

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO
DEPARTAMENTO DE ENGENHARIA DE COMPUTAÇÃO E AUTOMAÇÃO INDUSTRIAL

Leandro de Pinho Monteiro

Soluções Numéricas em um Modelo de Tecidos Baseado na Superfície de Cosserat

Orientadora: Profa. Dra. Wu, Shin-Ting

Dissertação de Mestrado apresentada à Faculdade de Engenharia Elétrica e de Computação como parte dos requisitos para obtenção do título de Mestre em Engenharia Elétrica. Área de concentração: **Engenharia de Computação**.

Campinas, SP
Dezembro de 2007

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO
DEPARTAMENTO DE ENGENHARIA DE COMPUTAÇÃO E AUTOMAÇÃO INDUSTRIAL

Leandro de Pinho Monteiro

Soluções Numéricas em um Modelo de Tecidos Baseado na Superfície de Cosserat

Orientadora: Profa. Dra. Wu, Shin-Ting

Dissertação de Mestrado apresentada à Faculdade de Engenharia Elétrica e de Computação como parte dos requisitos para obtenção do título de Mestre em Engenharia Elétrica. Área de concentração: **Engenharia de Computação**.

Banca Examinadora

Profa. Dra. Rosângela Barros Zanoni Lopes Moreno .. DEP/FEM/UNICAMP
Prof. Dr. Léo Pini Magalhães DCA/FEEC/UNICAMP
Prof. Dr. Clésio Luis Tozzi DCA/FEEC/UNICAMP

Campinas, SP
Dezembro de 2007

Resumo

Atualmente, os modelos físicos são amplamente utilizados em simulações computacionais de vestuários e tecidos em geral. Eles podem ser divididos em duas abordagens: mecânica de partículas e mecânica de contínuos. A mecânica de partículas é hoje a abordagem mais utilizada, possuindo vantagens como a simplicidade da formulação e um bom desempenho computacional. Já a mecânica de contínuos é reconhecida por ser a abordagem mais precisa sob o ponto de vista físico, porém tem um alto custo computacional. Muitas vezes, o baixo desempenho da mecânica de contínuos está relacionado à solução numérica utilizada para obter diferentes estados do tecido ao longo do tempo, em decorrência da sua formulação envolver grandezas geométricas diferenciáveis. As soluções encontradas na literatura são técnicas de elementos finitos ou de diferenças finitas, com integração semi-implícita ou implícita, todas com uma complexidade computacional não linear. Desta forma, este trabalho visa analisar a adequabilidade de utilizar soluções explícitas, que tem um comportamento linear, em um modelo físico fundamentado na teoria de superfície de Cosserat, cujos autores provam que ele consegue produzir dobras sob forças de compressão. Discretizando espacialmente este modelo com a técnica das diferenças finitas, métodos explícitos são avaliados para solucionar as equações diferenciais ordinárias resultantes. Com base nestas avaliações, foi selecionado o método de Verlet e implementado um simulador da dinâmica de tecidos retangulares provido de uma interface gráfica interativa. Isso viabilizou uma validação prática aprofundada do modelo, demonstrando a sua superioridade na produção de dobras em relação aos outros modelos existentes.

Palavras-chave: Computação Gráfica, Simulações Físicas, Simulação de Tecidos, Modelagem e Animação de Tecidos, Mecânica de Contínuos, Superfície de Cosserat.

Abstract

Currently, the physic models are very used for cloth simulations. They can be divided in two branches: particles mechanics and continuum mechanics. The particle mechanics is the most used today, having some advantages like an easy formulation and a high performance. The continuum mechanics is recognized to be a more precise branch in physic aspects, but it has a high computational cost. A lot of times, the low performance of continuum mechanics is related to the numerical solution that obtains cloth states along the time, because its formulation have differential geometric quantities. The solutions founded in the literature are the finite element and the finite differences techniques, with the semi-implicit and implicit integration, all with a non linear computational complexity. So, this work aims to analyze the suitability of apply explicit solutions that have a linear behavior, with a physic model founded on the theory of Cosserat surface, which the authors prove that it can produces naturally folds under pure compression forces. A spatial discretization is done with finite difference schema for evaluate explicit methods to solve the resulted ordinary differentials equations. Based on this evaluation, the Verlet method was selected and a rectangular mesh dynamics simulator implemented with an interactive graphics interface. This permits a practice validation of the cloth model, showing its superiority on production of folds compared with the others models.

Keywords: Computer Graphics, Physics Simulation, Cloth Simulation, Cloth Modeling and Animation, Continuum Mechanics, Cosserat Surface.

À minha mãe.

Agradecimentos

À minha mãe, Leda, pelo amor, suporte, carinho e paciência em todos os momentos.

À minha orientadora, Ting, pela ajuda e apoio intelectual e pela amizade conquistada nestes anos de pesquisa.

À professora Rosângela Barros Zanoni Lopes Moreno e aos professores Léo Pini Magalhães e Clésio Luis Tozzi, por aceitarem participar da banca examinadora desta dissertação. Em especial, agradeço ao professor Clésio pela atenção dedicada na fase dos créditos.

À minha namorada, Mariana, pelo amor, companheirismo e paciência durante a longa etapa dedicada a este mestrado.

Aos meus amigos do LCA, em especial ao Adler, Harlen, Jeremias e Alex.

Às pessoas que de algum modo contribuíram, por existirem.

À CAPES, pelo suporte financeiro.

Sumário

Lista de Figuras	iv
Lista de Tabelas	vii
Lista de Símbolos	viii
1 Introdução	1
1.1 Simulação de Tecidos	2
1.1.1 Modelo de Tecido	2
1.1.2 Integração Temporal	5
1.1.3 Problemas	7
1.2 Conteúdo do Trabalho	8
1.2.1 Motivação	9
1.2.2 Hipótese	9
1.2.3 Objetivos	9
1.2.4 Escopo	10
1.2.5 Contribuições	10
1.3 Organização da Dissertação	11
2 Simulações de Tecidos	13
2.1 Modelos Físicos baseados na Mecânica de Partículas	14
2.1.1 Integrações Explícitas	14
2.1.2 Integrações Implícitas	15
2.1.3 Integrações Semi-implícitas	16
2.2 Modelos Físicos baseados na Mecânica de Contínuos	17
2.2.1 Integrações Implícitas	17
2.2.2 Integrações Semi-implícitas	18
2.3 Classificação dos Modelos	21
2.4 Uma modelagem baseada em uma Superfície de Cosserat	22
2.4.1 Superfície de Cosserat	22
2.4.2 Restrições	28
2.5 Considerações Finais	31

3	Soluções Numéricas	33
3.1	Discretizações Espaciais	34
3.2	Métodos Explícitos de Integração	41
3.2.1	Método de Euler	43
3.2.2	Método do Ponto Intermediário	45
3.2.3	Método de Runge-Kutta	48
3.2.4	Método <i>Leapfrog</i>	51
3.2.5	Método de Verlet	52
3.3	Trabalhos Relacionados	54
3.4	Resultados	59
3.4.1	Simulações de Teste	59
3.4.2	Método de Euler	62
3.4.3	Método do Ponto Intermediário	64
3.4.4	Método de Runge-Kutta	65
3.4.5	Método <i>Leapfrog</i>	66
3.4.6	Método de Verlet	67
3.5	Análises	68
3.5.1	Estabilidade Numérica	68
3.5.2	Eficiência Computacional	71
3.5.3	Qualidade das Imagens	73
3.5.4	Escolha do Método Numérico	74
3.6	Considerações Finais	75
4	Implementação	77
4.1	Modelagem de dados	78
4.1.1	Fluxo de Dados	78
4.1.2	Algoritmos	79
4.1.3	Análise de Requisitos	81
4.1.4	Classe <i>malha</i>	83
4.1.5	Classe <i>render</i>	84
4.2	Interface Gráfica	86
4.2.1	Plataforma Linux	86
4.2.2	Plataforma Windows	88
4.2.3	Exemplos de Simulação	90
4.3	Considerações Finais	93
5	Resultados	95
5.1	Metodologia	96
5.1.1	Calibração de Parâmetros de Material	96
5.1.2	Calibração de Forças Externas	99
5.2	Simulações	101
5.2.1	<i>Bucklings</i>	102
5.2.2	<i>Drapings</i>	104
5.2.3	<i>Hangings</i>	107

5.3	Análises	108
5.3.1	Dinâmica	109
5.3.2	Dobras	113
5.4	Considerações Finais	116
6	Conclusões e Trabalhos Futuros	117
	Referências bibliográficas	121

Lista de Figuras

2.1	Modelos massa-mola: (a) Haumann [1] em 1987; e (b) Provot [2] em 1995.	15
2.2	Resultados do trabalho de Wu e de Melo [3] em 2004: (a) dobras; e (b) caimento. . .	20
2.3	Referenciais fixo e arbitrário em instantes diferentes.	23
3.1	Malha discreta.	34
3.2	Problemas nas bordas.	36
3.3	Derivadas aproximadas com diferenças finitas: (a) para pontos internos; (b) para pontos nas bordas B1 e B2; e (c) para pontos nas bordas B3 e B4.	40
3.4	Gráficos do (a) método explícito de Euler e (b) método semi-implícito de Euler usado por Provot [2] em 1995.	45
3.5	Gráfico comparativo entre os métodos de Euler e ponto intermediário.	47
3.6	Gráfico das aproximações feitas no método de Runge-Kutta de quarta ordem.	48
3.7	Saltos de posições e velocidades no método <i>Leapfrog</i>	51
3.8	Superfície inicial das simulações: (a) S_1, S_2, S_3, S_5 e S_6 ; e (b) S_4	60
3.9	Fotos dos estados de equilíbrio.	61
3.10	Resultados visuais do método de Euler.	63
3.11	Resultados visuais do método do ponto intermediário.	64
3.12	Resultados visuais do método de Runge-Kutta.	65
3.13	Resultados visuais do método <i>Leapfrog</i>	66
3.14	Resultados visuais do método de Verlet.	67
3.15	Instabilidade numérica causada por altos valores de Δt	69
3.16	Superfície da simulação S_1 com $\zeta_{\alpha\beta} = 5$	71
3.17	Tempos totais de processamento para executar 10000 iterações.	72
4.1	Fluxograma da dinâmica de tecidos neste trabalho.	78
4.2	Classe <i>malha</i> e suas sub-classes.	83
4.3	Modelagem da classe <i>objeto</i>	84
4.4	Modelagem da classe <i>render</i>	85
4.5	Exemplo de um arquivo de entrada com todos os dados de uma simulação.	86
4.6	Interface gráfica para a plataforma Linux.	87
4.7	Janela das restrições da simulação.	87
4.8	Janela de visualização da animação gerada.	88
4.9	Janelas de (a) simulação; e (b) visualização da animação gerada.	89
4.10	Janela de restrições.	90

4.11	Imagens de simulação.	90
4.12	Definição e ilustração das forças pontuais.	91
4.13	Efeitos do coeficiente ρ_D : (a) com ele e (b) sem ele.	93
5.1	Influência dos coeficientes elásticos.	97
5.2	Influência de ϕ	98
5.3	Atuação de forças externas pontuais.	100
5.4	<i>Bucklings</i> de diferentes materiais de tecido.	102
5.5	Formação de uma gravata borboleta (a) flexível e (b) rígida.	103
5.6	Pontos fixos na malha discreta para formar uma mesa (a) quadrada e (b) redonda.	104
5.7	Caimentos de tecidos sobre mesas redondas.	105
5.8	Caimento de tecido sobre uma mesa (a) redonda e (b) quadrada.	106
5.9	Caimento de um tecido sobre uma poltrona.	106
5.10	Cortinas suspensas de diferentes materiais: (a) seda; (b) algodão; e (c) plastificada.	107
5.11	Bandeira no estado (a) inicial e (b) deformado.	108
5.12	Dinâmica de <i>bucklings</i>	109
5.13	Dinâmica de <i>drappings</i>	111
5.14	Dinâmica de <i>hangings</i>	112
5.15	<i>Bucklings</i> em grades 60×60	114

Lista de Tabelas

2.1	Classificação dos trabalhos apresentados nas seções 2.1 e 2.2.	21
3.1	Passos de tempo máximos e iterações necessárias do método de Euler.	62
3.2	Tempos de processamento do método de Euler.	63
3.3	Passos de tempo máximos e iterações necessárias do método do ponto intermediário.	64
3.4	Tempos de processamento do método do ponto intermediário.	64
3.5	Passos de tempo máximos e iterações necessárias do método de Runge-Kutta.	65
3.6	Tempos de processamento do método de Runge-Kutta.	65
3.7	Passos de tempo máximos e iterações necessárias do método <i>Leapfrog</i>	66
3.8	Tempos de processamento do método <i>Leapfrog</i>	66
3.9	Passos de tempo máximos e iterações necessárias do método de Verlet.	67
3.10	Tempos de processamento do método de Verlet.	67
3.11	Passos de tempo máximos de S_1 com $\zeta_{\alpha\beta} = 5$	70
5.1	Dados das simulações.	113

Lista de Símbolos

S	- Superfície não deformada correspondente ao instante t_o
$s(t)$	- Superfície deformada no instante t
\mathbf{R}	- Vetor posição de um ponto da superfície S
\mathbf{r}	- Vetor posição de um ponto da superfície s
x^α	- Curvas coordenadas
\mathbf{a}_α	- Vetores de base
\mathbf{a}^α	- Vetores de base recíprocos
\mathbf{a}_3 e \mathbf{n}	- Vetor normal
$A_{\alpha\beta}$	- Tensor de métrica de S
$a_{\alpha\beta}$	- Tensor de métrica de s
$\varepsilon_{\alpha\beta}$	- Deformação da métrica de s em relação à S
$a = \det(a_{\alpha\beta})$	- Determinante da matriz composta pelos elementos do tensor métrico
$b_{\alpha\beta}$	- Tensor de curvatura
b_β^α	- Símbolos de Weingarten
$\kappa_{\alpha\beta}$	- Deformação da curvatura de s em relação à S
$\Gamma_{\alpha\beta}^\lambda$	- Símbolos de Christoffel
\mathbf{v}	- Vetor de velocidades
\mathbf{a}	- Vetor de acelerações
\mathbf{f}_e	- Vetor de forças externas
\mathbf{f}_i	- Vetor de forças internas
ϱ_D	- Coeficiente de amortecimento do meio
ϱ_F	- Coeficiente de densidade de fluido
$\frac{\delta \varepsilon(\mathbf{r}, t)}{\delta \mathbf{r}}$	- Variação da energia potencial em \mathbf{r} no instante t
$\mathbf{N}^\alpha _\alpha$	- Força elástica

Capítulo 1

Introdução

A simulação de tecidos é do interesse de diversas áreas de trabalho, como na propaganda e entretenimento, usando animações computadorizadas, e na indústria têxtil, auxiliando os profissionais da área de moda no projeto de vestuários e malhas em geral. Apesar de várias estratégias para modelagem e animação de tecidos por computador terem sido rapidamente desenvolvidas nas duas últimas décadas, o comportamento realista de sua dinâmica continua sendo um problema desafiador tanto para comunidade de computação gráfica quanto para a de engenharia têxtil. Enquanto os pesquisadores de computação gráfica estão mais interessados em um modelo estável e robusto numericamente para animar vestuários sobre personagens virtuais, os cientistas têxteis priorizam um modelo mecânico que relacione o comportamento de tecidos aos parâmetros de seu material.

Um dos grandes problemas relacionados à animação de tecidos é formular um modelo que agregue realismo e eficiência para atingir um comportamento natural de dobras sob forças de compressão. Fundamentado em superfícies elásticas de Cosserat, Wu e de Melo [3] propuseram um modelo de superfície deformável que consegue representar várias características peculiares de tecidos, incluindo a formação natural de dobras sob compressão, através de um número relativamente pequeno de parâmetros. Algumas simplificações foram necessárias para tornar o modelo computacionalmente factível. Embora tais simplificações não devam influenciar a dinâmica da simulação sob o ponto de vista teórico, as implementações disponíveis do modelo não permitem uma validação mais extensa. Esta dissertação apresenta uma alternativa de implementação, tornando os códigos mais modulares, como também viabilizando a validação prática das principais características do modelo.

Este capítulo está organizado da seguinte maneira: a seção 1.1 apresenta uma breve introdução sobre etapas e problemas envolvidos na simulação de tecidos; a seção 1.2 expõe o conteúdo deste trabalho, discutindo sua motivação, hipótese, foco, escopo e principais contribuições; por fim, a seção 1.3 descreve resumidamente como está organizado o restante desta dissertação.

1.1 Simulação de Tecidos

Os tecidos, em geral, possuem um comportamento bem característico, que é de se curvar com facilidade e se esticar com dificuldade. Esta característica faz com que eles sofram muitas deformações, as quais são governadas por leis físicas complexas do mundo real que devem ser levadas em consideração, em uma simulação computacional, para se obter resultados realistas. O paradigma de modelagem de tecidos é o que diferencia duas abordagens físicas: a mecânica de contínuos e a mecânica de partículas, as quais são resumidamente explicadas na subseção 1.1.1. Para simular a dinâmica de tecidos é preciso ainda obter as variações de sua geometria ao longo do tempo, o que pode ser feito basicamente de três formas: com uma integração implícita, semi-implícita ou explícita, conforme brevemente apresentado na subseção 1.1.2. A combinação destas duas etapas, a de modelagem do comportamento de um tecido sob forças e sua aplicação ao longo do tempo, resulta em uma sequência de formas que o tecido pode assumir. Há, porém, alguns problemas relacionados à eficiência e à estabilidade numérica neste processo, que são discutidos na subseção 1.1.3.

1.1.1 Modelo de Tecido

Tecidos são estruturas composta de fios, entrelaçados em um determinado padrão. Os fios são feitos de fibras, as quais determinam o material do tecido¹. A composição, o padrão de entrelaçamento e o tratamento dado aos fios de um tecido determinam seu comportamento quando sujeito à ação de forças externas. Devido à diversidade de parâmetros que combinados resultam em um comportamento bastante peculiar, a modelagem de um tecido pode ser feita de diferentes formas. Pode-se classificar estas formas basicamente em dois tipos: os modelos físicos, como por exemplo nos trabalhos [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 2, 16, 17, 1, 18, 19]; e os modelos geométricos, por exemplo [20, 21].

Os modelos geométricos não levam em consideração as propriedades físicas dos tecidos, utilizando entidades e transformações geométricas para descrever estados estacionários dos tecidos. Apesar da simplicidade em se usar modelos geométricos, eles não são adequados para reproduzir o movimento de tecidos, pois se limitam em descrever suas formas usando um certo padrão pré-definido [21], desprezando os fenômenos naturais que ocorrem na realidade. Já os modelos físicos visam representar com maior fidelidade a complexidade envolvida no comportamento de tecidos, fazendo uso de um mecanismo que satisfaça leis físicas que vigoram na vida real e, portanto, atingem um realismo maior quando comparados aos modelos geométricos.

Os modelos físicos são reconhecidos por serem os mais adequados para produzir uma aparência natural da dinâmica de tecidos. Atualmente, os principais *softwares* gráficos comerciais, como o 3D Studio Max e Maya, utilizam modelos físicos dinâmicos para simular o movimento de tecidos.

¹Estas fibras podem ser naturais como a lã e o algodão ou sintéticas como o elastano e o couro.

Terzopoulos et al. [18] propuseram modelar um tecido como uma superfície deformável cuja dinâmica é governada pela equação diferencial de equilíbrio

$$\mu \frac{\partial^2 \mathbf{r}}{\partial t^2} + \varrho \frac{\partial \mathbf{r}}{\partial t} + \frac{\delta \varepsilon(\mathbf{r}, t)}{\delta \mathbf{r}} = \mathbf{f}(\mathbf{r}, t), \quad (1.1)$$

na qual as forças externas $\mathbf{f}(\mathbf{r}, t)$ sobre cada ponto \mathbf{r} da superfície precisam ser balanceadas pela somatória de três elementos: força correspondente à energia cinemática, forças dissipativas e forças devido à variação da energia interna armazenada na superfície. É neste último termo, referente às forças internas de fricção que são geradas pela variação de energia potencial acumulada na superfície deformável, que se distinguem dois tipos de modelos físicos: os que utilizam a mecânica de partículas e os que utilizam a mecânica de contínuos.

Os modelos que utilizam a mecânica de partículas consideram uma superfície deformável como um conjunto de pontos discretos, que são as partículas, dispostos em uma grade na qual são conectados uns aos outros por molas lineares ou não-lineares, responsáveis pelo comportamento elástico de alguns tipos de tecido, como os feitos de elastano. Estas molas podem assumir diferentes estruturas de ligação entre as partículas, como, por exemplo, nas direções horizontal, vertical e diagonal, e se deformam dentro e fora do plano da superfície criando tensões, distensões, cisalhamentos e curvamentos que são calculados de acordo com a mecânica clássica, sendo o material do tecido aproximado pela rigidez das molas. As deformações das molas são então usadas para estimar a energia interna acumulada em cada partícula da superfície. Normalmente, a energia resultante de uma partícula é calculada pelas deformações das molas adjacentes a ela ou dentro de uma vizinhança pré-estabelecida.

