13
    0 14
    0 15
    0 16
    0 17
    0 18
    0 19
    TRANSLAÇÃO (-0.43 0.0 0.0)
    INI_TRANSL = 25
    FIM_TRANSL = 78
    TRANSLAÇÃO (0.57 0.0 0.0)
    INI_TRANSL = 100
    FIM_TRANSL = 140
  FIM_RESTRICÇÕES
FIM_SUPERFÍCIE

```

Figura A2.4 - Parâmetros de configuração das cortinas.

Na figura A2.5 são exibidas algumas cenas de uma sequência de animação semelhante a apresentada na figura 6.3. Trata-se da simulação de uma superfície que se projeta de outra superfície planar e fica pendurada por alguns pontos. Neste caso, assumiu-se uma superfície com baixa capacidade de curvatura e torção objetivando simular um material com maior rigidez, ou seja, pouco flexível. Assumiu-se, ainda, a força da gravidade nula e um vetor

de força externa parametrizado em  $y$  e  $z$ , como pode-se verificar o roteiro apresentado na figura A2.6..

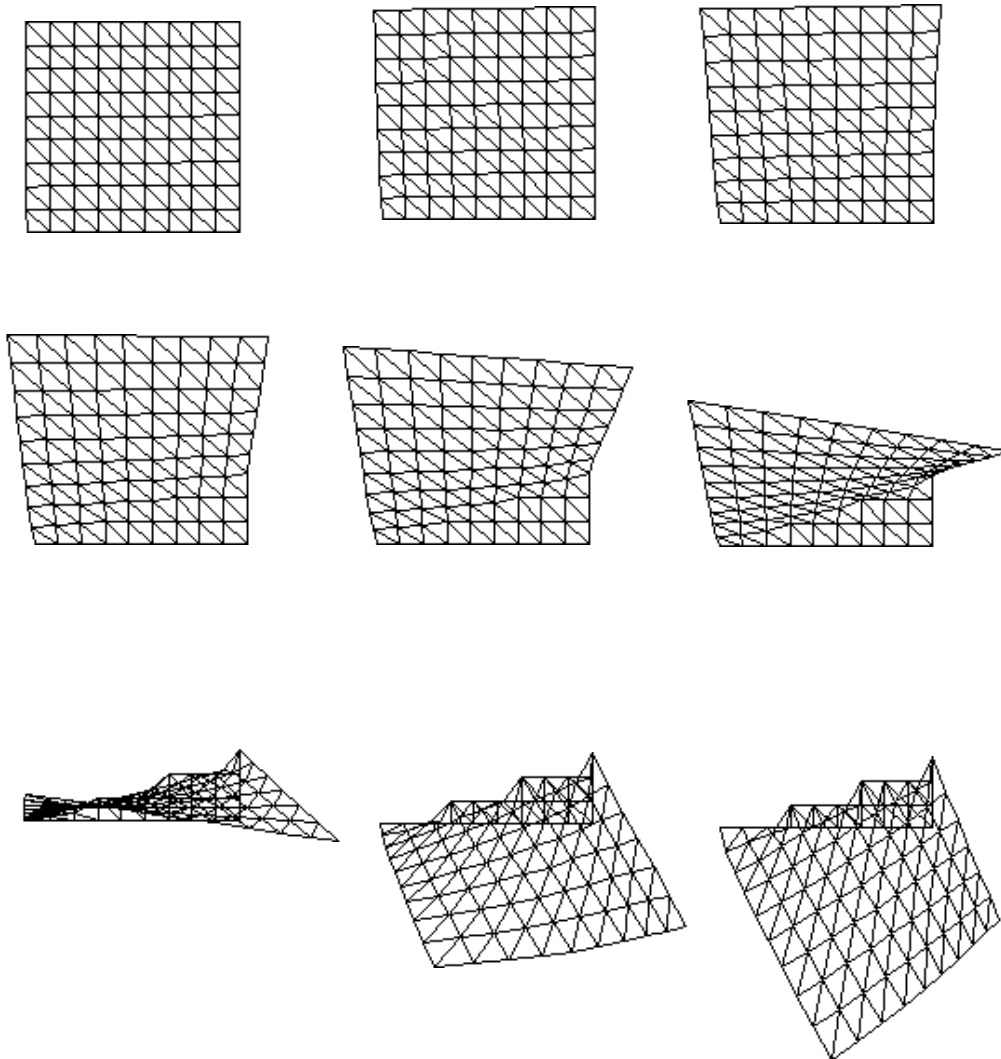


Figura A2.5 - Animação de uma superfície com baixa capacidade de curvatura e torção e que se projeta de um plano.