Já os modelos baseados na mecânica de contínuos consideram o tecido como uma superfície deformável contínua sobre a qual são aplicadas teorias de casca lineares e não-lineares para analisar os comportamentos de tensão, distensão, cisalhamento e curvamento nas deformações de um tecido. Estas deformações podem ser descritas como a variação da energia interna em relação aos elementos da geometria diferencial, como os tensores de métrica e de curvatura. O cálculo da variação da energia interna de um ponto da superfície do tecido é então obtida em função da variação dos coeficientes da primeira e da segunda forma fundamental em relação ao estado não deformado do tecido.

Contudo, os modelos baseados na mecânica de contínuos precisam ser discretizados no domínio espacial para serem computacionalmente implementáveis, de forma que ambas as abordagens de modelagem física de partículas e contínuos podem ter formulações similares, diferindo essencialmente nas equações constitutivas, as quais são responsáveis por descrever as forças internas $\frac{\delta \varepsilon(\mathbf{r}, t)}{\delta \mathbf{r}}$ causadas por deformações no tecido. As equações constitutivas são relações entre duas quantidades físicas, como a lei de Hooke, que relaciona a força elástica de forma diretamente proporcional ao produto do coeficiente de rigidez de uma mola pelo deslocamento sofrido por ela.

Existem três metas relacionadas à simulação de tecidos que são colocadas em primeiro plano dependendo da finalidade da aplicação: o **realismo**, em animações, como filmes, desenhos e propagandas; a **eficiência**, relevante nos jogos em geral, pois de nada adianta um jogo realista se sua interatividade ficar prejudicada por algum método custoso que, na prática, serve apenas para lhe dar um maior adereço; e a **precisão física**, importante para análise e simulação do comportamento de tecidos nas aplicações da Engenharia Têxtil. Neste sentido, não existe uma forma correta de simular tecidos, pois a escolha de qual modelo de tecido usar depende das necessidades do projetista.

O realismo na dinâmica de tecidos está relacionado com o uso de modelos físicos. A mecânica de contínuos consegue atingir uma precisão física superior à mecânica de partículas, pois se baseia, em geral, em um modelo geometricamente exato considerando o tecido da forma mais real possível, ao invés de imaginá-lo com uma grade conectada por molas. Entretanto, esta maior precisão física resulta em uma formulação mais complexa que torna o modelo de tecido mais difícil de ser implementado e também menos eficiente computacionalmente em relação à mecânica de partículas. Atualmente, os modelos baseados na mecânica de contínuos são reconhecidos por melhor representarem as deformações dos tecidos, porém na prática os modelos baseados na mecânica de partículas são os mais utilizados pela simplicidade na formulação das forças internas e pela eficiência.

Em 2000, Breen e House editaram um livro que compara em um de seus capítulos duas implementações com abordagens físicas distintas [22]. Nesta comparação foi utilizado o modelo de partículas de Breen et al. [16] e o modelo baseado na mecânica de contínuos de Eischen et al. [15]. A conclusão foi de que quando uma alta precisão é requerida, os modelos baseados na mecânica de contínuos são as melhores opções, enquanto em termos de velocidade computacional são recomendados os modelos baseados na mecânica de partículas. Desta forma, esta dissertação considera avaliar a utilização de um modelo baseado na mecânica de contínuos com uma implementação alternativa para simular tecidos que agregue realismo e precisão física junto com um comportamento linear de complexidade computacional. Para tanto, é necessário discretizar espacialmente o modelo baseado na mecânica de contínuos para que esta implementação fique clara.

A discretização espacial em modelos baseados na mecânica de partículas tem o objetivo de reduzir as soluções de equações diferenciais parciais, presentes na formulação do modelo, para equações diferenciais ordinárias em função do tempo. Estas equações são então solucionadas com um método numérico para obter diferentes estados do tecido ao longo do tempo, produzindo sua dinâmica. Este processo numérico de encontrar diferentes formas sequenciais do tecido ao longo do tempo é muitas vezes referenciado de integração temporal. Assim, antes de integrar o modelo de tecido baseado na mecânica de contínuos é preciso discretizá-lo no espaço, o que é pode ser feito utilizando os métodos de diferenças finitas ou elementos finitos. O método de diferenças finitas possui uma formulação simples e com bom desempenho computacional, porém necessita do tratamento de problemas de

contorno, que ocorrem quando não existem pontos adjacentes em determinadas direções nas fronteiras do tecido. Já o método dos elementos finitos não possui problemas de contorno, porém apresenta uma formulação mais complexa e demanda maior esforço computacional.

O método das diferenças finitas foi o escolhido para discretizar espacialmente o modelo elástico proposto por Terzopoulos et al., o qual era baseado na mecânica de contínuos e visava simular o comportamento de modelos elásticos deformáveis, incluindo os tecidos [18]. Inspirados nesta discretização, Wu e de Melo aplicaram um esquema similar para discretizar um modelo fundamentado na teoria de superfície de Cosserat, também baseado na mecânica de contínuos e formulado para simular a dinâmica de tecidos [3]. Já os elementos finitos foi a opção de Eischen et al., que utilizaram um modelo de casca [17] para o mesmo propósito [15]. Além deles, Etmuss et al. usaram os elementos finitos mostrando que é possível obter eficiência computacional com esta técnica na simulação de tecidos [8]. Em um outro trabalho, Etmuss et al. demonstraram que é possível derivar uma formulação baseada na mecânica de partículas a partir de uma baseada na mecânica de contínuos utilizando um esquema de diferenças finitas [23]. Com isso, pode-se obter uma formulação similar ao modelo de partículas, mesmo utilizando um modelo mais complexo para representar o tecido, para que se possam aproveitar algumas de suas vantagens de eficiência.

Portanto, visando explorar a alternativa de usar um modelo baseado na mecânica de contínuos para produzir a simulação de tecidos de forma eficiente, é interessante utilizar um modelo discretizado com as diferenças finitas para obter uma formulação similar àquela usada na mecânica de partículas. Neste sentido, foi selecionado o modelo proposto por Wu e de Melo, o qual fundamentado na teoria de superfície de Cosserat consegue reproduzir o comportamento natural de dobras que se formam nos tecidos sob apenas forças de compressão [3]. Este comportamento ocorre porque, diferentemente dos outros modelos, os autores incluíram um termo que relaciona as variações de métrica com as de curvatura, preservando a compatibilidade entre estas duas grandezas geométricas e assegurando a regularidade da superfície. Além disso, este modelo tem um controle intuitivo dos parâmetros do material do tecido.

1.1.2 Integração Temporal

Apesar das diferentes abordagens de modelagem física de tecidos, elas procuram solucionar o mesmo problema, que é encontrar de forma precisa e estável as novas formas do tecido, ao longo do tempo, quando sujeito a forças. Este processo, chamado de integração temporal, recai na solução da equação 1.1, que a partir de um instante t é usada para obter os pontos da superfície em $t + \Delta t$, sendo Δt chamado de passo de tempo. Existem muitos esquemas de solucionar esta equação, diferindo em características como estabilidade numérica, eficiência e precisão. Basicamente, estes esquemas podem ser divididos em três tipos de integração temporal: implícita, semi-implícita e explícita.

De forma similar à modelagem de tecidos, a escolha do método de integração temporal também depende da finalidade da aplicação. Para uma aplicação que priorize o realismo e a precisão física, os métodos implícitos combinados com modelos contínuos são os mais adequados. Já para uma aplicação em tempo real, como, por exemplo, nos jogos modernos, que precisam de um melhor desempenho, uma integração explícita junto com um modelo de partículas é o mais indicado. Naturalmente, à medida que o passo de tempo Δt se aproxima de 0, as soluções aproximadas dos novos pontos da superfície do tecido em cada instante de tempo $t + \Delta t$ tendem a ficar mais precisas, para qualquer método de integração temporal, devido a proximidade entre os valores dos elementos do modelo de tecido entre passos subseqüentes. Contudo, conforme o passo de tempo cresce, é possível notar a diferença de precisão e estabilidade numérica que cada tipo de integração consegue atingir.

A integração temporal implícita é a que obtém a solução aproximada com maior precisão numérica em relação à solução exata, pois nela os novos pontos da superfície do tecido são encontrados no instante $t + \Delta t$ com base nas forças do mesmo instante $t + \Delta t$. Além disso, este tipo de integração tem a vantagem de conseguir encontrar as soluções de forma estável com um passo de tempo maior do que os utilizados por esquemas semi-implícitos ou explícitos, embora tenha a desvantagem de precisar solucionar um sistema de equações lineares, implicando na linearização do modelo de tecido e também em um alto custo computacional. A equação 1.1 pode ser reescrita para uma integração temporal implícita como

$$\mu \frac{\partial^2 \mathbf{r}}{\partial t^2}(t + \Delta t) + \varrho \frac{\partial \mathbf{r}}{\partial t}(t + \Delta t) + \frac{\delta \varepsilon(\mathbf{r}, t + \Delta t)}{\delta \mathbf{r}} = \mathbf{f}(\mathbf{r}, t + \Delta t). \quad (1.2)$$

Já na integração temporal explícita é feito o contrário, de forma que os novos pontos da superfície são encontrados em $t + \Delta t$ com base nas forças exercidas no instante t , gerando um erro de aproximação proporcional ao tamanho do passo de tempo Δt , de forma que quanto menor for este valor, menor será o erro cometido. Por utilizar as forças do instante anterior t para encontrar as novas geometrias do tecido em $t + \Delta t$, esta é a integração temporal menos precisa e também a que apresenta mais problemas de estabilidade numérica, principalmente para valores grandes de passo de tempo, pois depende muito da suavidade com que as forças variam entre dois instantes seguidos. Entretanto, este é o tipo de integração mais fácil de ser implementado e o que apresenta a melhor eficiência computacional. A equação 1.1 pode ser reescrita de forma a explicitar os novos termos em $t + \Delta t$ em função dos anteriores em t

$$\mu \frac{\partial^2 \mathbf{r}}{\partial t^2}(t + \Delta t) = \mathbf{f}(\mathbf{r}, t) - \varrho \frac{\partial \mathbf{r}}{\partial t}(t) + \frac{\delta \varepsilon(\mathbf{r}, t)}{\delta \mathbf{r}}. \quad (1.3)$$

Por último, a integração temporal semi-implícita é uma mistura dos dois esquemas, equacionando os novos pontos da superfície no instante $t + \Delta t$ com base em forças deste mesmo instante, que

precisam ser formuladas em função dos novos pontos, e também usando forças do instante anterior t . Desta forma, o esquema herda vantagens e desvantagens dos dois lados, sendo uma excelente opção que balanceia as características relacionadas a uma integração temporal. Esta integração também pode ser equacionada a partir da equação 1.1, isolando-se os termos a serem encontrados em $t + \Delta t$ do lado esquerdo com base nos termos conhecido de t do lado direito

$$\mu \frac{\partial^2 \mathbf{r}}{\partial t^2}(t + \Delta t) + \varrho \frac{\partial \mathbf{r}}{\partial t}(t + \Delta t) = \mathbf{f}(\mathbf{r}, t) - \frac{\delta \varepsilon(\mathbf{r}, t)}{\delta \mathbf{r}}. \quad (1.4)$$

Na literatura muito já foi pesquisado sobre as integrações temporais para a simulação de tecidos. Os métodos de integração explícita dominaram os trabalhos da década de 90 [2], até que, em 1998, Baraff e Witkin demonstraram a superioridade do esquema numérico de integração implícita em comparação a explícita, atingindo uma dinâmica de tecido estável com o passo de tempo mais largo [13]. Para aliviar o alto custo computacional deste esquema, os autores propuseram um método do gradiente conjugado modificado. Os resultados foram tão promissores que desde então uma série de trabalhos tem se dedicado a aprimorar este sistema [24, 10].

Entretanto, o simples emprego de métodos implícitos não pode contornar algumas limitações dos modelos existentes, como uma resposta apropriada dos tecidos para forças de compressão. Neste sentido, o modelo proposto por Wu e de Melo considera esta resposta na modelagem do tecido, enquanto utiliza um método semi-implícito para montar um sistema de equações lineares e resolvê-lo com a decomposição \mathcal{LU} , atingindo simulações realistas do comportamento de tecidos [3, 4]. Porém, este método possui um alto custo computacional e impõe uma limitação relacionada ao número de pontos da malha devido à sua complexidade computacional. Esta baixa performance e limitação estão relacionadas com a dificuldade em linearizar a equação 1.1 para solucioná-la como um sistema de equações lineares, que resulta em um procedimento de complexidade computacional não linear. Desta forma, esta dissertação propõe avaliar as técnicas de integração explícita, as quais possuem um comportamento linear, para solucionar a equação 1.1 com o objetivo de contornar o problema de linearização das equações constitutivas sem sacrificar o realismo dos resultados visuais. Além disso, acredita-se que estas técnicas sejam mais apropriadas para as características peculiares do modelo, que é discretizado espacialmente com as diferenças finitas.

1.1.3 Problemas

Um dos problemas relacionados ao uso de uma integração temporal explícita refere-se à imposição em usar um passo de tempo pequeno. Esta imposição ocorre devido ao fato de se estimar as posições dos novos pontos da superfície do tecido em um instante $t + \Delta t$ usando as forças do tempo t . Quanto menor for este passo de tempo, mais suave é a variação das forças que atuam sobre a superfície entre

tempos subseqüentes e, conseqüentemente, menor é o erro de aproximação cometido. Normalmente, o passo de tempo de uma integração explícita gira em torno de 0.001. Esta limitação implica em outro problema, que diz respeito ao número de quadros que precisam ser gerados para uma animação com o intuito de atingir um determinado estado final. Assim, para se simular 10 segundos de uma simulação de tecidos que utiliza uma integração explícita, usando um passo de tempo de 0.001, são necessárias 10000 iterações, sendo cada uma delas correspondente a um novo estado do tecido, ao passo que uma integração implícita que agüente um passo de tempo de 0.01, precisaria de apenas 1000 iterações para simular o mesmo tempo total. Contudo, cada iteração de uma integração explícita, em geral, costuma ser computada mais rapidamente do que uma implícita.

Devido à existência de diversos métodos de integração explícita, os quais diferem em alguns aspectos como a precisão numérica e o número de passos usados na solução de cada iteração, é possível que exista diferença nos valores máximos do passo de tempo, bem como nos resultados visuais, quando aplicados para simular tecidos. Os métodos numéricos de integração explícita são usados, normalmente, para determinar o valor de um ponto aplicado a uma função sem saber sua formulação algébrica, porém conhecendo a expressão de sua derivada e ainda um valor inicial de um ponto da função. A maioria destes métodos baseia-se na expansão de série de Taylor, desprezando termos conforme a precisão que se deseja obter. No caso da simulação de tecidos, particularmente no modelo de tecido proposto por de Wu e de Melo, não se conhece a definição das derivadas, que em relação à posição dos pontos correspondem a sua velocidade e aceleração, dadas pelo cálculo de forças em uma iteração do modelo [4]. Desta forma, estes métodos explícitos precisam ser adaptados antes de serem aplicados no modelo de tecido ao longo do tempo.

Um outro problema relacionado à simulação de tecidos é a definição das condições de contorno, quando se utiliza a técnica de diferenças finitas para a discretização espacial do modelo de tecido, como em [18, 4]. Este problema consiste no cálculo de alguns elementos do modelo, como tensores de métrica e curvatura, nas fronteiras da malha discreta, nas quais pontos adjacentes necessários para os cálculos podem não existir. No modelo de tecido selecionado é utilizada a mesma estratégia usada por Terzopoulos et al., que corresponde em zerar os elementos nos pontos em que não existirem os pontos adjacentes necessários [18]. Esta consideração pode provocar, em alguns tipos de simulação, um desequilíbrio das forças internas nos pontos zerados, causando problemas de instabilidade. Este trabalho apresenta uma solução para contornar este problema.

1.2 Conteúdo do Trabalho

Este trabalho tem como principal conteúdo explorar a aplicação de técnicas de integração explícita para obter as variações do modelo de tecido, proposto por Wu e de Melo [3], ao longo do tempo.

1.2.1 Motivação

A motivação deste trabalho é desenvolver uma simulação de tecidos utilizando uma abordagem alternativa, em relação aos trabalhos atuais, que consiste em combinar um modelo de tecido baseado na mecânica de contínuos com uma integração temporal explícita. Para tanto, é usado um modelo de superfícies deformáveis geometricamente exato, proposto por Wu e de Melo [3], que, derivado de uma teoria de cascas [25], se destaca dos demais modelos baseados na mecânica de contínuos por possuir um tratamento apropriado na formação de dobras sob forças de compressão. Neste modelo de tecido, o realismo dos resultados é obtido com precisão física, o que proporciona um controle intuitivo dos parâmetros do material do tecido. Contudo, em sua implementação original o modelo é integrado ao longo do tempo com um esquema semi-implícito, que possui um alto custo computacional. Assim, existe o interesse em verificar se este modelo pode ser implementado com um método de integração de melhor desempenho, porém sem perder realismo nos resultados visuais.

1.2.2 Hipótese

A natureza do modelo de tecido escolhido impõe o uso de pequenos passos de tempo para garantir que a variação dos elementos geométricos ao longo do tempo seja suave. Desta forma, não é possível aproveitar uma das grandes vantagens de se usar os métodos implícitos, que é poder usar um passo de tempo mais largo em relação aos métodos explícitos. Este trabalho parte da hipótese de que, diferentemente da tendência dos trabalhos em curso, uma integração temporal explícita pode solucionar eficientemente a equação 1.1, gerando resultados realistas, desde que a variação entre as formas geométricas de dois passos de tempos subsequentes seja suave. É esperado que o realismo esteja inerente ao modelo proposto por de Wu e de Melo [3], enquanto o uso de um método de integração explícita proporcione um ganho de eficiência em relação à integração semi-implícita feita na implementação original do modelo.

1.2.3 Objetivos

O principal objetivo deste trabalho é melhorar a eficiência computacional da solução da equação 1.1, investigando qual a melhor técnica de integração explícita que pode ser aplicada para integrar ao longo do tempo o modelo de tecido selecionado. Como objetivos secundários, pretendesse apresentar uma validação prática mais aprofundada das principais características do modelo, mostrando através de resultados visuais de simulações de tecidos, geradas com o melhor método explícito avaliado, sua eficiência na criação de dobras com diversidade e em situações em que só existem forças de compressão.

Além disso, esta dissertação tem o objetivo de apresentar um projeto de *software* com uma nova estruturação da implementação feita por de Melo [4], com o intuito de tornar os códigos mais modulares para facilitar a integração de novos algoritmos, como os referentes ao tratamento de colisões. Neste *software* também é proposta e desenvolvida uma nova interface gráfica visando aumentar a usabilidade do usuário para manipular os parâmetros do modelo e gerar simulações de tecidos.

1.2.4 Escopo

Faz parte do escopo deste trabalho a análise da adequabilidade de técnicas de integração explícita para solucionar a equação 1.1 no modelo proposto por Wu e de Melo [3] em termos de: expectativa visual, estabilidade numérica e eficiência computacional. Para tanto, assume-se que a topologia da superfície que representa o tecido é uma malha retangular, pois se adota o mesmo esquema de discretização espacial proposta por Terzopoulos et al. [18] para determinar as variações das equações constitutivas do modelo deformável. Está fora do escopo desta dissertação incorporar algoritmos de tratamento de colisões à simulação de tecidos, embora eles sejam considerados na modularização do código-fonte para que possam ser agregados facilmente em trabalhos futuros.

1.2.5 Contribuições

A principal contribuição desta dissertação é desenvolver uma implementação, eficiente computacionalmente, do modelo de tecido proposto por Wu e de Melo [3]. Para este propósito, as técnicas de integração explícita mais populares são implementadas para serem usadas como solução numérica do modelo de tecido ao longo do tempo, o qual é baseado na mecânica de contínuos. Assim, apesar do uso de métodos explícitos para simular tecidos não ser uma novidade, bem como utilizar o modelo de tecido escolhido para este mesmo propósito, a combinação destas técnicas de integração com este tipo de modelo é inédita na literatura.

Esta dissertação também contribui fornecendo um procedimento para especificar os parâmetros do material de um tecido, bem como os parâmetros de simulação, que juntos, determinam uma grande diversidade de dobras que podem ser formadas. Para tanto, um projeto de *software* é apresentado visando desenvolver uma nova estrutura de implementação para o modelo de tecido usado, criando uma interface lógica, com o intuito de facilitar a integração de novos algoritmos, e também uma interface gráfica, objetivando aumentar a usabilidade do usuário para produzir simulações de tecidos. Além disso, esta dissertação contribui com uma validação prática aprofundada do modelo, destacando os resultados visuais de suas principais características.

1.3 Organização da Dissertação

Esta dissertação está organizada em seis capítulos, os quais, com exceção desta introdução, são brevemente descritos nos parágrafos abaixo.

Capítulo 2 discute brevemente os trabalhos relacionados com a simulação de tecidos, destacando suas abordagens de modelagem e suas integrações temporais. Ao final deste capítulo, o modelo de superfície deformável usado neste trabalho é explicado de forma detalhada.

Capítulo 3 apresenta a discretização espacial do modelo de tecido, selecionado neste trabalho, para descrever de que forma os métodos explícitos mais conhecidos podem ser aplicados sobre ele. Após mostrar as formulações, vantagens e desvantagens, estes métodos são implementados sobre o modelo e seus resultados analisados. A escolha do método de Verlet perante os demais é justificada em termos de eficiência computacional, estabilidade numérica e qualidade das imagens produzidas.

Capítulo 4 descreve o projeto de *software* que desenvolve uma interface lógica e gráfica para a abordagem de simular tecidos proposta neste trabalho. Ao final, alguns exemplos de simulação de tecidos são apresentados para mostrar sua usabilidade.

Capítulo 5 mostra os resultados visuais obtidos com o simulador desenvolvido, focando a qualidade e a diversidade das dobras produzidas, bem como a precisão física do modelo de tecidos. Este capítulo mostra que a abordagem proposta de simular tecidos é viável, além de permitir uma validação prática mais aprofundada do modelo, constatando sua superioridade na produção de dobras em relação aos outros modelos existentes.

Capítulo 6 resume as principais conclusões desta dissertação, destacando as suas contribuições e limitações. Por fim, as possibilidades de trabalhos futuros são comentadas.

Capítulo 2

Simulações de Tecidos

Este capítulo apresenta um panorama dos trabalhos atuais relacionados à simulação de tecidos, destacando neles a combinação da modelagem de tecidos com a solução numérica adotada, com o objetivo de contextualizar a proposta desta dissertação em uma lacuna existente na literatura. Como este trabalho é voltado para uma nova abordagem de simular a dinâmica de tecidos, neste capítulo são discutidos apenas modelos físicos, tanto os baseados na mecânica de contínuos quanto na de partículas, deixando de fora os modelos puramente geométricos, como [20, 21], por eles serem voltados para a simulação de tecidos em um estado estático. Para melhor organizar a discussão dos modelos físicos, a seção 2.1, que descreve os modelos baseados na mecânica de partículas, e a seção 2.2, que descreve os baseados na mecânica de contínuos, estão divididas de acordo com a solução numérica usada para integrar temporalmente os modelos, ou seja, com subseções referentes às integrações implícitas, semi-implícitas e explícitas. Após os principais modelos serem comentados, eles são sintetizados e classificados de acordo com a abordagem física e o método numérico usado na seção 2.3.

Em seguida, este capítulo justifica a escolha e detalha o modelo físico usado neste trabalho, o qual é proposto por Wu e de Melo, baseado na mecânica de contínuos e discretizado no espaço com diferenças finitas [3]. Este modelo de superfície deformável é fundamentado em uma teoria de casca geometricamente exata, conhecida como superfície de Cosserat elástica [25]. O resultado é um comportamento apropriado do tecido quando submetido às forças de compressão e um controle intuitivo dos parâmetros do material por possuírem interpretações geométricas, que são aspectos diferenciais deste modelo em relação aos demais baseados na mecânica de contínuos. Desta forma, uma explicação aprofundada deste modelo é feita na seção 2.4, dividindo-a em subseções que discutem a teoria geral da superfície de Cosserat e as restrições feitas sobre ela no modelo de tecido usado nesta dissertação. Por fim, a seção 2.5 traz considerações finais.

2.1 Modelos Físicos baseados na Mecânica de Partículas

Os modelos físicos que utilizam a mecânica de partículas representam os tecidos como uma grade de pontos interligados entre si com diferentes tipos de ligação. Conforme estes pontos ou partículas vão se deslocando devido a deformações, as forças exercidas sobre cada partícula são calculadas com base nas tensões existentes entre a ligação com as partículas adjacentes. O modelo mais conhecido é o modelo massa-mola, no qual cada partícula está ligada à outra por molas e as tensões são dadas pela lei de Hooke $\mathbf{F} = k\mathbf{x}$, que expressa a força elástica \mathbf{F} em relação ao coeficiente de rigidez da mola k e seu deslocamento \mathbf{x} a partir do ponto de repouso.

2.1.1 Integrações Explícitas

O trabalho precursor da mecânica de partículas é o trabalho de Haumann, que em 1987 desenvolveu o primeiro modelo massa-mola para modelar a dinâmica de uma superfície deformável, utilizando uma integração temporal explícita [1]. Neste modelo cada vértice da superfície possui uma massa e é conectado ao seu vizinho por uma mola, de forma que a cada duas faces triangulares, formadas por quatros pontos-massa, existe um outro tipo de ligação de mola, conforme a figura 2.1a. Incorporando forças aerodinâmicas como o vento e a resistência ao ar e, um algoritmo de detecção de colisões, o autor utilizou como solução numérica o esquema explícito de Euler para frente.

Em 1995, Provot incrementou o sistema de Haumann propondo o uso de três tipos de molas que conectam os pontos-massa da grade bidimensional: molas de estrutura, as quais são usadas entre todos os vizinhos imediatos nas direções horizontais e verticais; molas de flexão, usadas entre vizinhos espaçados de um ponto de distância também nas direções horizontais e verticais; e molas de cisalhamento, usadas entre vizinhos imediatos nas diagonais [2]. Assim, cada ponto-massa da grade de partículas pode ter sua tensão resultante estimada com pontos em uma vizinhança pré-estabelecida, como mostra a figura 2.1(b). Este trabalho conseguiu solucionar alguns dos principais problemas da época, que eram os efeitos de superelasticidade no tecido e instabilidade na simulação. O esquema explícito de Euler para frente foi aplicado para obter as soluções numéricas ao longo do tempo, descrito pelas equações

$$\begin{aligned}\mathbf{a}_{i,j}(t + \Delta t) &= \frac{\mathbf{F}_{i,j}(t)}{m}, \\ \mathbf{v}_{i,j}(t + \Delta t) &= \mathbf{v}_{i,j}(t) + \Delta t \mathbf{a}_{i,j}(t), \\ \mathbf{P}_{i,j}(t + \Delta t) &= \mathbf{P}_{i,j}(t) + \Delta t \mathbf{v}_{i,j}(t).\end{aligned}$$

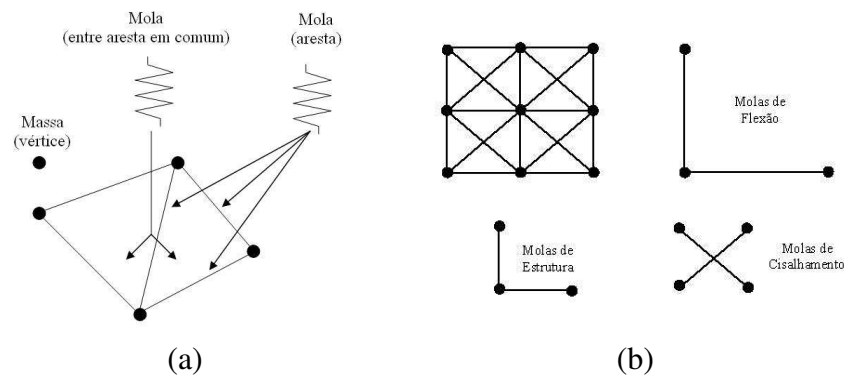


Figura 2.1: Modelos massa-mola: (a) Haumann [1] em 1987; e (b) Provot [2] em 1995.

2.1.2 Integrações Implícitas

Também usando a mecânica de partículas, mas sem considerar a abordagem massa-mola, Breen e House consideraram que o comportamento macroscópico do tecido é obtido modelando a sua estrutura microscópica e desenvolveram um modelo para o caimento de tecidos, representando as interações de seus fios em um reticulado de pontos tridimensionais [16]. Esta abordagem considera que um tecido seja melhor representado por um mecanismo de interação entre as partes mecânicas do que por uma substância contínua [22]. As conexões mecânicas entre as partículas são representadas como funções de energia gerada por distintas deformações, de forma que a energia acumulada E em cada partícula da grade é obtida por

$$E = E_{stretch} + E_{shear} + E_{bending}.$$

Cada uma destas parcelas de energia é calculada com o somatório de funções que medem o deslocamento de um ponto com base em expressões angulares. Como os autores observaram que um simples ponto pode curvar-se para fora do plano da superfície do tecido, eles propuseram descrever este fenômeno pelo ângulo formado entre cada conjunto de três partículas cruzadas adjacentes [16]. A princípio, este modelo estava focado apenas na configuração estável de tecidos após o caimento sobre alguns objetos. Esta configuração estável era alcançada com uma técnica de gradiente estocástico descendente, usada para relaxar as partículas até atingirem um estado de equilíbrio. Para o tratamento de colisões, Breen e House consideraram uma parcela adicional E_{repel} na composição da energia E para assegurar uma distância mínima entre os pontos. Em 1994, os mesmos autores mostraram que o modelo pode ser usado para reproduzir de maneira precisa o caimento de diferentes materiais, porém encontrando apenas o estado de equilíbrio final, sem produzir respostas dinâmicas necessárias para uma animação [16]. Em um trabalho de Breen e House em 2000, eles recomendaram usar uma integração temporal implícita para estender o modelo para este propósito [22].

2.1.3 Integrações Semi-implícitas

Os principais trabalhos que unem a mecânica de partículas com uma integração semi-implícita foram publicados nesta última década. Em 2002, Choi e Ko [11] propuseram usar uma solução com um esquema semi-implícito, em conjunto com um modelo físico baseado na mecânica de partículas, inspirado no modelo de tecido de Breen et al. [16]. Eles usaram uma fórmula de diferenças para trás de segunda ordem, que define as novas posições \dot{x} da superfície do tecido como

$$\dot{x} = \frac{1}{\Delta t} \left(\frac{3}{2}x^{n+1} - 2x^n + \frac{1}{2}x^{n-1} \right).$$

Para solucionar esta equação os autores usaram um método de gradiente conjugado pré-condicionado. Contudo, Choi e Ko constataram que somente o emprego de métodos semi-implícitos não garante a estabilidade numérica do modelo em situações que ocorrem deformações de *buckling*, que corresponde ao ato de enrugar, amassar ou criar dobras irregulares em um objeto deformável.

A instabilidade gerada após situações de *buckling* é muito comum em modelos massa-mola, pois quando acontecem grandes deformações para fora do plano da superfície do tecido, as molas que formam sua estrutura têm que ser suficientemente rígidas para não haver grandes deformações de métrica, mas flexíveis para a mudança de curvatura. Baraff e Witkin introduziram forças fictícias para contornar este problema [13], as quais muitas vezes acabavam eliminando qualquer tipo de dobra mais acentuada que pudesse ser formada, diminuindo o realismo da simulação [11]. A principal contribuição do modelo físico de Choi e Ko foi propor uma técnica usada no pós-processamento de *bucklings* sem recorrer a nenhuma força dissipadora fictícia, aumentando a estabilidade e o realismo da simulação [11]. Eles mostraram seus resultados com vestuários sobre atores virtuais.

Também em 2002, Bridson et al. propuseram algumas técnicas robustas para tratar colisões, contato e fricção em simulações de tecidos [24]. Estes autores utilizaram o modelo massa-mola de Provot [2], enfatizando que o foco deste trabalho era o controle robusto do tratamento de colisões, e não a modelagem do tecido. Para a discretização do tempo, eles destacaram a separação da evolução do tempo na dinâmica interna do tecido (e do ambiente ao seu redor) em relação ao algoritmo que processa as colisões. Pode-se classificar a solução numérica adotada neste trabalho como semi-explícita, pois o cálculo das novas posições é feito com um método explícito, enquanto a velocidade é calculada com uma equação implícita, similar àquela usada por Baraff e Witkin em [13].

No ano seguinte, Bridson et al. publicaram outro trabalho, incrementando o modelo de tecido anterior com várias considerações da mecânica de contínuos e expressões angulares para o cálculo das forças internas, deixando o modelo mais realista [10]. Eles também incrementaram o tratamento de colisões, solucionando problemas na formação de dobras e rugas que interpenetram a superfície do tecido com o objeto de colisão, como um personagem virtual. Neste último trabalho, os autores

destacam a solução numérica adotada, que descrevem como semi-implícita e detalham o tipo de cada passo do procedimento, descrito a seguir.

- Explícito: $\mathbf{v}^{n+\frac{1}{2}} = \mathbf{v}_n + \frac{\Delta t}{2} \mathbf{a}(t^n, x^n, v^n)$,
- Modificar $\mathbf{v}^{n+\frac{1}{2}}$ para obter $\tilde{\mathbf{v}}^{n+\frac{1}{2}}$, respeitando o algoritmo que processa as colisões,
- Explícito: $\mathbf{x}^{n+1} = \mathbf{x}_n + \Delta t \tilde{\mathbf{v}}^{n+\frac{1}{2}}$,
- Implícito: $\mathbf{v}^{n+1} = \mathbf{v}_{n+\frac{1}{2}} + \frac{\Delta t}{2} \mathbf{a}(t^{n+1}, x^{n+1}, v^{n+1})$,
- Modificar \mathbf{v}^{n+1} respeitando os limites que o algoritmo que processa as colisões impõe.

Assim, os autores agregam um esquema explícito chamado de *leapfrog*, que possui precisão de segunda ordem, para calcular as novas posições dos pontos da superfície, junto com um método implícito de regra trapezoidal, também de precisão de segunda ordem, para atualizar as velocidades.

2.2 Modelos Físicos baseados na Mecânica de Contínuos

Os modelos físicos baseados na mecânica de partículas não conseguem realmente refletir a forma da superfície na vizinhança de um ponto, comprometendo a precisão das forças internas. Breen et al. afirmaram, após apresentarem resultados comparativos entre implementações usando uma abordagem de partículas em relação a uma de contínuos, que quando o realismo é colocado em primeiro plano os modelos contínuos são mais apropriados, enquanto que, em termos de eficiência, os modelos de partículas são mais adequados [22]. Esta afirmação é enfatizada no trabalho de Bridson et al., em que os autores descreveram seu trabalho focado no tratamento de colisões com um modelo massa-mola porque o foco não era o realismo, mas indicaram que imagens da indústria cinematográfica, mais precisamente o quimono do personagem Yoda do filme Guerra nas Estrelas, foi criado com suas técnicas de colisões em adição a um modelo físico contínuo [24].

2.2.1 Integrações Implícitas

Um dos primeiros trabalhos da mecânica de contínuos pertence a Feynman, no qual ele simulava propriedades mecânicas dos tecidos definindo um conjunto de funções de energia sobre um reticulado de pontos tridimensionais [19]. A energia total de seu modelo considerava deformações de métrica, de curvatura e termos relativos à gravidade. Então, as forças internas da superfície deformável eram calculadas minimizando as energias de tensão e de curvamento produzidas pelas deformações. Este autor utilizava a integração temporal implícita com um método de relaxação.

Em 1998, Baraff e Witkin apresentaram um trabalho demonstrando a superioridade do esquema numérico implícito em relação ao explícito para integrar temporalmente um modelo de tecido com

passos de tempo largos [13]. Estes autores descreveram as forças internas que atuam sobre a superfície deformável com expressões derivadas da mecânica de contínuos junto com expressões angulares. O resultado estava focado na eficiência e na estabilidade numérica em utilizar um método de integração implícita com o passo de tempo em torno de uma ordem maior que o usado pelos métodos explícitos. A solução era obtida montando um sistema simétrico de equações que era resolvido por um método modificado dos gradientes conjugados. Segundo Baraff e Witkin, a complexidade computacional do seu esquema implícito é de $O(n^{1.5})$.

Contudo, estava claro que a prioridade deste trabalho era a eficiência e a estabilidade, pois as imagens obtidas, principalmente em relação às dobras, não possuíam grande realismo. Segundo Choi e Ko, o realismo deste modelo era prejudicado pela adição de forças artificiais fictícias para solucionar colisões, o que evitava ou mesmo eliminava a formação de dobras [11]. Mais tarde, em 2003, Baraff et al. usou o mesmo modelo de tecido e integração temporal implícita para contribuir com técnicas novas de tratamento de colisões e auto-colisões que ocorrem quando o objeto virtual em contato com o tecido tem pontos que se interpenetram [9]. Posteriormente, neste mesmo ano, Bridson et al. apresentaram outra solução para o mesmo problema [10].

2.2.2 Integrações Semi-implícitas

Um ano depois do trabalho de Feynman [19], Terzopoulos et al. propuseram um modelo de superfície deformável baseado na teoria da elasticidade e fundamentado na mecânica de contínuos [18]. Os autores descreveram uma formulação sem o objetivo de simular precisamente um determinado tipo de objeto deformável, propondo um sistema físico capaz de produzir com qualidade alguns tipos de comportamento sem possuir uma formulação muito complexa. Eles estimavam a força elástica do modelo através das variações da energia potencial, que eram calculadas pela variação dos tensores métricos e de curvatura do estado do objeto deformado em relação ao estado não deformado. Neste trabalho foi utilizada uma integração temporal semi-implícita, em que a solução da posição \mathbf{r} dos pontos da superfície no próximo instante $t + \Delta t$ era formulada como

$$\mathbf{A}_t \mathbf{r}_{t+\Delta t} = \mathbf{g}_t, \quad (2.1)$$

com a matriz \mathbf{A} definida por

$$\mathbf{A}_t = \mathbf{K}(\mathbf{r}_{t+\Delta t}) + \left(\frac{1}{\Delta t^2} \mathbf{M} + \frac{1}{2\Delta t} \mathbf{C} \right),$$

e o vetor \mathbf{g} por

$$\mathbf{g}_t = \mathbf{f}_t + \left(\frac{1}{\Delta t^2} \mathbf{M} + \frac{1}{2\Delta t} \mathbf{C} \right) \mathbf{r}_t + \left(\frac{1}{\Delta t} \mathbf{M} + \frac{1}{2\Delta t} \mathbf{C} \right) \mathbf{v}_t,$$

sendo

$$\mathbf{v}_t = \frac{\mathbf{r}_t - \mathbf{r}_{t-\Delta t}}{\Delta t}.$$

Desta forma, o esquema adotado é classificado como semi-implícito porque os termos da matriz \mathbf{A}_t , que multiplica o vetor incógnita $\mathbf{r}_{t+\Delta t}$, são formulados em função dos novos pontos a serem encontrados em $t + \Delta t$, e o vetor \mathbf{g}_t do lado direito da equação 2.1, obtido com termos de instantes anteriores, como a força externa e a velocidade. Tanto a matriz \mathbf{A} , quanto a matriz \mathbf{K} , sendo esta última chamada de matriz de rigidez, são de ordem $n^2 \times n^2$, em que n é o número total de pontos que formam a superfície. Terzopoulos et al. resolveram o sistema 2.1 com métodos diretos, como a decomposição de Choleski e métodos de relaxação, como a de Gauss-Seidel [18]. O custo computacional deste procedimento de integração semi-implícita é aproximadamente da ordem de $O(n^2)$, devido às dimensões das matrizes que formam o sistema linear.

Um dos grandes problemas dos modelos baseados em mecânica de contínuos para modelar o comportamento de tecidos deve-se à dificuldade em definir a diversidade de dobras que podem ser formadas. Enquanto as deformações de métrica dentro do plano da superfície do tecido são bem estimadas com a variação do tensor métrico, as deformações de curvatura na direção para fora do plano da superfície ainda não atingiram o mesmo consenso. Isto ocorre porque existem deformações de curvatura nos tecidos que podem ser provocadas somente por forças tangenciais, como em dobras que se formam ao esticar ou comprimir um tecido somente em uma direção. Desta forma, a ligação entre as medidas de métrica e curvatura é essencial na modelagem de tecidos. A teoria da superfície de Cosserat, a qual é fundamentada na mecânica de contínuos, possui esta ligação representada pelo produto entre as variações de métrica e de curvatura [25].

Utilizar uma teoria de casca como a superfície de Cosserat para modelagem de tecidos não é uma novidade. Depois do trabalho de Green et al. ser publicado em 1965, descrevendo de forma completa a teoria geral de uma superfície de Cosserat [25], alguns autores começaram a usá-la para modelagem de objetos deformáveis. Eischen et al. apresentaram em 1996 um modelo fundamentado em uma superfície de Cosserat [15], após a publicação de três artigos por Simo e Fox em 1989, os quais mostraram que, apesar da complicada formulação, ela pode ser implementada de uma maneira numérica eficiente [17]. A principal contribuição deles foi descobrir uma parametrização que evita termos como os símbolos de Christoffel e os coeficientes da segunda forma fundamental, adotando relações que não associam explicitamente as quantidades deformadas com as estáticas.