DURAÇÃO = 20  
QUADROS\_SEG = 1  
DELTA\_t = 0.001  
FORÇA\_EXTERNA = (0.0 -0.39 0.2)

```

ATRITO = 0.18
OBJETOS_NA_CENA = 0
NUM_SUPERFÍCIES = 1
SUPERFÍCIE papelao
    MALHA (10 10)
    DENSIDADE = 2.0
    POSIC_NO_SRU (0.0 40.0 0.0)
    VETOR_NORMAL (0.0 0.0 1.0)
    MASSA = 0.5
    CONST_ELASTICIDADE = (10.0 10.0 10.0)
    CONST_CURVATURA = (10.0 10.0 10.0)
    RESTRIÇÕES
        PONTOS_FIXOS = 22
            9 0
            9 1
            9 2
            9 3
            9 4
            9 5
            9 6
            9 7
            9 8
            9 9
            8 3
            8 4
            8 5
            8 6
            8 7
            8 8
            8 9
            7 6
            7 7
            7 8
            7 9
            6 9
        TRANSLAÇÃO (0.0 0.0 0.0)
        INI_TRANSL = 0
        FIM_TRANSL = 0
    FIM_RESTRIÇÕES
FIM_SUPERFÍCIE

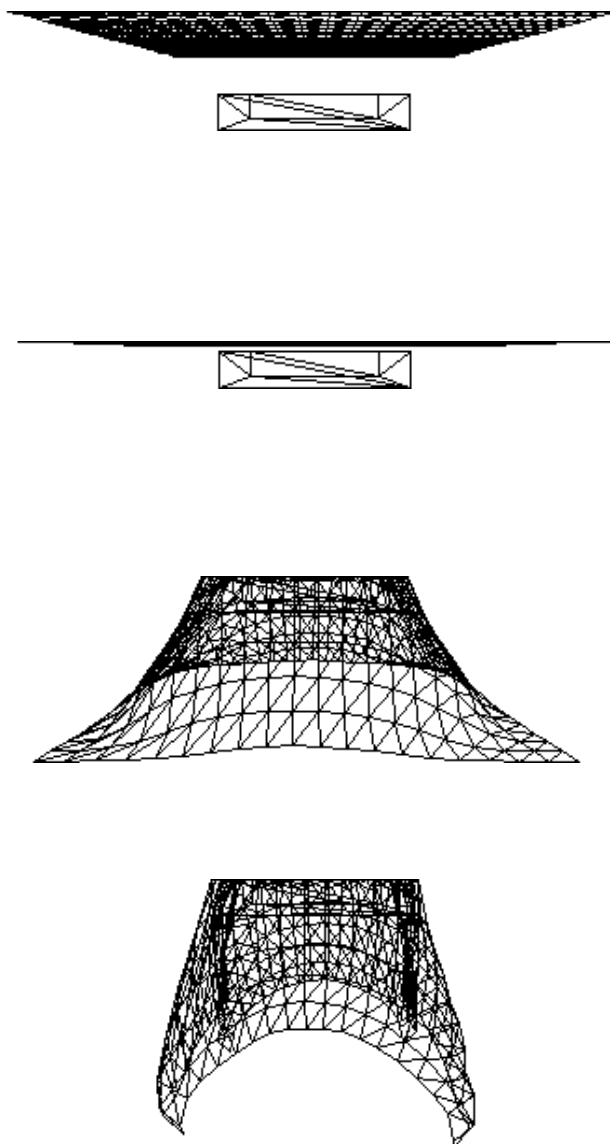
```

Figura A2.6 - Parâmetros de configuração da superfície apresentada na figura A2.5.

Por fim, será apresentado um exemplo através do qual procurou-se apresentar a simulação da queda uma toalha sobre uma mesa (figura A2.7). Para tanto, utilizou-se uma malha com uma quantidade considerável de partículas (400) e um objeto externo para simular a mesa, no caso um



paralelepípedo, conforme figura A2.8. A simulação ocorre considerando-se que a superfície flexível está em queda livre ao colidir com o objeto sólido.