Em 2004, Wu e de Melo apresentaram um modelo de superfície deformável visando o controle de caimentos e dobras, baseado em uma superfície de Cosserat [3]. Neste trabalho os autores propõem simplificações e restrições, específicas para tecidos, sobre a teoria geral de superfície de Cosserat [25], conseguindo obter um tratamento apropriado das dobras de tecidos quando eles são submetidos às forças de compressão. Este comportamento adequado, que aumenta o realismo do modelo, ocorre em virtude da consideração do termo referente ao acoplamento entre as variações de métrica com as de curvatura para estimar a energia interna, ao contrário de Terzopoulos et al., que desprezam este termo [18]. Esta consideração é equivalente a dizer que o modelo incorpora as condições de compatibilidade de Weingarten, as quais governam as deformações dentro e fora do plano da superfície ao longo do tempo, garantindo assim a regularidade da superfície deformável durante toda a simulação.

Além disso, esse modelo deformável apresenta um controle mais intuitivo dos parâmetros do material do tecido, principalmente quando comparado aos modelos massa-mola, que consideram a definição do material atribuindo-se valores aos coeficientes de rigidez das molas que formam o tecido. No modelo proposto por Wu e de Melo um material é definido com um conjunto de três parâmetros que possuem interpretações geométricas, como resistência à métrica, resistência à curvatura e grau de acoplamento entre métrica e curvatura [3]. Os autores utilizam como solução numérica um esquema muito similar ao proposto por Terzopoulos et al. [18], porém resolvendo o sistema com uma decomposição LU . O custo computacional desta solução numérica é elevado e, assim como o de Terzopoulos et al., possui complexidade de ordem $O(n^2)$, sendo n o número de pontos que formam a superfície do tecido. Devido a este baixo desempenho computacional, os resultados deste trabalho foram produzidos com malhas de pequena resolução, conforme as imagens de simulações de tecidos nas figuras 2.2(a), que mostra dobras em um pedaço de pano, e 2.2(b), que mostra um caimento de uma toalha sobre uma mesa redonda.

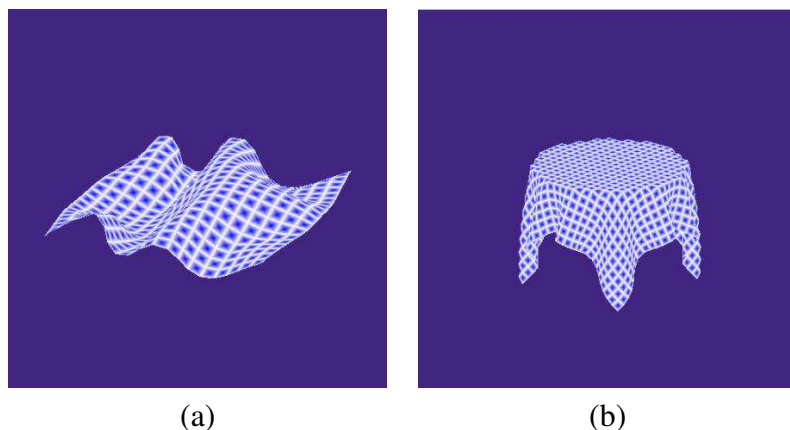


Figura 2.2: Resultados do trabalho de Wu e de Melo [3] em 2004: (a) dobras; e (b) caimento.

2.3 Classificação dos Modelos

Como visto na seção 2.2, ainda não existe na literatura um trabalho que aplique métodos de integração explícita sobre um modelo de tecido baseado na mecânica de contínuos. Isto se deve principalmente à ascensão dos modelos baseados na mecânica de partículas na metade da década de 90, como nos trabalhos de Breen et al. [16] e Provot [2], que possuem grande facilidade de implementação e eficiência computacional. Além disso, alguns anos depois, Baraff e Witkin [13] apresentaram um modelo contínuo em conjunto com uma integração implícita capaz de realizar a dinâmica de tecidos de maneira estável para valores de passo de tempo maiores do que os usados por integrações explícitas. Contudo, este último trabalho estava mais focado em atingir uma grande eficiência computacional do que o realismo nas simulações, tendo como ponto forte a solução numérica adotada, ao invés da modelagem de tecido. A partir daí, surgiram trabalhos que agregavam as vantagens da modelagem baseada na mecânica de partículas com as de se usar a integração implícita.

O processamento de colisões em uma simulação de tecidos é de fundamental importância para se alcançar um alto realismo nos resultados visuais, porém este processo envolve algoritmos de alto custo computacional. Assim, os pesquisadores começaram a investir mais esforços em aprimorar estes algoritmos sobre um *framework* de modelagem da dinâmica interna de tecidos bem aceito, que é o de utilizar integrações implícitas e semi-implícitas para solucionar numericamente modelos baseados na mecânica de partículas. Contudo, esta dissertação propõe um *framework* alternativo para simular tecidos, considerando que para agregar realismo e eficiência computacional é mais apropriado utilizar um modelo de tecido baseado na mecânica de contínuos com uma solução explícita ao longo do tempo. Desta forma, o realismo está inerente ao uso de um modelo de grande precisão física, enquanto o desempenho computacional está relacionado ao uso de uma integração explícita.

Os trabalhos comentados nas seções 2.1 e 2.2 estão sintetizados na tabela 2.1, classificados de acordo com a abordagem de modelagem física e a solução numérica usada. Nela se nota uma lacuna exatamente onde se encaixa a proposta deste trabalho. Assim, com o objetivo de melhor preencher esta lacuna escolheu-se, dentre os diversos modelos baseados na mecânica de contínuos, o modelo proposto por Wu e de Melo [3, 4]. Ele é explicado detalhadamente na seção 2.4, enquanto os possíveis métodos explícitos que podem solucioná-lo ao longo do tempo é o tema do próximo capítulo desta dissertação.

Modelos Físicos / Integrações	Explícita	Semi-Implícita	Implícita
Partículas	Haumann [1] Provot [2]	Choi e Ko [11] Bridson et al. [24, 10]	Breen et al. [16, 26, 22]
Contínuos		Terozpoulos et al. [18] Wu e de Melo [3, 4]	Feynman [19] Baraff et al. [13, 9]

Tabela 2.1: Classificação dos trabalhos apresentados nas seções 2.1 e 2.2.

2.4 Uma modelagem baseada em uma Superfície de Cosserat

Devido ao comportamento característico de um tecido, de ser muito flexível para curvar e pouco para esticar, o ponto crítico de sua modelagem acontece no momento de descrever a diversidade de dobras que nele se formam com facilidade. Neste sentido, o modelo de superfície deformável escolhido, proposto por Wu e de Melo, é apropriado para descrever uma grande variedade de dobras formadas sob distintas condições, incluindo compressões e distensões [4]. Além disso, neste modelo o material de um tecido é definido com parâmetros que possuem significados geométricos intuitivos, que são: a resistência à mudança de métrica, a resistência à mudança de curvatura e o grau de formação de dobras.

O modelo de superfície deformável selecionado baseia-se em um modelo de casca geometricamente exato, conhecido como superfície de Cosserat [25] e descrito resumidamente na subseção 2.4.1. A formulação desta superfície é complexa, pois ela utiliza muitos elementos da geometria diferencial, além de integrar princípios termodinâmicos, para descrever de forma geral a dinâmica de uma superfície sob condições específicas de deformação. Green et al. derivaram alguns casos especiais, como o da superfície ser elástica, em [25]. Neste trabalho esta teoria é exposta de forma simplificada e resumida, para que fiquem claras as restrições e simplificações, específicas para a dinâmica de tecidos, feitas sobre ela. O modelo considerado neste trabalho é detalhado na subseção 2.4.2, usando como principal referência a tese de doutorado apresentada por de Melo [4].

2.4.1 Superfície de Cosserat

A teoria de superfície de Cosserat foi proposta pelos irmãos Cosserat em 1909, redescoberta por Ericksen e Truesdell em 1958 para a modelagem de corpos orientados e, em 1965, usada para modelagem de cascas por Green et al. [25]. Estes últimos autores integraram os princípios dinâmicos e termodinâmicos da mecânica de contínuos à análise da dinâmica de uma superfície de Cosserat e a descreveram de forma geral e completa. Esta descrição é apresentada resumidamente a seguir fornecendo primeiro alguns conceitos de geometria diferencial para então definir uma superfície de Cosserat, suas medidas físicas e geométricas e as forças que atuam sobre ela.

Consideram-se um sistema fixo de coordenadas cartesianas z^i ($i = 1, 2, 3$) e um sistema arbitrário de coordenadas curvilíneas x^i ($i = 1, 2, 3$), de tal forma que a transformação entre estes referenciais não seja singular e tenha uma única inversa, ou seja, $\det \left[\frac{\partial z^i}{\partial x^j} \right] > 0$. Considera-se também uma superfície s , mergulhada no espaço euclidiano tridimensional z^i , definida pela equação $x^3 = 0$ e assumem-se as coordenadas convencionadas x^α ($\alpha = 1, 2$) em s e x^3 na direção normal à s . As coordenadas x^i são chamadas de convencionadas porque qualquer ponto em s tem as mesmas coordenadas curvilíneas no estado inicial e no estado deformado.

Adicionalmente, cada ponto $p = (x^1, x^2) \in s$, em um instante de tempo t , é representado em relação à origem do sistema cartesiano fixo pelo vetor posição $\mathbf{r}(x^1, x^2, t)$. No instante inicial $t = 0$, chamado t_0 , a superfície s é representada por S e o vetor posição \mathbf{r} por \mathbf{R} , de forma que $\mathbf{r}(x^1, x^2, t_0) = \mathbf{R}(x^1, x^2)$. Esta mesma convenção é usada para diferenciar os valores iniciais e correntes de outros elementos da superfície, como também se convencionou $f_{,\alpha}$ representando a derivada parcial da função f em relação à coordenada x^α e a omissão do símbolo \sum para somatórias. A figura 2.3 ilustra todas as considerações feitas até aqui exibindo configurações da superfície s em dois momentos diferentes: em um estado sem deformações, no instante inicial t_0 , e em um estado deformado, no instante $t = t_0 + \Delta t$, chamado t_1 . O símbolo Δt é conhecido como passo de tempo.

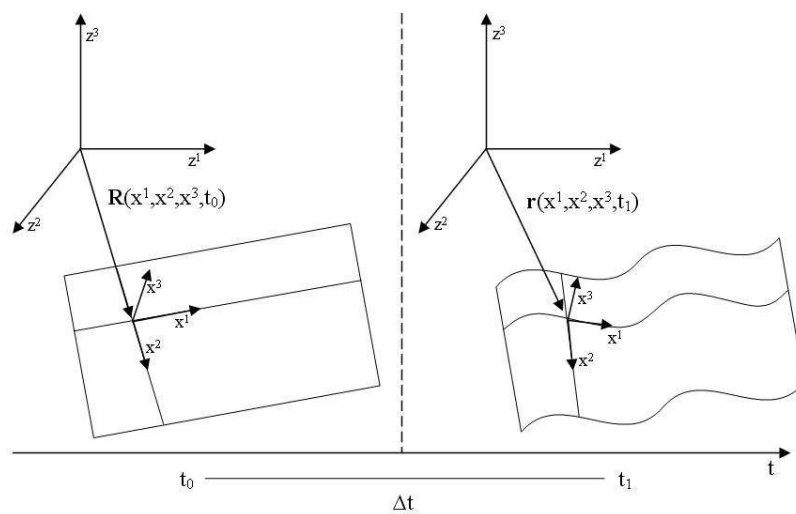


Figura 2.3: Referenciais fixo e arbitrário em instantes diferentes.

Em qualquer ponto p sobre a superfície s , em um instante t , é possível definir dois vetores de base \mathbf{a}_1 e \mathbf{a}_2 , contidos no plano tangente ao ponto p , que são vetores linearmente independentes e podem ser expressos pela derivada parcial do vetor posição \mathbf{r} em relação às coordenadas x^1 e x^2

$$\mathbf{a}_\alpha(x^1, x^2, t) = \frac{\partial \mathbf{r}}{\partial x^\alpha}(x^1, x^2, t), \quad (2.2)$$

com $\alpha = 1, 2$. O vetor normal \mathbf{n} , representado por \mathbf{a}_3 , é o vetor que aponta na direção perpendicular ao plano tangente à p e normalmente é definido como o produto vetorial entre os vetores de base

$$\mathbf{n} = \mathbf{a}_3(x^1, x^2, t) = \frac{\frac{\partial \mathbf{r}}{\partial x^1}(x^1, x^2, t) \times \frac{\partial \mathbf{r}}{\partial x^2}(x^1, x^2, t)}{\left| \frac{\partial \mathbf{r}}{\partial x^1}(x^1, x^2, t) \times \frac{\partial \mathbf{r}}{\partial x^2}(x^1, x^2, t) \right|}. \quad (2.3)$$

Os vetores de base \mathbf{a}_1 e \mathbf{a}_2 , juntamente com o vetor normal \mathbf{a}_3 , formam um triedro que representa um sistema de referência local em cada ponto $p \in s$, conforme as direções x^i na figura 2.3.

Além dos elementos desse triedro, existem também os tensores métrico, métrico recíproco e de curvatura, associados a cada ponto $p \in s(t)$ e fundamentais para a descrição do modelo. Estes três tensores, juntos com os vetores que formam o referencial local, são elementos da geometria diferencial usados para expressar as deformações, forças e energias que atuam sobre uma superfície de Cosserat. Para tornar as equações mais genéricas, até o final deste capítulo é usada a convenção de que índices latinos (i, j, k) podem assumir os valores 1, 2 e 3, enquanto índices gregos (α, β, γ) apenas os valores 1 e 2, bem como omitir o argumento (x^1, x^2, t) das próximas definições por considerar que elas são válidas para todos os pontos da superfície em qualquer instante de tempo.

O tensor métrico $a_{\alpha\beta}$ é um tensor simétrico covariante de segunda ordem definido como positivo e pode ser representado pela matriz $\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$, na qual seus elementos são conhecidos na geometria diferencial como os coeficientes da primeira forma fundamental. Este tensor pode ser usado para determinar o comprimento de um pequeno arco ou ângulo sobre as curvas coordenadas de $s(t)$ e pode ser expresso pelo produto escalar entre os vetores de base \mathbf{a}_1 e \mathbf{a}_2

$$a_{\alpha\beta} = \frac{\partial \mathbf{r}}{\partial x^\alpha} \cdot \frac{\partial \mathbf{r}}{\partial x^\beta}. \quad (2.4)$$

O tensor métrico recíproco $a^{\alpha\beta}$ é um tensor simétrico contravariante de segunda ordem expresso pelo produto escalar entre os vetores de base recíprocos \mathbf{a}^α . Os vetores \mathbf{a}^α e \mathbf{a}_α são chamados de recíprocos se

$$\mathbf{a}_i \cdot \mathbf{a}^j = \delta_i^j = \begin{cases} 1 & \text{se } i = j \\ 0 & \text{se } i \neq j \end{cases}.$$

O determinante a da matriz formada pelos coeficientes do tensor métrico resulta em número real que muitas vezes é usado para estimar a área delimitada pelos vetores de base no plano tangente em um ponto qualquer da superfície $s(t)$. Os coeficientes do tensor métrico recíproco podem ser expressos em função dos termos do tensor métrico $a_{\alpha\beta}$

$$a^{11} = \frac{a_{22}}{a}, \quad a^{12} = a^{21} = -\frac{a_{12}}{a}, \quad a^{22} = \frac{a_{11}}{a}. \quad (2.5)$$

O tensor de curvatura $b_{\alpha\beta}$, assim como tensor métrico, também é um tensor simétrico covariante de segunda ordem e pode ser expresso pela matriz $\begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}$, em que seus elementos são conhecidos como coeficientes da segunda forma fundamental. Este tensor é usado para estimar a curvatura gaussiana de um ponto em relação à superfície e é expresso por

$$b_{\alpha\beta} = \mathbf{a}_3 \cdot \frac{\partial^2 \mathbf{r}}{\partial x^\alpha \partial x^\beta}. \quad (2.6)$$

Uma superfície de Cosserat é uma superfície mergulhada no espaço euclidiano tridimensional em que para cada um de seus pontos é assinalado um vetor deformável \mathbf{d} , não tangente à superfície s [25]. Este vetor, chamado diretor, não necessariamente é unitário ou está na direção normal à superfície s , mas possui a propriedade de permanecer invariável em comprimento sobre transformações de corpo rígido. Uma superfície de Cosserat, portanto, é uma superfície mergulhada em um espaço euclidiano tridimensional com vetores diretores deformáveis em cada ponto que satisfazem estas propriedades. O mergulho corresponde à maneira como a superfície é inserida no espaço circundante a ela e como se relacionam seus elementos de medição.

Dessa forma, considerando um vetor diretor \mathbf{d} associado a cada posição \mathbf{r} de um ponto $p \in s(t)$, pode-se caracterizar o movimento de uma superfície de Cosserat por diferentes configurações dos vetores \mathbf{r} e \mathbf{d} em instantes de tempo t diferentes

$$\mathbf{r}(t) \equiv \mathbf{r}(x^1, x^2, t), \quad \mathbf{d}(t) \equiv \mathbf{d}(x^1, x^2, t),$$

sobre a restrição de que o vetor $\mathbf{d}(t)$ precisa satisfazer

$$\mathbf{d}(t) = d_i \mathbf{a}^i(t) = d_\alpha \mathbf{a}^\alpha + d_3 \mathbf{a}^3.$$

A variação ao longo do tempo das medidas do tensor métrico, tensor de curvatura e vetor diretor define três medidas físicas, ou variáveis cinemáticas, que caracterizam as deformações em uma superfície de Cosserat. Estas variações representam as tensões e distensões que ocorrem em uma superfície $s(t)$ em relação ao seu estado não deformado S , expressas por:

1. Deformações de membrana ($\varepsilon_{\alpha\beta}$)

$$\varepsilon_{\alpha\beta} = \frac{1}{2}(a_{\alpha\beta} - A_{\alpha\beta}). \quad (2.7)$$

2. Deformações de curvatura ($\kappa_{\beta i}$)

$$\begin{aligned} \kappa_{\beta\alpha} &= -(b_{\beta\alpha} - B_{\beta\alpha}), \\ \kappa_{\beta 3} &= 0. \end{aligned} \quad (2.8)$$

3. Deformações de vetor diretor (γ_i)

$$\gamma_i = d_i - D_i. \quad (2.9)$$

Seja σ a área de $s(t)$, limitada por uma curva fechada \mathcal{C} e seja $\nu = \nu_\alpha \mathbf{a}^\alpha$ o vetor normal unitário exterior à \mathcal{C} e posicionado sobre s . Se \mathbf{N} é um campo vetorial tridimensional e se, para todos os campos arbitrários de velocidade \mathbf{v} , o escalar $\mathbf{N} \cdot \mathbf{v}$ representar a taxa de trabalho por unidade de comprimento de \mathcal{C} , então \mathbf{N} é um vetor de força de curva medido por unidade de comprimento. Similarmente, se \mathbf{M} é um campo vetorial tridimensional e se, para todo os campos arbitrários de velocidade diretora \mathbf{w} , o escalar $\mathbf{M} \cdot \mathbf{w}$ é a taxa de trabalho por unidade de comprimento, então \mathbf{M} é chamado de vetor de força diretora, por unidade de comprimento de \mathcal{C} . Estes campos vetoriais podem ser expressos em termos de \mathbf{a}_i

$$\mathbf{N} = N^i \mathbf{a}_i = \nu_\alpha \mathbf{N}^\alpha = \nu_\alpha (N^{i\alpha} \mathbf{a}_i) = (\nu_\alpha N^{i\alpha}) \mathbf{a}_i, \quad (2.10)$$

$$\mathbf{M} = M^i \mathbf{a}_i = \nu_\alpha \mathbf{M}^\alpha = \nu_\alpha (M^{i\alpha} \mathbf{a}_i) = (\nu_\alpha M^{i\alpha}) \mathbf{a}_i, \quad (2.11)$$

onde $N^{\beta\alpha}$, $N^{3\alpha}$, $M^{\beta\alpha}$ e $M^{3\alpha}$ são os tensores de superfície sobre transformações de coordenadas.

Seja $\mathbf{F} = F^i \mathbf{a}_i$ e $\mathbf{L} = L^i \mathbf{a}_i$, respectivamente, os campos tridimensionais de força atribuída e força diretora, por unidade de massa de $s(t)$, tais que $\mathbf{F} \cdot \mathbf{v}$ e $\mathbf{L} \cdot \mathbf{w}$, para todos \mathbf{v} e \mathbf{w} arbitrários, representem a taxa de trabalho por unidade de área de $s(t)$. Se μ é a densidade de massa por unidade de área de $s(t)$, Green et al. derivaram em [25] equações de equilíbrio de energia com a consideração de que $s(t)$ permanece inalterada sobre transformações de corpo rígido, obtendo a equação de conservação para o momento linear

$$\mu \dot{\mathbf{v}} - \mathbf{N}^\alpha|_\alpha = \mu \mathbf{F}(\mathbf{r}, t), \quad (2.12)$$

e para o momento angular

$$\mathbf{m} = \mathbf{M}^\alpha|_\alpha + \mu \bar{\mathbf{L}}(\mathbf{r}, t), \quad (2.13)$$

onde $\dot{\mathbf{v}} = \frac{\partial^2 \mathbf{r}}{\partial t^2}$, \mathbf{m} é o vetor diretor conjugado intrínseco e $\bar{\mathbf{L}}$ é a diferença entre a força diretora \mathbf{L} e a força inercial devido ao deslocamento do vetor \mathbf{d} . A barra vertical $|_\alpha$ representa a derivada covariante em relação à coordenada x^α .

O termo $N^\alpha|_\alpha$ corresponde à força interna de reação da superfície quando submetida às ações externas. Green et al. também descreveram a energia armazenada em uma superfície de Cosserat Elástica e a formularam para o caso especial do vetor diretor coincidir com o vetor normal unitário em $s(t)$, ou seja, $\mathbf{d} = \mathbf{a}_3$ para qualquer $p \in s$ em um instante t . Neste caso, $\gamma_i = 0$ e

$$N^{3\alpha} = M^{\alpha\beta}|_\beta + \mu \bar{L}^\alpha, \quad (2.14)$$

onde $\bar{L}^\alpha = \mathbf{L} \cdot \mathbf{a}_\alpha$ é a componente tangencial do vetor $\bar{\mathbf{L}}$ referente a \mathbf{a}_α e $M^{\alpha\beta}|_\alpha$ corresponde à derivada covariante de um tensor de superfície contravariante de ordem 2 em relação à x^α .

Considerando \mathcal{A} a energia interna acumulada em $s(t)$ por unidade de massa, Green et al. mostraram

ainda que as equações constitutivas de uma superfície de Cosserat Elástica, no caso em que $\mathbf{d} = \mathbf{a}_3$, podem ser expressas por

$$N^{*\alpha\beta} = \mu \frac{\partial \mathcal{A}}{\partial \varepsilon_{\alpha\beta}} \quad (2.15)$$

$$M^{i\alpha} = \mu \frac{\partial \mathcal{A}}{\partial \kappa_{i\alpha}}, \quad (2.16)$$

com

$$N^{\alpha\beta} = N^{*\alpha\beta} - b_{\lambda}^{\alpha} M^{\beta\lambda}, \quad (2.17)$$

onde

$$b_{\lambda}^{\alpha} = a^{\rho\alpha} b_{\lambda\rho} \quad (2.18)$$

são as relações de compatibilidade de Weingarten.

Se a superfície s é inicialmente homogênea, livre de forças diretoras e, se as deformações ocorrem em temperatura e entropia constantes, então a aproximação para a energia interna \mathcal{A} pode ser expressa em termos das quantidades cinemáticas $\varepsilon_{\alpha\beta}$ e $\kappa_{\alpha\beta}$, e dos parâmetros que caracterizam as propriedades do material, $H^{\alpha\beta\lambda\rho}$, $B^{\alpha\beta\lambda\rho}$ e $C^{\alpha\beta\lambda\rho}$

$$\mu_0 \mathcal{A} = \left[H^{\alpha\beta\lambda\rho} \varepsilon_{\alpha\beta} \varepsilon_{\lambda\rho} + B^{\alpha\beta\lambda\rho} \kappa_{\alpha\beta} \kappa_{\lambda\rho} + C^{\alpha\beta\lambda\rho} \varepsilon_{\alpha\beta} \kappa_{\lambda\rho} \right]. \quad (2.19)$$

O primeiro e segundo termo do lado direito da equação 2.19 são as formas quadráticas das medidas relativas à métrica e à curvatura, respectivamente, enquanto o terceiro termo possui um produto entre elas, representando um acoplamento de ambos os efeitos.

Além disso, alguns dos parâmetros do material na equação. 2.19 satisfazem condições de simetria se a superfície de Cosserat elástica for isotrópica. Denotando os coeficientes elásticos por β_i , estes parâmetros assumem a forma

$$\begin{aligned} H^{\alpha\beta\lambda\rho} &= H^{\beta\alpha\lambda\rho} = H^{\alpha\beta\rho\lambda} = H^{\lambda\rho\alpha\beta} \\ &= \beta_1 A^{\alpha\beta} A^{\lambda\rho} + \beta_2 (A^{\alpha\lambda} A^{\beta\rho} + A^{\alpha\rho} A^{\beta\lambda}) \\ B^{\alpha\beta\lambda\rho} &= B^{\beta\alpha\lambda\rho} = B^{\alpha\beta\rho\lambda} = B^{\lambda\rho\alpha\beta} \\ &= \beta_3 A^{\alpha\beta} A^{\lambda\rho} + \beta_4 (A^{\alpha\lambda} A^{\beta\rho} + A^{\alpha\rho} A^{\beta\lambda}) \\ C^{\alpha\beta\lambda\rho} &= C^{\beta\alpha\lambda\rho} = C^{\alpha\beta\rho\lambda} = C^{\lambda\rho\alpha\beta} \\ &= \beta_5 A^{\alpha\beta} A^{\lambda\rho} + \beta_6 (A^{\alpha\lambda} A^{\beta\rho} + A^{\alpha\rho} A^{\beta\lambda}), \end{aligned} \quad (2.20)$$

os quais são dependentes apenas dos coeficientes de elasticidade e da configuração inicial da superfície.

2.4.2 Restrições

Nesta subseção são apresentadas todas as restrições que de Melo fez em [4] sobre a teoria geral de superfície de Cosserat descrita resumidamente na subseção 2.4.1 e detalhada em [25]. A primeira grande restrição é que modelo de superfície deformável possui uma abordagem puramente mecânica, na qual é desprezada a parte térmica do modelo de Green et al. [25], simplificando a formulação do modelo. Além disso, este modelo baseia-se no caso especial descrito no final da subseção 2.4.1, em que a superfície de Cosserat tem um comportamento elástico e o vetor diretor coincide com o vetor normal unitário obtido pelo produto vetorial entre os vetores de base

$$\mathbf{n} = \mathbf{a}_3 = \frac{\frac{\partial \mathbf{r}}{\partial x^1} \times \frac{\partial \mathbf{r}}{\partial x^2}}{\left| \frac{\partial \mathbf{r}}{\partial x^1} \times \frac{\partial \mathbf{r}}{\partial x^2} \right|} = \frac{\mathbf{r}_{,\alpha} \times \mathbf{r}_{,\beta}}{|\mathbf{r}_{,\alpha} \times \mathbf{r}_{,\beta}|}. \quad (2.21)$$

O ponto crucial deste modelo diz respeito em como simplificar as componentes da força elástica $\mathbf{N}^\alpha|_\alpha$ na equação 2.12 sem sacrificar a flexibilidade na reprodução de diferentes comportamentos de dobras características de tecidos. Em essência, o problema consiste em, dadas as medidas de métrica e curvatura, determinar os deslocamentos dos pontos da superfície s ao longo do tempo de forma coerente. Como é conveniente expressar a derivada covariante $\mathbf{N}^\alpha|_\alpha$ de forma independente do sistema de coordenadas escolhido, usam-se as derivadas parciais ordinárias em relação a x^α e substitui-se as derivadas de segunda ordem pelas fórmulas de transformação com os símbolos de Christoffel $\Gamma_{\beta\alpha}^\alpha$

$$\begin{aligned} \mathbf{N}^\alpha|_\alpha &\equiv (\mathbf{N}_{,\alpha}^\alpha + \Gamma_{\beta\alpha}^\alpha \mathbf{N}^\beta) = (\mathbf{N}_{,\alpha}^\alpha + \Gamma_{\alpha\lambda}^\lambda \mathbf{N}^\alpha) \\ &\equiv \left[(N^{i\alpha} \mathbf{a}_i)_{,\alpha} + \Gamma_{\alpha\lambda}^\lambda N^{i\alpha} \mathbf{a}_i \right]. \end{aligned} \quad (2.22)$$

Os símbolos de Christoffel $\Gamma_{\beta\alpha}^\alpha$ são usados na geometria diferencial para o cálculo de transformações entre distintos sistemas de referência. Eles são associados ao sistema de coordenadas de superfície x^a e definidos pela métrica associada a esse sistema da forma

$$\Gamma_{\alpha\beta}^\lambda = \mathbf{r}_{,\alpha\beta} \cdot \mathbf{a}^\lambda. \quad (2.23)$$

Uma outra restrição feita por de Melo em [4] para obter um modelo de superfície deformável adequado para tecidos refere-se à espessura da superfície ser suficientemente pequena a ponto de ser considerada nula, caso em que a superfície recebe o nome de membrana na teoria de cascas. Desta forma, a força diretora \mathbf{L} pode ser considerada nula na equação 2.14, resultando em

$$N^{3\alpha} = M^{\alpha\beta}|_\beta. \quad (2.24)$$

Substituindo a equação 2.24 em 2.22, obtém-se a seguinte expressão para a força elástica

$$\begin{aligned} \mathbf{N}^\alpha|_\alpha &\equiv \left[(N^{\beta\alpha} \mathbf{a}_\beta)_{,\alpha} + \Gamma_{\alpha\lambda}^\lambda N^{\beta\alpha} \mathbf{a}_\beta \right] \\ &+ \left[(M^{\alpha\beta}|_\beta \mathbf{a}_3)_{,\alpha} + \Gamma_{\alpha\lambda}^\lambda M^{\alpha\beta}|_\beta \mathbf{a}_3 \right], \end{aligned} \quad (2.25)$$

na qual a derivada covariante do tensor de superfície $M^{\alpha\beta}$ pode ser expressa por

$$M^{\alpha\beta}|_\beta = (M^{\alpha\beta})_{,\beta} + (\Gamma_{\beta\rho}^\alpha M^{\rho\beta} + \Gamma_{\beta\rho}^\beta M^{\alpha\rho}). \quad (2.26)$$

Dessa forma, a solução da força elástica na equação 2.25 pode ser resolvida com o cálculo dos termos $N^{\beta\alpha}$ expressos pela equação 2.17, que relaciona $N^{*\beta\alpha}$ e $M^{\alpha\beta}$ definidos em 2.15 e 2.16, respectivamente. Como estes últimos termos estão relacionados com a determinação da energia interna \mathcal{A} , o problema agora recai sobre a simplificação da equação 2.19. Entretanto, é importante salientar que as componentes da força elástica $N^\alpha|_\alpha$ na equação 2.25 não estão prontamente computáveis.

Com alguns experimentos, de Melo [4] observou que a equação 2.20 pode ser simplificada sem perda de qualidade do efeito visual, assumindo que apenas os termos com índices $\rho = \alpha$ e $\lambda = \beta$, ou $\rho = \beta$ e $\lambda = \alpha$ são termos não-nulos, resultando em

$$\begin{aligned} \Phi^{\alpha\beta} &= \Phi^{\beta\alpha} = H^{\alpha\beta\alpha\beta} = H^{\beta\alpha\beta\alpha} \\ &= \beta_1 A^{\alpha\alpha} A^{\beta\beta} + 2\beta_2 (A^{\alpha\beta})^2 \\ \Psi^{\alpha\beta} &= \Psi^{\beta\alpha} = B^{\alpha\beta\alpha\beta} = B^{\beta\alpha\beta\alpha} \\ &= \beta_3 A^{\alpha\alpha} A^{\beta\beta} + 2\beta_4 (A^{\alpha\beta})^2 \\ \Theta^{\alpha\beta} &= \Theta^{\beta\alpha} = C^{\alpha\beta\alpha\beta} = C^{\beta\alpha\beta\alpha} \\ &= \beta_5 A^{\alpha\alpha} A^{\beta\beta} + 2\beta_6 (A^{\alpha\beta})^2. \end{aligned} \quad (2.27)$$

Considerando

$$\beta_1 = \beta_2 = \zeta_{\alpha\beta},$$

$$\beta_3 = \beta_4 = \xi_{\alpha\beta},$$

$$\beta_5 = \beta_6 = \phi_{\alpha\beta},$$

a equação 2.27 pode ser reescrita como

$$\begin{aligned} \Phi^{\alpha\beta} &= \Phi^{\beta\alpha} = \zeta_{\alpha\beta} (A^{\alpha\alpha} A^{\beta\beta} + 2(A^{\alpha\beta})^2) \\ \Psi^{\alpha\beta} &= \Psi^{\beta\alpha} = \xi_{\alpha\beta} (A^{\alpha\alpha} A^{\beta\beta} + 2(A^{\alpha\beta})^2) \\ \Theta^{\alpha\beta} &= \Theta^{\beta\alpha} = \phi_{\alpha\beta} (A^{\alpha\alpha} A^{\beta\beta} + 2(A^{\alpha\beta})^2). \end{aligned} \quad (2.28)$$

Os termos $\Phi^{\alpha\beta}$ e $\Psi^{\alpha\beta}$ afetam predominantemente os comportamentos da métrica e da curvatura, respectivamente, de uma superfície de Cosserat. Denomina-se $\zeta_{\alpha\beta}$ e $\xi_{\alpha\beta}$ como coeficientes elásticos e os parâmetros $\Phi^{\alpha\beta}$ e $\Psi^{\alpha\beta}$ como parâmetros de material. O termo $\Theta^{\alpha\beta}$, por sua vez, está relacionado com o número de ondulações que ocorrem na superfície deformada na direção para fora do plano quando sujeita às forças tangenciais, assim $\phi_{\alpha\beta}$ é chamado de fator de *buckling*. Substituindo a equação 2.28 em 2.15 e 2.16 obtêm-se $N^{*\alpha\beta}$ e $M^{\alpha\beta}$ em termos da variação da área da superfície $\frac{S_0}{S}$, das medidas físicas $\varepsilon_{\alpha\beta}$ e $\kappa_{\alpha\beta}$, dos parâmetros de material $\Phi^{\alpha\beta}$ e $\Psi^{\alpha\beta}$ e do termo relacionado ao *buckling* $\Theta^{\alpha\beta}$

$$\begin{aligned} N^{*\alpha\beta} &= \mu \frac{\partial \mathcal{A}}{\partial \varepsilon_{\alpha\beta}} = \frac{\mu}{\mu_0} (2\Phi^{\alpha\beta} \varepsilon_{\alpha\beta} + \Theta^{\alpha\beta} \kappa_{\alpha\beta}) \\ &= \frac{S_0}{S} (2\Phi^{\alpha\beta} \varepsilon_{\alpha\beta} + \Theta^{\alpha\beta} \kappa_{\alpha\beta}) \end{aligned} \quad (2.29)$$

$$\begin{aligned} M^{\alpha\beta} &= \mu \frac{\partial \mathcal{A}}{\partial \kappa_{i\alpha}} = \frac{\mu}{\mu_0} (2\Psi^{\alpha\beta} \varepsilon_{\alpha\beta} + \Theta^{\alpha\beta} \kappa_{\alpha\beta}) \\ &= \frac{S_0}{S} (2\Psi^{\alpha\beta} \varepsilon_{\alpha\beta} + \Theta^{\alpha\beta} \kappa_{\alpha\beta}). \end{aligned} \quad (2.30)$$

Finalmente, substituindo as equações 2.29 e 2.30 em 2.17 pode-se expressar a componente $N^{\beta\alpha}$ da força interna elástica em função de termos já conhecidos, como

$$N^{\alpha\beta} = \frac{S_0}{S} [(2\Phi^{\alpha\beta} \varepsilon_{\alpha\beta} + \Theta^{\alpha\beta} \kappa_{\alpha\beta}) - b_{\alpha}^{\alpha} (2\Psi^{\alpha\beta} \kappa_{\alpha\beta} + \Theta^{\alpha\beta} \varepsilon_{\alpha\beta})], \quad (2.31)$$

e também a componente $N^{3\alpha}$, usando 2.28 em 2.24, que resulta em

$$N^{3\alpha} = M^{\alpha\beta}|_{\beta} = \frac{S_0}{S} [(2\Psi^{\alpha\beta} \kappa_{\alpha\beta} + \Theta^{\alpha\beta} \varepsilon_{\alpha\beta})]|_{\beta}. \quad (2.32)$$

Resumidamente, de Melo [4] adota uma formulação de energia similar àquela usada na teoria de cascas finas, a qual é familiar à comunidade de computação gráfica. Pode-se dizer que a principal contribuição deste modelo deformável é o comportamento apropriado na formação de dobras em tecidos quando eles são submetidos à forças de compressão. Esta qualidade é obtida com a consideração do termo $\Theta_{\alpha\beta}$, relacionado ao fator de *buckling*, que faz com que o aspecto visual das simulações atinja um maior grau de realismo. Além disso, os parâmetros de material $\Phi_{\alpha\beta}$ e $\Psi_{\alpha\beta}$ proporcionam a definição de diferentes materiais de tecido atribuindo-se valores reais não-negativos a eles, de forma que a manipulação destes valores é feita de forma intuitiva, como por exemplo, uma calça *jeans* possui um valor maior de resistência de curvatura do que uma calça de algodão, que por sua vez possui maior resistência métrica do que uma feita de elastano.

2.5 Considerações Finais

Neste capítulo foi apresentado um panorama dos trabalhos relacionados com a simulação de tecidos, destacando neles sua abordagem de modelagem física e solução numérica. Na seção 2.1 os modelos de partículas foram discutidos, destacando vantagens, como a facilidade de implementação e a eficiência computacional para calcular as forças internas, e também desvantagens, como não conseguir representar de forma apropriada o comportamento peculiar dos tecidos, diminuindo características como realismo e precisão física. Um dos representantes dos modelos físicos baseados na mecânica de partículas são os modelos massa-mola, amplamente usados na simulação de tecidos. As integrações explícitas foram utilizadas em conjunto com os modelos de partículas principalmente na década de 90, perdendo espaço desde então para os métodos implícitos e semi-implícitos.

Já na seção 2.2 os modelos baseados na mecânica de contínuos foram apresentados, destacando sua formulação mais precisa, sob o ponto de vista físico, que consegue modelar tecidos de forma mais realista que os modelos baseados em partículas. Contudo, as forças internas são obtidas pelas variações de tensores e elementos da geometria diferencial na superfície deformável, fazendo com que os modelos desta abordagem necessitem de mais parâmetros e cálculos, apresentando uma menor eficiência e, devido a formulação mais complexa, uma menor facilidade de implementação quando comparados aos modelos de partículas. As integrações implícitas e semi-implícitas foram muito utilizadas em conjunto com estes modelos há mais de duas décadas, sendo esta combinação também o foco de alguns trabalhos recentes. Contudo, como se viu na seção 2.3, ainda não existe na literatura a abordagem proposta nesta dissertação, que é simular tecidos unindo uma integração explícita com um modelo de tecido baseado na mecânica de contínuos.

Também foi visto neste capítulo que o modelo físico proposto por de Melo [4], fundamentado na mecânica de contínuos, foi o escolhido para ser usado neste trabalho em adição a um método numérico explícito. A seção 2.4 descreveu de forma detalhada a formulação deste modelo de tecido, apresentando um resumo da teoria geral de superfície de Cosserat para depois expor as simplificações e restrições formuladas sobre ela. Com o intuito de sintetizar as qualidades que justificam sua escolha neste trabalho, as seguintes propriedades do modelo [4] foram decisivas:

1. Ele é adequado para analisar grandes deformações.
2. As condições de compatibilidade de Weingarten são satisfeitas, assegurando a regularidade da superfície em qualquer estado que o tecido possa assumir.
3. A analogia entre as quantidades estáticas (forças de contato e acoplamentos) e geométricas (esticamentos e curvamentos) pode ser facilmente estabelecida.

4. Os poucos parâmetros usados para definir um material possuem significados geométricos claros, tornando o controle do modelo mais fácil.

Com o modelo físico baseado na mecânica de contínuos devidamente selecionado, justificado e explicado, o próximo capítulo apresenta a discretização espacial utilizando a técnica de diferenças finitas para que então este modelo pode ser integrado no tempo usando métodos explícitos.