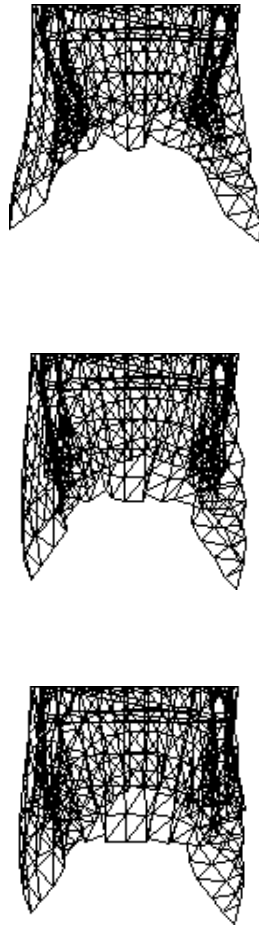


Figura A2.7 - Simulação da queda de uma toalha sobre uma mesa.

```
DURAÇÃO = 10
QUADROS_SEG = 2
DELTA_t = 0.001
FORÇA_EXTERNA = (0.0 -9.8 0.0)
ATRITO = 0.6
OBJETOS_NA_CENA = 1
mesa.obt
NUM_SUPERFÍCIES = 1
SUPERFÍCIE toalha
    MALHA (20 20)
    DENSIDADE = 2.0
    POSIC_NO_SRU (0.0 15.0 -1.0)
    VETOR_NORMAL (0.0 1.0 0.0)
    MASSA = 2.0
    CONST_ELASTICIDADE = (50.0 50.0 50.0)
```

```
CONST_CURVATURA = (0.0 0.0 0.0)
RESTRICÇÕES
  PONTOS_FIXOS = 0
  TRANSLAÇÃO (0.0 0.0 0.0)
  INI_TRANSL = 0
  FIM_TRANSL = 0
FIM_RESTRICÇÕES
FIM_SUPERFÍCIE
```

Figura A2.8 - Parâmetros de configuração da animação apresentada na figura A2.8.

## BIBLIOGRAFIA

- [AMO 92] AMORIM, M.G.; PÍCCOLO, H.L. KIKO: sistema de animação para representação de movimento humano. In: SIBGRAPI, 5, 1992, Águas de Lindóia, S.P. *Anais...* [s.l.:s.n.], 1992. 330p. p. 243-250.
- [ARM 85] ARMSTRONG, W.W.; GREEN, M.W. The Dynamics of articulated rigid bodies for purposes of animation. In: GRAPHICS INTERFACE, May 1985, Toronto. *Proceedings...* [s.l.]:Canadian Information Processing Soc., 1985. p. 407-415.
- [ARM 86] ARMSTRONG, W.W.; GREEN, M.; LAKE, R. Near-Real-Time control of human figure models. In: GRAPHICS INTERFACE; VISION INTERFACE, 1986, Canada. *Proceedings...* [s.l.:s.n.], 1986. p. 147-151.
- [ARN 89] ARNALDI, B.; DUMONT, G.; HÉGRON, G.; MAGNENAT-THALMANN, N.; THALMANN, D. Animation control with dynamics. State-of-the-art in computer animation. In: COMPUTER ANIMATION, 1989, Geneva, Switzerland. *Proceedings...* Tokyo:Springer-Verlag, 1989. p. 113-123.
- [BAD 90] BADOUEL, D. An Efficient Ray-Polygon Intersection. In: GLASSNER, Andrew S., ed. *Graphics gems*. San Diego: Academic Press, 1990. p. 390-393.
- [BAF 90] BARAFF, D. Curved surfaces and coherence for non-penetrating rigid body simulation. *ACM Computer Graphics*, New York, N.Y., v. 24, n. 4, p. 19-23, Aug. 1990.

- [BAF 92] BARAFF, D.; Witkin, A. Dynamic simulation of non-penetrating flexible bodies. *ACM Computer Graphics*, New York, N.Y., v. 26, n. 2, p. 303-308, July 1992.
- [BAR 84] BARR, A.H. Global and local deformations of solid primitives. *ACM Computer Graphics*, New York, N.Y., v. 18, n. 3, p. 21-30, July 1984.