Capítulo 3

Soluções Numéricas

Este capítulo apresenta diferentes soluções numéricas explícitas que podem integrar ao longo do tempo o modelo de tecido proposto por de Melo [4]. Entretanto, antes de apresentar os métodos numéricos é necessário discretizar no espaço o modelo de tecido para transformar as suas equações diferenciais parciais em equações diferenciais ordinárias em função do tempo. Problemas de contorno na discretização espacial, como a indefinição de pontos necessários para o cálculo de elementos do modelo nas bordas da malha discreta, são solucionados com um esquema de diferenças finitas condicionado diferentemente para cada fronteira, que é apresentado na seção 3.1. Este passo é essencial para o modelo baseado na mecânica de contínuos possuir uma formulação similar àquela usada na mecânica de partículas, aproveitando algumas de suas vantagens de eficiência computacional.

Os principais métodos de integração são explicados na seção 3.2, destacando suas formulações, vantagens e desvantagens. No modelo de tecido usado neste trabalho não existe a definição de uma função que relacione as coordenadas da superfície com suas posições tridimensionais, sendo o modelo responsável por, dada uma amostra de superfície de Cosserat deformada, estimar as forças internas e externas existentes em cada ponto e determinar as acelerações resultantes. Desta forma, os métodos explícitos apresentados são adaptados para atender o propósito de produzir o movimento de tecidos, encontrando diferentes superfícies ao longo do tempo. Em seguida, este capítulo apresenta na seção 3.3 os trabalhos relacionados à simulação de tecidos que utilizam os métodos explícitos discutidos, com o intuito de destacar a eficácia desta abordagem em conjunto com um modelo contínuo, em relação às demais existentes atualmente. Para melhor preencher a lacuna destacada no capítulo 2, a seção 3.4 apresenta resultados comparativos entre as implementações dos diferentes métodos numéricos com o objetivo de investigar qual é a melhor opção em termos de precisão e custo computacional. Na seção 3.5 é feita uma análise dos resultados obtidos para justificar a escolha do método numérico que agrega de forma balanceada fatores como o realismo e eficiência aplicados ao modelo proposto por de Melo [4]. Por fim, a seção 3.6 apresenta considerações finais deste capítulo.

3.1 Discretizações Espaciais

A primeira preocupação na discretização espacial do modelo proposto por de Melo [4] é tornar o domínio contínuo em um conjunto discreto de nós bidimensionais. Dessa forma, a superfície é discretizada por uma malha retangular de $m \times n$ pontos, dispostos em duas direções, x^1 e x^2 , cujos espaçamentos entre nós consecutivos são Δx^1 e Δx^2 , respectivamente. A segunda preocupação é aproximar as derivadas de primeira e de segunda ordem, que é feito aqui com o método das diferenças finitas. Neste método, cada nó $[k, l]$, com $k = 1..m$ e $l = 1..n$, refere-se a um ponto discreto no domínio de acordo com a convenção da figura 3.1. Qualquer nó $[k, l]$ da malha bidimensional corresponde a um ponto P cuja posição tridimensional é dada pelo vetor posição \mathbf{r} , de forma que $\mathbf{r}[k, l] = (P_x, P_y, P_z)$, sendo P_i o valor da coordenada do ponto P no eixo i .

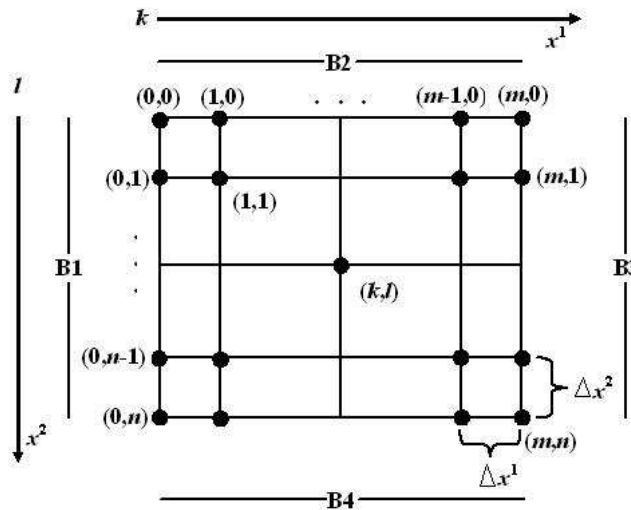


Figura 3.1: Malha discreta.

Considerando uma função qualquer f são definidos, em uma determinada ordem, operadores avançados, atrasados e centrais de acordo com direção em que os pontos vizinhos são utilizados para aproximar as derivadas desejadas [27]. Os operadores de diferença posterior (ou de diferença avançada) de primeira ordem podem ser expressos por

$$D_1^+(f[k, l]) = \frac{f[k+1, l] - f[k, l]}{\Delta x^1}, \quad (3.1)$$

$$D_2^+(f[k, l]) = \frac{f[k, l+1] - f[k, l]}{\Delta x^2}, \quad (3.2)$$

e os operadores de diferença anterior (ou diferença atrasada) de primeira ordem por

$$D_1^-(f[k, l]) = \frac{f[k, l] - f[k-1, l]}{\Delta x^1}, \quad (3.3)$$

$$D_2^-(f[k, l]) = \frac{f[k, l] - f[k, l-1]}{\Delta x^2}. \quad (3.4)$$

Para as derivadas parciais de segunda ordem utilizam-se operadores definidos pela combinação das equações anteriores, sendo os operadores de diferença cruzada posterior e anterior definidos por

$$D_{12}^+(f[k, l]) = D_{21}^+(f[k, l]) = \frac{f[k+1, l+1] - f[k+1, l] - f[k, l+1] + f[k, l]}{\Delta x^1 \Delta x^2}, \quad (3.5)$$

$$D_{12}^-(f[k, l]) = D_{21}^-(f[k, l]) = \frac{f[k-1, l-1] - f[k-1, l] - f[k, l-1] + f[k, l]}{\Delta x^1 \Delta x^2}, \quad (3.6)$$

e os operadores de diferença central de 2ª ordem por

$$D_{11}(f[k, l]) = \frac{f[k+1, l] - 2 * f[k, l] + f[k-1, l]}{\Delta x^1 \Delta x^1}, \quad (3.7)$$

$$D_{22}(f[k, l]) = \frac{f[k, l+1] - 2 * f[k, l] + f[k, l-1]}{\Delta x^2 \Delta x^2}. \quad (3.8)$$

Considerando a convenção de bordas ($B1, B2, B3$ e $B4$) e orientação dos índices ilustrada na figura 3.1, de Melo [4] considerou os operadores de primeira ordem avançados para definir os vetores de base \mathbf{a}_α de um ponto $[k, l]$ usando a função \mathbf{r} da seguinte forma

$$\mathbf{a}_\alpha[k, l] = D_\alpha^+(\mathbf{r}[k, l]). \quad (3.9)$$

Desta forma, para os pontos da malha discreta que estão nas bordas $B3$ e $B4$ as derivadas direcionais para frente na direção x^1 e x^2 , respectivamente, não existem, pois não há pontos sucessores. Neste caso, de Melo [4] considerou estas derivadas como nulas, ou seja,

$$\mathbf{a}_\alpha[k, l] = 0, \quad \text{para } (k, l) \in \{B3, B4\}. \quad (3.10)$$

A restrição da equação 3.10 implica em anular outros elementos do modelo deformável, como o tensor métrico, que tem seus coeficientes calculados por produtos escalares entre os vetores de base. A área associada a cada ponto, calculada pelo determinante da matriz formada com os coeficientes do tensor métrico, também é zerada nas mesmas bordas por consequência. Estas áreas anuladas afetam a área total da superfície, usada no cálculo da força interna do modelo. Além disso, a variação do tensor

métrico ao longo do tempo fornece medidas de deformação que também são usadas no cálculo das forças internas. Desta forma, as bordas $B1$ e $B2$ possuem valores corretos desses elementos enquanto $B3$ e $B4$ não, causando um desequilíbrio na superfície que pode resultar em problemas de estouro, em que pontos apresentam grande deslocamentos, como ilustrado por uma imagem de simulação na figura 3.2(a), ou na má formação de dobras, conforme a figura 3.2(b).

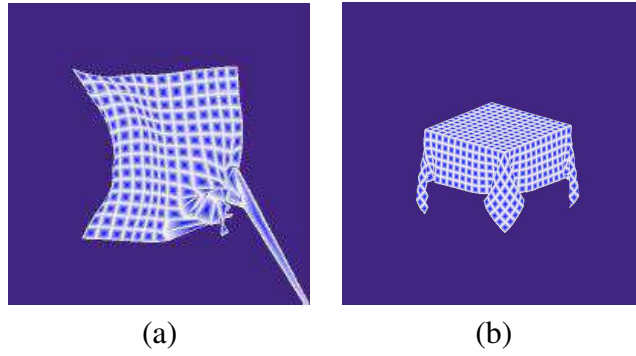


Figura 3.2: Problemas nas bordas.

A solução adotada visa não anular nenhuma derivada de primeira ou segunda ordem, pegando sempre pontos vizinhos na direção em que eles existem. Assim, para os vetores de base que utilizam o operador de primeira ordem para frente sobre a função $\mathbf{r}[k, l]$, os pontos das bordas $B3$ e $B4$ não possuem vizinhos nas direções positivas, mas possuem vizinhos em ambas as direções no sentido negativo, de forma que o operador de primeira ordem para trás, expressos pelas equações 3.3 e 3.4, podem ser usados nestas bordas. Esta nova consideração é formulada como

$$\mathbf{a}_\alpha[k, l] = D_\alpha^-(\mathbf{r}[k, l]), \text{ para } (k, l) \in \{B3, B4\}. \quad (3.11)$$

Prosseguindo com a discretização espacial dos demais elementos do modelo, o vetor normal, que em conjunto com os vetores de base formam o referencial local a cada ponto da superfície do tecido, são calculados utilizando todas as faces adjacentes existentes em cada ponto. Assim, para cada ponto da malha discreta o vetor normal é aproximado pela média normalizada dos vetores normais, estes também normalizados, das quatro (no máximo) faces adjacentes em cada ponto

$$\begin{aligned} \mathbf{a}_3^0[k, l] &= \frac{(\mathbf{r}[k+1, l] - \mathbf{r}[k, l]) \times (\mathbf{r}[k, l+1] - \mathbf{r}[k, l])}{|(\mathbf{r}[k+1, l] - \mathbf{r}[k, l]) \times (\mathbf{r}[k, l+1] - \mathbf{r}[k, l])|}, \\ \mathbf{a}_3^1[k, l] &= \frac{(\mathbf{r}[k, l+1] - \mathbf{r}[k, l]) \times (\mathbf{r}[k-1, l] - \mathbf{r}[k, l])}{|(\mathbf{r}[k, l+1] - \mathbf{r}[k, l]) \times (\mathbf{r}[k-1, l] - \mathbf{r}[k, l])|}, \\ \mathbf{a}_3^2[k, l] &= \frac{(\mathbf{r}[k-1, l] - \mathbf{r}[k, l]) \times (\mathbf{r}[k, l-1] - \mathbf{r}[k, l])}{|(\mathbf{r}[k-1, l] - \mathbf{r}[k, l]) \times (\mathbf{r}[k, l-1] - \mathbf{r}[k, l])|}, \end{aligned}$$

$$\mathbf{a}_3^3[k, l] = \frac{(\mathbf{r}[k, l-1] - \mathbf{r}[k, l]) \times (\mathbf{r}[k+1, l] - \mathbf{r}[k, l])}{|(\mathbf{r}[k, l-1] - \mathbf{r}[k, l]) \times (\mathbf{r}[k+1, l] - \mathbf{r}[k, l])|},$$

de forma que média do vetor normal resultante, em cada ponto, pode ser expressa por

$$\mathbf{a}_3[k, l] = \frac{\sum_{i=0}^3 \mathbf{a}_3^i}{|\sum_{i=0}^3 \mathbf{a}_3^i|}. \quad (3.12)$$

Assim, para os pontos de todas as bordas em que não existem as quatro faces adjacentes, a equação 3.12 é utilizada apenas com as faces existentes, de forma que os quatro cantos da malha retangular possuem apenas uma face adjacente e os demais pontos das bordas, duas.

Partindo para a discretização espacial dos elementos de segunda ordem, que dependem do cálculo de uma derivada parcial segunda, como o tensor de curvatura e os símbolos de Christoffel, de Melo [4] utilizou os operadores avançados de segunda ordem para $\alpha \neq \beta$, e os operadores centrais para $\alpha = \beta$, definindo o tensor de curvatura por

$$\begin{aligned} b_{\alpha\beta}[k, l] &= \mathbf{a}_3 \cdot D_{\alpha\beta}^+(\mathbf{r}[k, l]), \quad \text{se } \alpha \neq \beta, \text{ e} \\ b_{\alpha\beta}[k, l] &= \mathbf{a}_3 \cdot D_{\alpha\beta}(\mathbf{r}[k, l]), \quad \text{se } \alpha = \beta. \end{aligned} \quad (3.13)$$

Novamente, para as derivadas de segunda ordem existirá o problema de indefinição de pontos vizinhos nas bordas, mais especificamente nas bordas $B3$ e $B4$ para os operadores cruzados avançados e nas quatro bordas para o operador central. Nestes casos, assim como a derivada de primeira ordem, o resultado da derivação é considerado nulo, ou seja

$$D_{\alpha\beta}^+(\mathbf{r}[k, l]) = 0, \quad \text{para } (k, l) \in \{B3, B4\}, \text{ e} \quad (3.14)$$

$$D_{\alpha\beta}(\mathbf{r}[k, l]) = 0, \quad \text{para } (k, l) \in \{B1, B2, B3, B4\}. \quad (3.15)$$

Em relação à derivada segunda da função posição $\mathbf{r}[k, l]$, a solução tem que ser dividida conforme as coordenadas na qual a função está sendo derivada. Para índices cruzados, ou seja, $\alpha \neq \beta$, é usado o operador de segunda ordem para frente expresso na equação 3.5, de forma que o problema reside novamente nas bordas $B3$ e $B4$, conforme a equação 3.14. Convencionando-se C_i , com $i = 1..4$, como os cantos entre as bordas da malha discreta, sendo o canto C_4 o que fica entre as bordas $B4$ e $B1$, o canto C_1 entre as bordas $B1$ e $B2$ e assim sucessivamente, a solução pode ser dividida em quatro situações que necessitam de diferentes direções para a obtenção da derivada:

- Para o ponto $p = (k, l) = C3$, a derivada é expressa por 3.6, ou seja,

$$\mathbf{r}_{,12}(k, l) = \mathbf{r}_{,21}(k, l) = D_{12}^-(\mathbf{r}[k, l]) = D_{21}^-(\mathbf{r}[k, l]). \quad (3.16)$$

- Para o ponto $p = (k, l) = C4$, a derivada é expressa por

$$\mathbf{r}_{,12}(k, l) = \mathbf{r}_{,21}(k, l) = \frac{(\mathbf{r}[k+1, l-1] + \mathbf{r}[k+1, l] + \mathbf{r}[k, l-1] + \mathbf{r}[k, l])}{\Delta x^1 \Delta x^2}. \quad (3.17)$$

- Para os pontos $p = (k, l) \in \{B3 - C3\}$

$$\mathbf{r}_{,12}(k, l) = \mathbf{r}_{,21}(k, l) = \frac{(\mathbf{r}[k-1, l+1] + \mathbf{r}[k-1, l] + \mathbf{r}[k, l+1] + \mathbf{r}[k, l])}{\Delta x^1 \Delta x^2}. \quad (3.18)$$

- Para pontos $p = (k, l) \in \{B4 - C3 - C4\}$

$$\mathbf{r}_{,12}(k, l) = \mathbf{r}_{,21}(k, l) = \frac{(\mathbf{r}[k+1, l-1] + \mathbf{r}[k+1, l] + \mathbf{r}[k, l-1] + \mathbf{r}[k, l])}{\Delta x^1 \Delta x^2}. \quad (3.19)$$

Desta forma, o objetivo é sempre usar os pontos vizinhos existentes para que nenhum ponto da malha possua uma derivada de primeira ou segunda ordem nula, evitando o desequilíbrio de forças internas. Em casos como nos pontos $p \in \{B3 - C2 - C3\}$ ou $p \in \{B3 - C3 - C4\}$, é possível usar também o operador de segunda ordem cruzado atrasado, expresso na equação 3.6 e utilizado em $C3$, entretanto, optou-se por usar as direções convencionais nos casos em que existem mais de uma opção. Quanto a derivada segunda para índices centrais, para os quais de Melo [4] utilizou os operadores das equações 3.7 e 3.8, o problema ocorre em todas as quatro bordas, pois os operadores centrais pegam um ponto vizinho de cada uma das duas direções, de forma que sempre em uma borda não existirá uma direção. Como solução, o problema é solucionado inicialmente para os quatro cantos da malha discreta da seguinte forma:

- Para o ponto $p = (k, l) = C1$ a derivada é expressa por 3.5,

$$\mathbf{r}_{,11}(k, l) = \mathbf{r}_{,22}(k, l) = D_{12}^+(\mathbf{r}[k, l]) = D_{21}^+(\mathbf{r}[k, l]). \quad (3.20)$$

- Para o ponto $p = (k, l) = C2$ a derivada é expressa por

$$\mathbf{r}_{,11}(k, l) = \mathbf{r}_{,22}(k, l) = \frac{(\mathbf{r}[k-1, l+1] + \mathbf{r}[k-1, l] + \mathbf{r}[k, l+1] + \mathbf{r}[k, l])}{\Delta x^1 \Delta x^2}. \quad (3.21)$$

- Para o ponto $p = (k, l) = C3$ a derivada é expressa por 3.6,

$$\mathbf{r}_{,11}(k, l) = \mathbf{r}_{,22}(k, l) = D_{12}^-(\mathbf{r}[k, l]) = D_{21}^-(\mathbf{r}[k, l]). \quad (3.22)$$

- Para o ponto $p = (k, l) = C4$ a derivada é expressa por

$$\mathbf{r}_{,11}(k, l) = \mathbf{r}_{,22}(k, l) = \frac{(\mathbf{r}[k+1, l-1] + \mathbf{r}[k+1, l] + \mathbf{r}[k, l-1] + \mathbf{r}[k, l])}{\Delta x^1 \Delta x^2}. \quad (3.23)$$

Assim, os quatro pontos dos cantos possuem suas derivadas de segunda ordem, em relação aos índices iguais, aproximadas por operadores de diferenças finitas cruzados, pois os operadores centrais necessitam de um ponto adjacente inexistente. Nestes pontos dos cantos os índices 11 e 22 utilizam as mesmas soluções de índices cruzados, pois considerando uma vizinhança de no máximo oito pontos imediatamente vizinhos, eles possuem apenas três pontos adjacentes. Portanto, as derivadas de segunda ordem dos pontos localizados nos quatro cantos da malha discreta possuem o mesmo cálculo para qualquer $\alpha\beta$. Quanto às derivadas de segunda ordem de índices iguais para o restante dos pontos das bordas, excluindo-se os cantos, pode-se dividir a solução conforme as situações:

- Para os pontos $p = (k, l) \in \{B1 - C1 - C4\}$ derivados em relação à x^1

$$\mathbf{r}_{,11}(k, l) = D_{12}^+(\mathbf{r}[k, l]). \quad (3.24)$$

- Para os pontos $p = (k, l) \in \{B1 - C1 - C4\}$ derivados em relação à x^2

$$\mathbf{r}_{,22}(k, l) = D_{22}(\mathbf{r}[k, l]). \quad (3.25)$$

- Para os pontos $p = (k, l) \in \{B2 - C2 - C1\}$ derivados em relação à x^1

$$\mathbf{r}_{,11}(k, l) = D_{11}(\mathbf{r}[k, l]). \quad (3.26)$$

- Para os pontos $p = (k, l) \in \{B2 - C2 - C1\}$ derivados em relação à x^2

$$\mathbf{r}_{,22}(k, l) = D_{12}^+(\mathbf{r}[k, l]). \quad (3.27)$$

- Para os pontos $p = (k, l) \in \{B3 - C3 - C2\}$ derivados em relação à x^1

$$\mathbf{r}_{,11}(k, l) = \frac{(\mathbf{r}[k-1, l+1] + \mathbf{r}[k-1, l] + \mathbf{r}[k, l+1] + \mathbf{r}[k, l])}{\Delta x^1 \Delta x^2}. \quad (3.28)$$

- Para os pontos $p = (k, l) \in \{B3 - C3 - C2\}$ derivados em relação à x^2

$$\mathbf{r}_{,22}(k, l) = D_{22}(\mathbf{r}[k, l]). \quad (3.29)$$

- Para os pontos $p = (k, l) \in \{B4 - C4 - C3\}$ derivados em relação à x^1

$$\mathbf{r}_{,11}(k, l) = D_{11}(\mathbf{r}[k, l]). \quad (3.30)$$

- Para os pontos $p = (k, l) \in \{B4 - C4 - C3\}$ derivados em relação à x^2

$$r_{,22}(k, l) = \frac{(\mathbf{r}[k+1, l-1] + \mathbf{r}[k+1, l] + \mathbf{r}[k, l-1] + \mathbf{r}[k, l])}{\Delta x^1 \Delta x^2}. \quad (3.31)$$

Desta forma, é dada a preferência pelas direções padrões quando existirem duas opções. Esta discretização espacial se baseia em uma vizinhança 1-anel, por isso se aproxima a derivada segunda de um canto, como $C1$, por exemplo, com operadores cruzados ao invés de usar operadores avançados para índices centrais, de forma a utilizar os pontos $[k, l]$, $[k+1, l]$ e $[k+2, l]$, sendo este último ponto pertencente a vizinhança 2-anel. Apesar dos exemplos desta subseção estarem focados nas derivações da função de posição $\mathbf{r}[k, l]$, todas as demais são calculadas da mesma forma. A figura 3.3 resume as diferentes considerações para as bordas.