- [BAR 88] BARR, A.H. Introduction to physically-based modelling. In: *SIGGRAPH: Developments in Physically-Based Modelling*, Aug. 1988, Atlanta, Georgia. *Proceedings...* [s.l.:s.n.], 1988. p. A1-A5. (Course Notes, 27)
- [BAR 88a] BARR, A.H. Teleological modelling. In: *SIGGRAPH: Developments in Physically-Based Modelling*, Aug. 1988, Atlanta, Georgia. *Proceedings...* [s.l.:s.n.], 1988. p. B1-B6. (Course Notes, 27)
- [BAZ 88] BARZEL, R.; BARR, A.H. A modelling system on dynamic constraints. *ACM Computer Graphics*, New York, N.Y., v. 22, n. 4, p. 179-188, Aug. 1988.
- [BEC 92] BECHMANN, D.; DUBREUIL, N.. Animation through space and time based on a space deformation model. In: *EUROGRAPHICS, Workshop on Animation and Simulation*, 3, Sept. 1992, Cambridge, England. *Proceedings...* [s.l.:s.n.], 1992.
- [CAI 92] CARIGNAN, M.; YANG, Y.; MAGNENAT-THALMANN, N.; THALMANN, D.. Dressing animated synthetic actors with complex deformable clothes. *ACM Computer Graphics*, New York, N.Y., v. 26, n. 2, p. 99-104, July 1992.

- [CAL 91] CALVERT, T. Composition of realistic animation sequences for multiple human figures. In: MAKING THEM MOVE: mechanics, control, and animation of articulated figures, San Mateo, CA:Morgan Kaufmann, 1991. p. 35-50.
- [CAR 92] CARVALHO, P.C.P.; GOMES, J.M.; MARTHA, L.F.; VELHO, L.. Modelagem em computação gráfica. In: SIBGRAPI, 1992, Águas de Lindóia, S.P. Notas de Tutorial. (Não editado) .
- [CEL 91] CELNIKER, G.; GOSSARD, D.. Deformable curve and surface finite-elements for free-form shape design. **ACM Computer Graphics**, New York, N.Y., v. 25, n. 4, p. 257-266, July 1991.
- [CHA 89] CHADWICK, J.E.; HAUMANN, D.R.; PARENT, R.E. Layered construction for deformable animated characters. **ACM Computer Graphics**, New York, N.Y., v. 23, n. 3, p. 243-252, July 1989.
- [CLA 89] CLÁUDIO, D.M.; MARINS, J.M. **Cálculo numérico computacional: teoria e prática**. São Paulo:Atlas , 1989. 464 p.
- [COL 89] COLONNA, J.F. Animation of fractal objects. In: GRAPHICS INTERFACE, 1989, [s.l.]. **Proceedings...** [s.l.:s.n.], 1989. p. 242-248.
- [COQ 91] COQUILLARD, S.; JANCÈNE, P. Animated free-form deformation: an interactive animation technique. **ACM Computer Graphics**, New York, N.Y., v. 25, n. 4, p. 23-26, July 1991.
- [DEU nn] DEUSEN, M.V.; KOONS, R. Breaking the utah teapot, In: **THE FILM Computer Visions**.

- [DIA 91] DIAS, J.M.S.; REBORDÃO, J.M. Um Modelo energético para simulação de superfícies deformáveis. In: ENCONTRO PORTUGUÊS DE COMPUTAÇÃO GRÁFICA, 4, nov. 1991, Sesimbra, Portugal. Actas... [s.l.:s.n.], 1991. p. 109-124.
- [FOL 90] FOLEY, J.; VAN DAM, A.; FEINER, S.K.; HUGHES, J.F. **Computer graphics: principles and practice**. USA:Addison - Wesley , 1990. 1174p.
- [GAS 92] GASCUEL, J.D.; GASCUEL, M.P. Displacement constraints: a new method for interactive dynamic animation of articulated solids. In: EUROGRAPHICS, Workshop on Animation and Simulation, 3., Sept. 1992, Cambridge, England. **Proceedings...** [s.l.:s.n.], 1992.
- [GEA 71] GEAR, C.W. **Numerical initial value problems in ordinary differential equations**. Englewood Cliffs, N.J.:Prentice-Hall, 1971. 253 p.
- [GEA 84] GEAR, C.W.; OSTERBY, O. Solving ordinary differential equations with discontinuities. **ACM Transactions on Mathematical Software**, New York, N.Y., v. 10, n. 1, p. 23-44, Mar. 1984.
- [GIR 91] GIRARD, M. Constrained optimization of articulated animal movement in computer animation. In: **MAKING THEM MOVE: mechanics, control, and animation of articulated figures**. San Mateo, CA:Morgan Kaufmann, 1991. p. 209-232.
- [GOS 90] Goss, M.E. A Real time particle system for display of ship wakes. **IEEE Computer Graphics and Applications**, Los Alamitos, CA, v. 10, n. 3, p. 30-35, 1990.

- [GRE 88] GREEN, M.; SUN, H. MML: A Language for procedural modelling and motion. In: GRAPHICS INTERFACE, 1988, [s.l.]. **Proceedings...** [s.l.:s.n.], 1988. p. 16-25.
- [GRE 91] GREEN, M. Using dynamics in computer animation: control and solution issues. In: MAKING THEM MOVE: mechanics, control, and animation of articulated figures. San Mateo, CA:Morgan Kaufmann, 1991. p. 281-314.
- [HEG 89] HÉGRON, G.; PALAMIDESE, P.; THALMANN, D. Motion control in animation, simulation and visualization. **Computer Graphics Forum**, Amsterdam, v. 8, p. 347-352, 1989.
- [HOU 92] HOUSE, D.H.; BREEN, D.C.; GETTO, P.H. On the dynamic simulation of physically-based particle-system models. In: EUROGRAPHICS, Workshop on Animation and Simulation, 3, Sept. 1992, Cambridge, England. **Proceedings...** [s.l.:s.n.], 1992.
- [HSU 92] HSU, W.M.; HUGHES, J.F.; KAUFMAN, H. Direct manipulation of free-form deformations. **ACM Computer Graphics**, New York, N.Y., v. 26, n. 2, p. 177-184, July 1992.
- [HUN 77] HUNTER, G.M. Computer animation survey. **Computers and Graphics**, Oxford, v. 2, p. 225-229, 1977.
- [ISA 87] ISAACS, P.M.; COHEN, M.F. Controlling dynamic simulation with kinematic constraints, behavior functions and inverse dynamics. **ACM Computer Graphics**, New York, N.Y., v. 21, n. 4, p. 215-224, July 1987.



- [ISA 88] ISAACS, P.M.; COHEN, M.F. Mixed methods for complex kinematic constraints in dynamic figure animation. **The Visual Computer**, [s.l.], v. 4, p. 296-305, 1988.
- [KOC 87] KOCHANEK, D.; HIGGINS, T.; FORSEY, D. **Introduction to computer animation**. Canada:National Film Board of Canada, Computer Animation Centre, 1987. 116p.
- [KRÖ 89] KRÖMKER, D.; HOFMANN, G.R. Computer Animation. In: **EUROGRAPHICS: European Computer Graphics Conference and Exhibition**, 10, Sept. 4-8, 1989, Hamburg, FRG. **Tutorial Notes**. Amsterdam: North-Holland, 1989.
- [LOP 92] LOPES, P.F.; GOMES, M.R. A Computer model for pinscreen simulation: a new animation paradigm. **Computer Graphics Forum**, Amsterdam, v. 11, n. 2, p. 31-42, 1992.
- [LOS 92] LOSINA SILVA, R. **ANIMAKER: sistema de animação**. Porto Alegre:UFRGS, Instituto de Informática, 1992. 96p. (Projeto de Diplomação).
- [LUN 87] LUNDIN, D. Ruminations of a model maker. **IEEE Computer Graphics and Applications**, Los Alamitos, CA, v. 7, n. 5, p. 3-5, May 1987.
- [MAG 85] MAGNENAT-THALMANN, N.; THALMANN, D. **Computer animation: theory and practice**. Tokyo:Springer-Verlag, 1985. 240p.

- [MAG 85a] MAGNENAT-THALMANN, N.; THALMANN, D. Subactor data types as hierarchical procedural models for computer animation. In: EUROGRAPHICS, 1985, Copenhagen. **Proceedings...** [s.l.]:North-Holland, 1985. p. 121-128.
- [MAG 88] MAGNENAT-THALMANN, N.; LAPERRIÈRE, R.; THALMANN, D. Joint-dependent local deformations for hand animation and object grasping. In: GRAPHICS INTERFACE, 1988, [s.l.]. **Proceedings...** [s.l.:s.n.], 1988. p. 26-33.