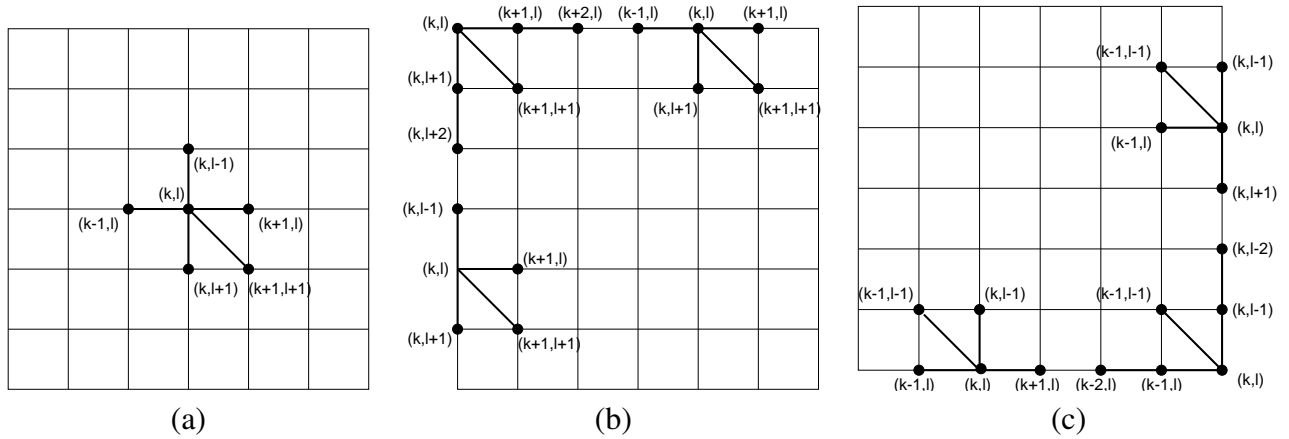


Figura 3.3: Derivadas aproximadas com diferenças finitas: (a) para pontos internos; (b) para pontos nas bordas B1 e B2; e (c) para pontos nas bordas B3 e B4.

Com estas considerações para as derivadas de segunda ordem dos elementos do modelo, os símbolos de Christoffel, que usam a derivada de segunda ordem do vetor posição \mathbf{r} em relação as duas

curvas coordenadas x^1 e x^2 , podem ser expressos por

$$\Gamma_{\alpha\beta}^\lambda[k, l] = \mathbf{r}_{,\alpha\beta}[k, l] \cdot \mathbf{a}^\lambda. \quad (3.32)$$

Estes últimos elementos são usados na definição da derivada covariante $M^{\alpha\beta}|_\beta$, diferindo das demais derivadas apresentadas até aqui. Com a consideração de que os componentes da força diretora $M^{\alpha\beta}$ sejam dados pela função discreta $m_{\alpha\beta}[k, l]$, sua derivada covariante em relação à x^α é expressa por

$$M^{\alpha\beta}|_\beta[k, l] = D_\alpha^-(m_{\alpha\beta}[k, l]) + (\Gamma_{\alpha\beta}^\lambda M^{\alpha\beta} + \Gamma_{\alpha\beta}^\lambda M^{\alpha\beta}). \quad (3.33)$$

Por fim, para a aproximação da derivada parcial de termos mistos, como o termo $(N^{\beta\alpha}\mathbf{a}_\beta)_{,\alpha}$ da equação 2.25, o seguinte desenvolvimento é usado

$$\begin{aligned} D_1^-(N^{\beta 1}\mathbf{a}_\beta)[k, l] &= \frac{N^{\beta 1}[k, l]}{\Delta x^1} \mathbf{a}_\beta[k, l] - \frac{N^{\beta 1}[k-1, l]}{\Delta x^1} \mathbf{a}_\beta[k-1, l], \\ D_2^-(N^{\beta 2}\mathbf{a}_\beta)[k, l] &= \frac{N^{\beta 2}[k, l]}{\Delta x^2} \mathbf{a}_\beta[k, l] - \frac{N^{\beta 2}[k, l-1]}{\Delta x^2} \mathbf{a}_\beta[k, l-1]. \end{aligned} \quad (3.34)$$

Como exemplo, substituindo a equação 3.9 em 3.34 e considerando $\beta = 1$, obtêm-se

$$\begin{aligned} D_1^-(N^{11}\mathbf{a}_1)[k, l] &= \frac{N^{11}[k, l]}{\Delta x^1} \frac{(\mathbf{r}[k+1, l] - \mathbf{r}[k, l])}{\Delta x^1} - \frac{N^{11}[k-1, l]}{\Delta x^1} \frac{(\mathbf{r}[k-1, l] - \mathbf{r}[k-2, l])}{\Delta x^1}, \\ D_2^-(N^{12}\mathbf{a}_1)[k, l] &= \frac{N^{12}[k, l]}{\Delta x^2} \frac{(\mathbf{r}[k+1, l] - \mathbf{r}[k, l])}{\Delta x^1} - \frac{N^{12}[k, l-1]}{\Delta x^2} \frac{(\mathbf{r}[k+1, l-1] - \mathbf{r}[k, l-1])}{\Delta x^1}. \end{aligned} \quad (3.35)$$

3.2 Métodos Explícitos de Integração

Existem diversos métodos explícitos de integração numérica que podem ser de passos e ordens diferentes, variando características como estabilidade e precisão numérica. Estes métodos são muito utilizados na matemática para encontrar a solução numérica de equações diferenciais. Geralmente, a solução visa encontrar o valor de uma função em um ponto sem saber a definição da função, porém sabendo a definição de uma ou mais de suas derivadas e também um valor inicial da função em um ponto suficientemente próximo ao que se deseja encontrar. Em outras palavras, considerando uma função $x(t)$, cuja definição é desconhecida, mas sabendo a definição de $x'(t) = f(t, x(t))$ e também do valor inicial $x(t_0)$, deseja-se obter a solução da função em $x(t_1)$, sendo t_0 e t_1 separados por Δt , chamado de comprimento de passo. Este problema pode ser solucionado com a expansão em série de

Taylor, representada por

$$x(t_0 + \Delta t) = x(t_0) + \Delta t x'(t_0) + \frac{\Delta t^2}{2!} x''(t_0) + \dots + \frac{\Delta t^n}{n!} \frac{\partial^n x}{\partial t^n} + \dots \quad (3.36)$$

A equação 3.36 é a base de vários métodos explícitos, os quais truncam e manipulam seus termos para obter diferentes precisões numéricas, conforme é mostrado nas subseções desta seção. Neste trabalho, estes métodos são usados para encontrar as novas geometrias da superfície do tecido ao longo do tempo. O objetivo é, a partir de um instante inicial t_0 , encontrar as novas posições $\mathbf{r}(t + \Delta t)$, no instante $t + \Delta t$, com base nos dados do instante anterior t . Desta forma, é gerado um erro por aproximar os pontos da superfície no tempo atual utilizando os dados correspondentes ao tempo anterior, sendo este erro associado ao tamanho do incremento Δt , chamado neste contexto de passo de tempo. A principal limitação dos métodos explícitos é a imposição de se usar um valor pequeno de passo de tempo para garantir a suavidade na mudança das forças que atuam sobre a superfície. Entretanto, eles são simples de serem implementados e possuem boa eficiência computacional.

As subseções seguintes apresentam os principais métodos explícitos que podem ser usados para integrar ao longo do tempo o modelo de tecido proposto por de Melo [4]. Em cada subseção, primeiro é apresentada a teoria do método, como sua formulação e precisão numérica, para então mostrar de que forma a teoria é adaptada para encontrar as novas posições da superfície do tecido em cada instante de tempo. Neste modelo deformável não existe a definição de uma função que relacione as coordenadas da malha discreta com as posições da superfície deformada no espaço contínuo, contudo, para cada estado diferente $\mathbf{r}(t)$ da superfície é possível obter a aceleração resultante $\mathbf{a}_{k,l}(t)$ de cada ponto (k, l) . A aceleração é encontrada isolando o termo $\frac{\partial^2 \mathbf{r}}{\partial t^2}$ a partir da equação 1.1

$$\mu \frac{\partial^2 \mathbf{r}}{\partial t^2} = \mathbf{f}(\mathbf{r}, t) - \varrho \frac{\partial \mathbf{r}}{\partial t} - \frac{\delta \varepsilon(\mathbf{r}, t)}{\delta \mathbf{r}},$$

que considerando a aceleração de cada ponto em um instante de tempo t como $\mathbf{a}_{k,l}(t) = \frac{\partial^2 \mathbf{r}}{\partial t^2}$, a velocidade $\mathbf{v}_{k,l}(t) = \frac{\partial \mathbf{r}}{\partial t}$, o termo $\frac{\delta \varepsilon(\mathbf{r}, t)}{\delta \mathbf{r}} = \mathbf{f}_{k,l}(t)$ correspondente às forças internas e $\mathbf{f}(\mathbf{r}, t) = \mathbf{f}_{e,k,l}(t)$ às forças externas, pode ser reescrita como

$$\mathbf{a}_{k,l}(t) = \frac{\mathbf{f}_{e,k,l}(t) - \varrho \mathbf{v}_{k,l}(t) - \mathbf{f}_{i,k,l}(t)}{\mu}. \quad (3.37)$$

Desta forma, todos os métodos explícitos apresentados nesta seção utilizam a equação 3.37, a qual dada a posição, velocidade, forças externas e internas de cada um dos pontos de uma superfície de Cosserat deformada, resulta na sua aceleração resultante daquele instante de tempo. A partir desta informação, os métodos explícitos devem encontrar a nova geometria da superfície $\mathbf{r}(t + \Delta t)$ em um

processo iterativo até atingir determinado tempo ou número de iterações.

3.2.1 Método de Euler

No campo da matemática, física e computação, o método explícito de Euler, também chamado de método de Euler para frente, é o procedimento numérico mais simples e básico usado para encontrar a solução de equações ordinárias diferenciais com um dado valor inicial. Este método explícito usa somente os dois primeiros termos da expansão de Taylor, expressa na equação 3.36, gerando um erro de truncamento de segunda ordem

$$x(t_0 + \Delta t) = x(t_0) + \Delta t x'(t_0) + O(\Delta t^2).$$

Considerando a notação apresentada em [28], em que a derivada de primeira ordem da função $y(t)$ é definida por uma dada função de incremento $y'(t) = f(t, y(t))$, um comprimento de passo h e uma solução inicial $y(t_0) = y_0$, pode-se reescrever a fórmula do método de Euler como

$$y_{n+1} = y_n + hf(t_n, y_n) + O(h^2), \quad (3.38)$$

a qual avança a solução de t_n para $t_{n+1} \equiv t_n + h$. O termo $O(h^2)$ corresponde ao erro de truncamento por desprezar as demais parcelas de ordens superiores na expansão de Taylor da equação 3.36.

Como o método explícito de Euler possui um erro de truncamento de segunda ordem, diz-se que ele é um método de precisão numérica de primeira ordem, sendo um método pouco preciso, embora seja simples de ser implementado e um dos mais eficientes computacionalmente. Com base na aceleração $\mathbf{a}_{k,l}(t)$ obtida com a equação 3.37, cada iteração deste método corresponde a encontrar a nova posição e velocidade de cada ponto da superfície do tecido no instante $t + \Delta t$. Assim, considerando (k, l) como os índices da malha bidimensional discreta que forma o tecido, o procedimento numérico pode ser descrito pelos seguintes passos:

1. Cálculo da aceleração resultante $\mathbf{a}_{k,l}(t)$ com a equação 3.37,
2. Atualização das posições

$$\mathbf{r}_{k,l}(t + \Delta t) = \mathbf{r}_{k,l}(t) + \mathbf{v}_{k,l}(t)\Delta t, \quad (3.39)$$

3. Atualização das velocidades

$$\mathbf{v}_{k,l}(t + \Delta t) = \mathbf{v}_{k,l}(t) + \mathbf{a}_{k,l}(t)\Delta t. \quad (3.40)$$

Desta forma, em qualquer instante de tempo t , primeiro a aceleração de cada ponto da superfície é encontrada com a equação 3.37, para a nova posição em $t + \Delta t$ ser atualizada com a velocidade do instante t e então atualizar a velocidade do próximo instante $t + \Delta t$. Este procedimento obtém em cada iteração uma nova geometria da superfície do tecido determinando as posições e velocidades em $t + \Delta t$ com base na aceleração do instante t .

Contudo, nessa formulação original do método explícito de Euler para frente as posições dos pontos são computadas antes das velocidades serem atualizadas com a aceleração resultante encontrada no primeiro passo do procedimento. A aceleração da equação 3.37 é considerada referente ao instante t devido ao fato das forças do lado direito da equação serem relativas ao mesmo instante. Entretanto, Provot considerou esta aceleração como relativa ao instante $t + \Delta t$ e usou uma fórmula física mais simples ($\mathbf{F} = m\mathbf{a}$) para encontrar a aceleração resultante de cada ponto, utilizando o método explícito de Euler com uma pequena variação em relação ao procedimento apresentado: ao invés de calcular primeiro a nova posição com a equação 3.39 e depois a nova velocidade com a equação 3.40, ele inverte estes dois passos, sendo o último passo, relacionado a atualização das posições, calculado usando a nova velocidade em $t + \Delta t$ conforme o procedimento [2]:

1. Cálculo da aceleração resultante $\mathbf{a}_{k,l}(t + \Delta t)$ com a equação 3.37,

2. Atualização das velocidades

$$\mathbf{v}_{k,l}(t + \Delta t) = \mathbf{v}_{k,l}(t) + \mathbf{a}_{k,l}(t + \Delta t)\Delta t, \quad (3.41)$$

3. Atualização das posições

$$\mathbf{r}_{k,l}(t + \Delta t) = \mathbf{r}_{k,l}(t) + \mathbf{v}_{k,l}(t + \Delta t)\Delta t. \quad (3.42)$$

Existe também quem considere esse esquema usado por Provot como semi-implícito [12], pois nele primeiro estimam-se as velocidades do próximo instante $t + \Delta t$ para depois calcular as novas posições dos pontos em $t + \Delta t$ já usando as novas velocidades estimadas, ou seja, a atualização das posições é feita com dados do instante anterior (termo $\mathbf{r}_{k,l}(t)$) e do atual (termo $\mathbf{v}_{k,l}(t + \Delta t)$). A figura 3.4 ilustra graficamente a diferença entre o método explícito de Euler e o usado por Provot [2], considerando a curva $\mathbf{r}(t)$ correspondente a posição de um ponto qualquer da superfície ao longo do tempo.

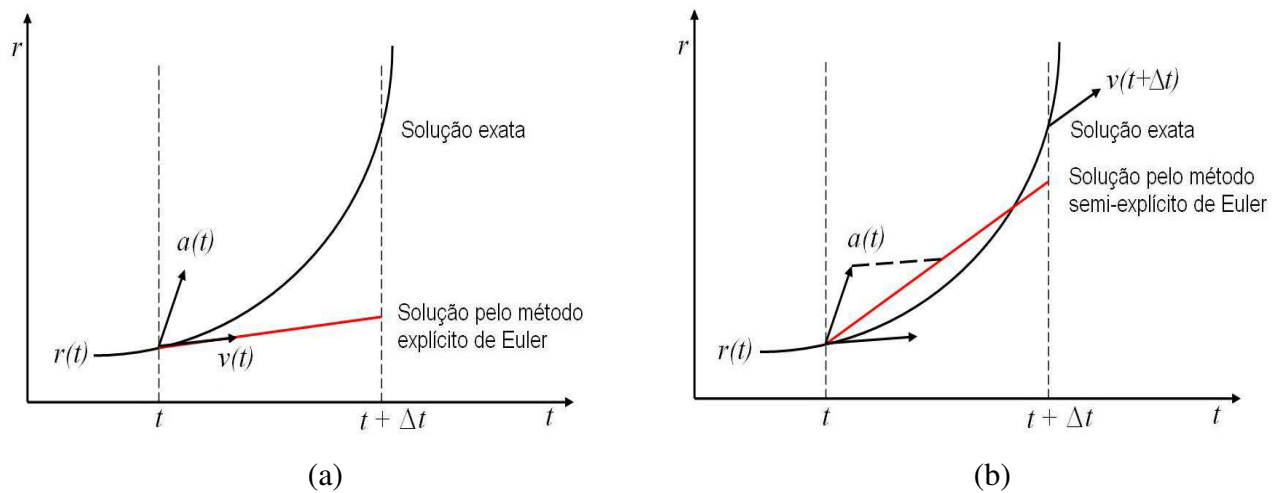


Figura 3.4: Gráficos do (a) método explícito de Euler e (b) método semi-implícito de Euler usado por Provot [2] em 1995.

3.2.2 Método do Ponto Intermediário

O método do ponto intermediário é uma técnica de previsão e correção que equivale a um método de Euler melhorado. Também conhecido como método de Runge-Kutta de segunda ordem, é um método que utiliza um passo de Euler para encontrar a solução em um ponto intermediário entre dois passos subsequentes e a usa para chegar a solução do passo final, em cada iteração. Em outras palavras, este método estima a solução em um instante $t + \frac{\Delta t}{2}$, para então avançar do instante t até $t + \Delta t$ com esta estimativa. Este método também pode ser relacionado com a expansão em série de Taylor, expressa na equação 3.36, em que os três primeiros termos do lado direito são considerados, sendo o segundo e o terceiro agregados fazendo-se uma nova estimativa da primeira derivada em um ponto intermediário de forma que

$$x(t_0 + \Delta t) = x(t_0) + \Delta t f \left(t + \frac{\Delta t}{2}, x(t_0) + \frac{\Delta t}{2} f(t_0, x(t_0)) \right) + O(\Delta t^3).$$

Usando a notação apresentada em [28], o método do ponto intermediário pode ser descrito com dois passos, conforme o procedimento

$$\begin{aligned} k_1 &= h f(t_n, y_n), \\ k_2 &= h f\left(t_n + \frac{h}{2}, y_n + \frac{k_1}{2}\right), \\ y_{n+1} &= y_n + k_2 + O(h^3). \end{aligned} \tag{3.43}$$

Desta forma, o método do ponto intermediário contém o termo $O(h^3)$, que representa um erro de

truncamento de terceira ordem e, portanto, é dito ter precisão numérica de segunda ordem. Embora seja uma ordem mais precisa que o método explícito de Euler, ele demanda mais cálculos por iteração.