- [MAG 91] MAGNENAT-THALMANN, N.; THALMANN, D. Human body deformations using joint-dependent local operators and finite-element theory. In: MAKING THEM MOVE: mechanics, control, and animation of articulated figures. San Mateo, CA:Morgan Kaufmann, 1991. p. 243-262.
- [MET 92] METAXAS, D.; TERZOPOULOS, D. Dynamic deformation of solid primitives with constraints. **ACM Computer Graphics**, New York, N.Y., v. 26, n. 2, p. 309-312, July 1992.
- [MIL 91] MILLER, J.V.; BREEN, D.E.; LORENSEN, W.E.; O'BARA, R.M.; WOZNY, M.J. Geometrically deformed models: a method for extracting closed geometric models from volume data. **ACM Computer Graphics**, New York, N.Y., v. 25, n. 4, p. 217-226, July 1991.
- [MLL 87] MILLER, G.S.P. The Motion dynamics of snakes and worms. **ACM Computer Graphics**, New York, N.Y., v. 22, n. 4, p. 169-178, Aug. 1988.

- [MLL 91] MILLER, G.S.P. Goal-directed animation of tubular articulated figures or how snakes play golf. In: MAKING THEM MOVE: mechanics, control, and animation of articulated figures, San Mateo, CA:Morgan Kaufmann, 1991. p. 233-241.
- [MOO 88] MOORE, M.; WILHELMS, J. Collision detection and response for computer animation. *ACM Computer Graphics*, New York, N.Y., v. 22, n. 4, p. 289-298, Aug. 1988.
- [MUS 92] MUSSE, S.R. **Um Estudo sobre animação computadorizada de objetos rígidos articulados**, Porto Alegre:CPGCC da UFRGS, 1992. 50p. (Trabalho Individual, 275)
- [MUS 92a] MUSSE, S.R. et al. A Realização do filme CG com Filtro, In: SEMANA DE COMPUTAÇÃO, 1, 1992, Porto Alegre, RS. **SECOMP**. Porto Alegre:UFRGS, Instituto de Informática, 1992. (Não editado)
- [NED 91] NEDEL, L.P. **PINSCREEN: modelagem e animação em computador**. Porto Alegre:CPGCC da UFRGS, 1991. 134p. (Trabalho Individual, 239)
- [NED 92] NEDEL, L.P. Modelagem e Animação de Superfícies Deformáveis. In: CONFERENCIA LATINOAMERICANA DE INFORMATICA, ago. 1992, Las Palmas de Gran Canaria, España. *Actas...* [s.l.:s.n.], 1992. p. 989-996.
- [OPP 86] OPPENHEIMER, P.E. Real time design and animation of fractal plants and trees. *ACM Computer Graphics*, New York, N.Y., v. 20, n. 4, p. 55-64, Aug. 1986.

- [PEN 89] PENTLAND, A.; WILLIAMS, J. Good vibrations: modal dynamics for graphics and animation. *ACM Computer Graphics*, New York, N.Y., v. 23, n. 3, p. 215-222, July 1989.
- [PLA 88] PLATT, J.; TERZOPOULOS, D.; FLEISCHER, K.; BARR, A. Elastically deformable models. In: *SIGGRAPH: Developments in Physically-Based Modelling*, Aug. 1988, Atlanta, Georgia. *Proceedings...* [s.l.:s.n.], 1988. p. K1-K25. (Course Notes, 27)
- [PLA 88a] PLATT, J.C.; BARR, A.H. Constraint methods for flexible models. *ACM Computer Graphics*, New York, N.Y., v. 22, n. 4, p. 279-288, Aug. 1988.
- [PLA 88b] PLATT, J. Another introduction to numerical analysis. In: *SIGGRAPH: Developments in Physically-Based Modelling*, Aug. 1988, Atlanta, Georgia. *Proceedings...* [s.l.:s.n.], 1988. p. C1-C9. (Course Notes, 27)
- [PRE 88] PRESS, W.H.; FLANNERY, B.P.; TEUKOLSKY, S.A.; VETTERLING, W.T. *Numerical recipes in C*. Cambridge, England:Cambridge University Press, 1988. 735p.
- [PUE 88] PUEYO, X.; TOST, D. A Survey of computer animation. *Computer Graphics Forum*, Amsterdam, v. 7, p. 281-300, 1988.
- [REE 83] REEVES, W.T. Particle Systems: a technique for modelling a class of fuzzy objects. *ACM Computer Graphics*, New York, N.Y., v. 17, n. 3, p. 359-376, July 1983.

- [RES 84] RESNICK, R.; HALLIDAY, D. **Física 1**. Rio de Janeiro: Livros Técnicos e Científicos, 1984. 348p.
- [RIG 92] RIGO, S. **Animação por computador: um estudo de técnicas**. Porto Alegre: CPGCC da UFRGS, 1992. 100p. (Trabalho Individual, 275).
- [SCH 92] SCHMIDT, A.E.F. **Um Estudo sobre sistemas baseados em restrições**. Porto Alegre: CPGCC da UFRGS, 1992. 49p. (Trabalho Individual, 269).
- [SCL 91] SCLAROFF, S.; PENTLAND, A. Generalized implicit functions for computer graphics. **ACM Computer Graphics**, New York, N.Y., v. 25, n. 4, p. 247-250, July 1991.
- [SED 86] SEDERBERG, T.W.; PARRY, S.R. Free-form deformation of solid geometric models. **ACM Computer Graphics**, New York, N.Y., v. 20, n. 4, p. 151-160, Ago. 1986.
- [SMI 84] SMITH, A.R. Plants, fractals, and formal languages. **ACM Computer Graphics**, New York, N.Y., v. 18, n. 3, p. 1-10, July 1984.
- [SUT 63] SUTHERLAND, I.E. **SKETCHPAD: a man-machine graphical communication system**. Massachusetts: Massachusetts Institute of Technology, Lincoln Laboratory, 1963. (Technical Report).
- [SZE 92] SZELISKI, R.; TONNESEN, D. Surface modeling with oriented particle systems. **ACM Computer Graphics**, New York, N.Y., v. 26, n. 2, p. 185-194, July 1992.

- [TER 88] TERZOPOULOS, D.; FLEISCHER, K. Deformable models. **The Visual Computer**, [s.l.], v. 4, p. 306-331, 1988.
- [TER 88a] TERZOPOULOS, D.; WITKIN, A. Physically based models with rigid and deformable components. **IEEE Computer Graphics and Applications**, Los Alamitos, CA, v. 8, n. 6, p. 41-51, Nov. 1988.
- [TER 88b] TERZOPOULOS, D.; FLEISCHER, K. Modeling inelastic deformation: viscoelasticity, plasticity, fracture. **ACM Computer Graphics**, New York, N.Y., v. 22, n. 4, p. 269-278, Aug. 1988.
- [WEI 86] WEIL, J. The Synthesis of cloth objects. **ACM Computer Graphics**, New York, N.Y., v. 20, n. 4, p. 49-54, Ago. 1986.
- [WEL 69] WELLS, D.A. **Lagrangian dynamics**. New York:McGraw-Hill, 1969. (Shaum's Outline Series).
- [WIL 87] WILHELMS, J. Using dynamic analysis for realistic animation of articulated bodies. **IEEE Computer Graphics and Applications**, Los Alamitos, CA, v. 7, n. 6, p. 12-27, June 1987.
- [WIL 87a] WILHELMS, J. Toward automatic motion control. **IEEE Computer Graphics and Applications**, Los Alamitos, CA, v. 7, n. 4, p. 11-22, Apr. 1987.
- [WIL 91] WILHELMS, J. Dynamic experiences. In: **MAKING THEM MOVE: mechanics, control, and animation of articulated figures**. San Mateo, CA:Morgan Kaufmann, 1991. p. 265-280.

- [WIT 87] WITKIN, A.; FLEISCHER, K.; BARR, A. Energy constraints on parameterized models. **ACM Computer Graphics**, New York, N.Y., v. 21, n. 4, p. 225-232, July 1987.
- [WIT 88] WITKIN, A.; KASS, M. Spacetime constraints. **ACM Computer Graphics**, New York, N.Y., v. 22, n. 4, p. 159-168, Aug. 1988.
- [WIT 90] WITKIN, A.; WELCH, W. Fast animation and control of nonrigid structures. **ACM Computer Graphics**, New York, N.Y., v. 24, n. 4, p. 243-252, Aug. 1990.