A adaptação do método do ponto intermediário para integrar o modelo de tecido ao longo do tempo é feita de forma similar ao método explícito de Euler. Contudo, ao contrário do procedimento apresentado para o método de Euler, o primeiro passo não consiste em calcular a aceleração de todos os pontos com a equação 3.37, mas estimar a geometria da superfície em um instante intermediário $t + \frac{\Delta t}{2}$ a cada dois passos subseqüentes t e $t + \Delta t$ e, com ela, obter a aceleração correspondente a este instante usando o modelo de tecido. Esta nova aceleração serve para estimar a velocidade do ponto intermediário, que é usada para encontrar as posições finais, conforme o seguinte procedimento:

1. Armazenar $\mathbf{r}(t)$ e $\mathbf{v}(t)$ e encontrar $\mathbf{r}(t + \frac{\Delta t}{2})$ usando o método de Euler

$$\begin{aligned} k_1 &= \Delta t \mathbf{v}_{k,l}(t), \\ \mathbf{r}_{k,l}(t + \frac{\Delta t}{2}) &= \mathbf{r}_{k,l}(t) + \frac{k_1}{2}, \end{aligned}$$

2. Calcular uma iteração do modelo de tecido para determinar as novas acelerações de $\mathbf{r}(t + \frac{\Delta t}{2})$ com a equação 3.37

$$\mathbf{a}_{k,l}(t + \frac{\Delta t}{2}) = \frac{\mathbf{f}_{e,k,l}(t + \frac{\Delta t}{2}) - \varrho \mathbf{v}_{k,l}(t + \frac{\Delta t}{2}) - \mathbf{f}_{i,k,l}(t + \frac{\Delta t}{2})}{\mu},$$

3. Determinar a velocidade do ponto intermediário $t + \frac{\Delta t}{2}$ usando a aceleração $\mathbf{a}_{k,l}(t + \frac{\Delta t}{2})$ e calcular k_2

$$\begin{aligned} \mathbf{v}_{k,l}(t + \frac{\Delta t}{2}) &= \mathbf{v}_{k,l}(t) + \frac{\Delta t}{2} \mathbf{a}_{k,l}(t + \frac{\Delta t}{2}), \\ k_2 &= \Delta t \mathbf{v}_{k,l}(t + \frac{\Delta t}{2}), \end{aligned}$$

4. Atualizar a posição final usando a nova velocidade do ponto intermediário para avançar de $\mathbf{r}(t)$ para $\mathbf{r}(t + \Delta t)$

$$\mathbf{r}_{k,l}(t + \Delta t) = \mathbf{r}_{k,l}(t) + k_2, \quad (3.44)$$

5. Atualizar a velocidade final com a velocidade do ponto intermediário

$$\mathbf{v}_{k,l}(t + \Delta t) = \mathbf{v}_{k,l}(t) + \Delta t \mathbf{v}_{k,l}(t + \frac{\Delta t}{2}). \quad (3.45)$$

Desta forma, as posições e velocidades de cada ponto da superfície do tecido são atualizadas em uma iteração desse procedimento fazendo o uso de novos cálculos dos elementos do modelo de tecido. Isto é feito porque neste método é preciso estimar a velocidade em um ponto intermediário sem existir a definição de uma função que relacione diretamente esta estimativa. Assim, por não existir uma definição das funções derivadas, uma nova geometria da superfície é estimada em $t + \frac{\Delta t}{2}$ somente para que se possam obter as acelerações deste instante e com elas atualizar as velocidades, que são então usadas para levar a geometria original $r(t)$ para $r(t + \Delta t)$. Por fim, as velocidades em $t + \Delta t$ são atualizadas com as velocidades estimadas no ponto intermediário. O gráfico da figura 3.5 ilustra a aproximação que é feita usando o método do ponto intermediário sobre a curva $r(t)$, que representa a geometria de uma superfície ao longo do tempo, em comparação com o método explícito de Euler.

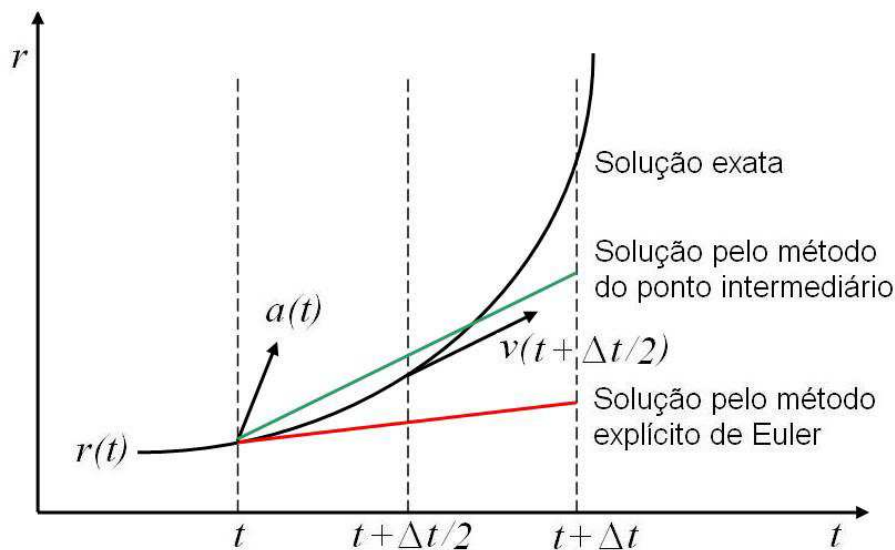


Figura 3.5: Gráfico comparativo entre os métodos de Euler e ponto intermediário.

Pode-se dizer que o método do ponto intermediário é mais preciso que o método explícito de Euler, porém, como ele possui mais cálculos e, no caso específico desta adaptação para o modelo de tecido proposto por de Melo [4], a necessidade de uma iteração adicional do modelo, a cada solução da geometria da superfície ao longo do tempo, tem um custo computacional significativo. Além disso, de forma similar ao método de Euler usado por Provot [2], o método do ponto intermediário também pode ser considerado como um esquema semi-implícito, pois ele ao prever e depois corrigir as soluções utiliza dados de $t + \frac{\Delta t}{2}$ para avançar de $r(t)$ para $r(t + \Delta t)$. A diferença para o esquema semi-implícito de Provot [2] é que este usa a velocidade em $t + \Delta t$ para atualizar $r(t + \Delta t)$, enquanto no método do ponto intermediário é usada a velocidade em $t + \frac{\Delta t}{2}$.

3.2.3 Método de Runge-Kutta

O método de Runge-Kutta é baseado na idéia de estimar valores intermediários dentro do intervalo existente entre dois passos subseqüentes, usando estas estimativas para melhorar a precisão da solução numérica entre os passo atual e o seguinte. Existem diferentes versões do método de Runge-Kutta que são divididas de acordo com a ordem de precisão numérica, sendo o método do ponto intermediário equivalente ao método de Runge-Kutta de segunda ordem. A versão mais utilizada do método de Runge-Kutta é a de quarta ordem, que embora demande muitos cálculos, possui uma precisão numérica de quarta ordem. Este método pode ser descrito pelo seguinte procedimento:

$$\begin{aligned}
 k_1 &= hf(t_n, y_n), \\
 k_2 &= hf\left(t_n + \frac{h}{2}, y_n + \frac{k_1}{2}\right), \\
 k_3 &= hf\left(t_n + \frac{h}{2}, y_n + \frac{k_2}{2}\right), \\
 k_4 &= hf(t_n + h, y_n + k_3), \\
 y_{n+1} &= y_n + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6} + O(h^5).
 \end{aligned} \tag{3.46}$$

Desta forma, o método de Runge-Kutta de quarta ordem consiste no cálculo de quatro valores intermediários, que são os produtos entre o comprimento de passo h por valores da derivada $y'(t)$ em diferentes pontos, para então calcular a solução da função y_{n+1} com base em uma média ponderada destes valores. Nota-se que o método do ponto intermediário calcula os mesmos fatores k_1 e k_2 para encontrar a solução da função na equação 3.46, entretanto no método de Runge-Kutta de quarta ordem existem mais dois fatores, k_3 e k_4 , que são encontrados de acordo com uma seqüência de dependência entre eles. O gráfico da figura 3.6 ilustra o processo de aproximação deste método.

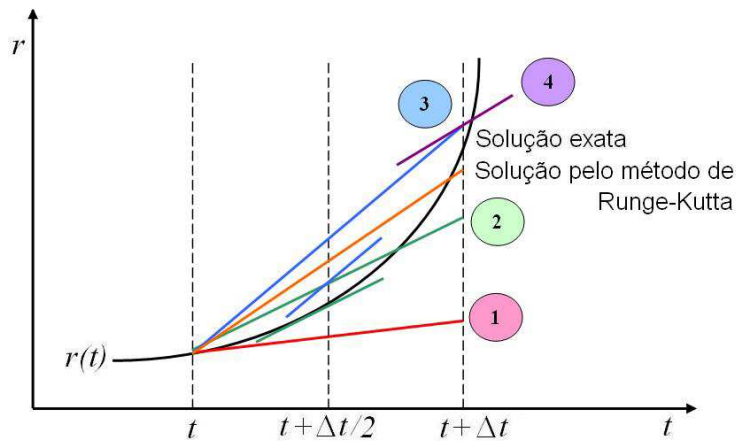


Figura 3.6: Gráfico das aproximações feitas no método de Runge-Kutta de quarta ordem.

De forma similar ao que foi feito na adaptação do método do ponto intermediário para integrar o modelo de tecido ao longo do tempo, o método de Runge-Kutta precisa gerar novas iterações do modelo com a geometria de superfícies auxiliares para obter as estimativas de velocidade nos pontos intermediários do intervalo $[t, t + \Delta t]$. Entretanto, neste método é preciso de três iterações do modelo em uma única iteração da solução numérica, que correspondem as três transições entre os fatores k_1 e k_2 , k_2 e k_3 , e k_3 e k_4 . Os seguintes passos representam este método:

1. Armazenar $\mathbf{r}(t)$ e $\mathbf{v}(t)$ e encontrar $\mathbf{r}(t + \frac{\Delta t}{2})$ usando o método de Euler

$$\begin{aligned} k_1 &= \Delta t \mathbf{v}_{k,l}(t), \\ \mathbf{r}_{k,l}(t + \frac{\Delta t}{2}) &= \mathbf{r}_{k,l}(t) + \frac{k_1}{2}, \end{aligned}$$

2. Calcular uma iteração do modelo de tecido para determinar as novas acelerações de $\mathbf{r}(t + \frac{\Delta t}{2})$ com a equação 3.37

$$\mathbf{a}_{k,l}(t + \frac{\Delta t}{2}) = \frac{\mathbf{f}\mathbf{e}_{k,l}(t + \frac{\Delta t}{2}) - \varrho \mathbf{v}_{k,l}(t + \frac{\Delta t}{2}) - \mathbf{f}\mathbf{i}_{k,l}(t + \frac{\Delta t}{2})}{\mu},$$

3. Determinar a velocidade no ponto intermediário $t + \frac{\Delta t}{2}$ usando a aceleração $\mathbf{a}_{k,l}(t + \frac{\Delta t}{2})$ e calcular k_2

$$\begin{aligned} \mathbf{v}_{k,l}(t + \frac{\Delta t}{2}) &= \mathbf{v}_{k,l}(t) + \frac{\Delta t}{2} \mathbf{a}_{k,l}(t + \frac{\Delta t}{2}), \\ k_2 &= \Delta t \mathbf{v}_{k,l}(t + \frac{\Delta t}{2}), \end{aligned}$$

4. Calcular a nova posição dos pontos da superfície no ponto intermediário

$$\mathbf{r}_{k,l}(t + \frac{\Delta t}{2}) = \mathbf{r}_{k,l}(t) + \frac{k_2}{2},$$

5. Calcular uma outra iteração do modelo de tecido para determinar as acelerações de $\mathbf{r}(t + \frac{\Delta t}{2})$

$$\mathbf{a}_{k,l}(t + \frac{\Delta t}{2}) = \frac{\mathbf{f}\mathbf{e}_{k,l}(t + \frac{\Delta t}{2}) - \varrho \mathbf{v}_{k,l}(t + \frac{\Delta t}{2}) - \mathbf{f}\mathbf{i}_{k,l}(t + \frac{\Delta t}{2})}{\mu},$$

6. Determinar a nova velocidade no ponto intermediário $t + \frac{\Delta t}{2}$ usando a aceleração $\mathbf{a}_{k,l}(t + \frac{\Delta t}{2})$

e calcular k_3

$$\begin{aligned}\mathbf{v}_{k,l}(t + \frac{\Delta t}{2}) &= \mathbf{v}_{k,l}(t) + \frac{\Delta t}{2} \mathbf{a}_{k,l}(t + \frac{\Delta t}{2}), \\ k_3 &= \Delta t \mathbf{v}_{k,l}(t + \frac{\Delta t}{2}),\end{aligned}$$

7. Calcular a nova posição dos pontos da superfície no ponto $t + \Delta t$

$$\mathbf{r}_{k,l}(t + \Delta t) = \mathbf{r}_{k,l}(t) + k_3,$$

8. Calcular a última iteração do modelo para determinar as acelerações de $\mathbf{r}(t + \Delta t)$

$$\mathbf{a}_{k,l}(t + \Delta t) = \frac{\mathbf{f}_{e,k,l}(t + \Delta t) - \varrho \mathbf{v}_{k,l}(t + \Delta t) - \mathbf{f}_{i,k,l}(t + \Delta t)}{\mu},$$

9. Determinar a nova velocidade no ponto $t + \Delta t$ usando a aceleração $\mathbf{a}_{k,l}(t + \Delta t)$ e calcular k_4

$$\begin{aligned}\mathbf{v}_{k,l}(t + \Delta t) &= \mathbf{v}_{k,l}(t) + \Delta t \mathbf{a}_{k,l}(t + \Delta t), \\ k_4 &= \Delta t \mathbf{v}_{k,l}(t + \frac{\Delta t}{2}),\end{aligned}$$

10. Atualizar a posição final usando $\mathbf{r}_{k,l}(t)$ e os quatro fatores k_1, k_2, k_3 e k_4

$$\mathbf{r}_{k,l}(t + \Delta t) = \mathbf{r}_{k,l}(t) + \frac{k_1}{6} + \frac{k_2}{4} + \frac{k_3}{4} + \frac{k_4}{6}, \quad (3.47)$$

11. Atualizar a velocidade final usando $\mathbf{v}_{k,l}(t)$ e a última aceleração calculada $\mathbf{a}_{k,l}(t + \Delta t)$

$$\mathbf{v}_{k,l}(t + \Delta t) = \mathbf{v}_{k,l}(t) + \Delta t \mathbf{a}_{k,l}(t + \Delta t). \quad (3.48)$$

Assim, o método de Runge-Kutta de quarta ordem possui mais passos e cálculos que os métodos explícitos de Euler e do ponto intermediário, demandando um maior tempo de processamento, principalmente por ter que realizar três iterações do modelo de tecido, porém ele possui precisão numérica de quarta ordem, a qual é superior aos demais métodos apresentados. O último passo deste procedimento, referente à atualização das velocidades, poderia ser calculado da mesma forma que as posições, utilizando como fatores a aceleração inicial em t e as três calculadas durante este procedimento. Contudo, optou-se por adaptá-la desta forma devido à última velocidade encontrada ser a mais precisa dentre as estimativas usadas para calcular os quatro fatores.

3.2.4 Método *Leapfrog*

O esquema *Leapfrog* é um método muito similar ao explícito de Euler, possuindo uma formulação semelhante e sendo tão eficiente quanto a ele, contudo, possui uma precisão numérica maior. Neste método, ao invés de atualizar as posições e as velocidades dos pontos da superfície no mesmo instante $t + \Delta t$, primeiro a posição é atualizada no instante $t + \Delta t$ com base na velocidade de $t + \frac{\Delta t}{2}$, para a velocidade ser encontrada em $t + \frac{3\Delta t}{2}$. A nova velocidade de cada passo é então atualizada em $t + \frac{\Delta t}{2}$ sempre com base na velocidade de $t - \frac{\Delta t}{2}$ e assim sucessivamente, de forma que as posições e velocidades vão se saltando ao longo do tempo, como mostra a figura 3.7.

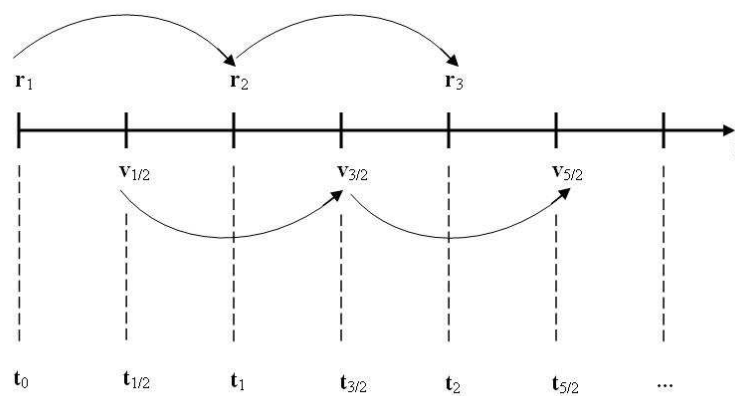


Figura 3.7: Saltos de posições e velocidades no método *Leapfrog*.

As atualizações da posição e velocidade dos pontos da superfície são feitas com o método explícito de Euler, porém cada uma delas feitas em instantes diferentes, separados de $\frac{\Delta t}{2}$. Como as velocidades são calculadas sempre no meio de cada passo de tempo, elas tendem a aumentar a precisão da solução das posições, de forma similar ao que acontece no método do ponto intermediário, fazendo com que o método de *Leapfrog* possua uma precisão numérica de segunda ordem. Ao adaptar este método para integrar o modelo de tecido ao longo do tempo é preciso, no instante inicial t_0 , além de todos os elementos do modelo, a velocidade em $t + \frac{\Delta t}{2}$ para que o procedimento possa iniciar os saltos das estimativas de velocidade e posição ao longo do eixo do tempo. Para estimar esta velocidade inicial em $t + \frac{\Delta t}{2}$ optou-se por usar o método do ponto intermediário, somente na primeira iteração a partir do instante t_0 , de forma que as demais iterações podem ser descritas pelos passos:

1. Cálculo da aceleração resultante

$$\mathbf{a}_{k,l}(t) = \frac{\mathbf{f}_{e,k,l}(t) - \rho \mathbf{v}_{k,l}(t) - \mathbf{f}_{k,l}(t)}{\mu}, \quad (3.49)$$

2. Atualização das posições

$$\mathbf{r}_{k,l}(t + \Delta t) = \mathbf{r}_{k,l}(t) + \mathbf{v}_{k,l}(t + \frac{\Delta t}{2})\Delta t, \quad (3.50)$$

3. Atualização das velocidades

$$\mathbf{v}_{k,l}(t + \frac{3\Delta t}{2}) = \mathbf{v}_{k,l}(t + \frac{\Delta t}{2}) + \Delta t \mathbf{a}_{k,l}(t). \quad (3.51)$$

Desta forma, o método *Leapfrog*, partindo da superfície de posições $\mathbf{r}(t_0)$ e da velocidade em $t_0 + \frac{\Delta t}{2}$, sendo esta última obtida pelo método do ponto intermediário, calcula em cada iteração do procedimento três passos, que correspondem ao cálculo da aceleração resultante e às atualizações de posição e velocidade. Este esquema, à primeira vista, parece ser muito parecido com o método explícito de Euler adaptado para integrar o modelo de tecido, pois após a aceleração resultante $\mathbf{a}(t)$ ser encontrada pela equação 3.49, a posição e velocidade são atualizadas usando intervalos de um passo de tempo inteiro com a formulação de Euler. A diferença reside no fato delas serem encontradas em instantes diferentes, fazendo com que a eficiência computacional do esquema de Euler seja aplicada com a idéia do método do ponto intermediário para aumentar a precisão numérica.

3.2.5 Método de Verlet

Outra forma de realizar uma integração explícita ao longo do tempo é com o método de Verlet. Neste método são usados dois estados da superfície, no instante corrente t e no anterior $t - \Delta t$, e a evolução da solução numérica em $t + \Delta t$ ao longo do tempo é feito sem o cálculo direto da velocidade, que é aproximada pela variação entre as geometrias da superfície no intervalo $[t - \Delta t, t + \Delta t]$. Este método de integração é muito usado na dinâmica molecular para integrar equações de movimento, sendo baseado na série de Taylor. Partindo da equação 3.36 é possível escrever duas expansões truncadas no elemento de quarta ordem, feita uma para frente $t + \Delta t$

$$x(t_0 + \Delta t) = x(t_0) + \Delta t x'(t_0) + \frac{\Delta t^2}{2!} x''(t_0) + \frac{\Delta t^3}{3!} x'''(t_0) + O(\Delta^4), \quad (3.52)$$

e outra para trás $t - \Delta t$

$$x(t_0 - \Delta t) = x(t_0) - \Delta t x'(t_0) + \frac{\Delta t^2}{2!} x''(t_0) - \frac{\Delta t^3}{3!} x'''(t_0) + O(\Delta^4). \quad (3.53)$$

Somando as equações 3.52 e 3.53, chega-se a expressão

$$x(t_0 + \Delta t) - x(t_0 - \Delta t) = 2x(t_0) + \Delta t^2 x''(t_0) + O(\Delta^4),$$

que reescrita com o termo $x(t_0 - \Delta t)$ do lado direito torna-se a expressão convencional de Verlet

$$x(t_0 + \Delta t) = 2x(t_0) - x(t_0 - \Delta t) + \Delta t^2 x''(t_0) + O(\Delta^4). \quad (3.54)$$

Desta forma, o método de Verlet atinge uma precisão numérica de terceira ordem por possuir um termo de erro de truncamento de quarta ordem $O(\Delta t^4)$ em sua formulação. A grande idéia deste método é atingir esta boa precisão numérica manipulando as expansões em série de Taylor de forma a cortar o segundo e o quarto elemento das séries, aumentando a precisão sem aumentar o número de cálculos, como acontece nos métodos do ponto intermediário e de Runge-Kutta. Entretanto, como este método requer o armazenamento de dois vetores de posições, sendo um referente ao instante atual e outro ao anterior, a dinâmica de tecidos precisa de duas soluções iniciais para começar a encontrar novos estados do tecido, ao contrário dos demais métodos apresentados até aqui nesta seção.

Considerando o método de Verlet para integrar o modelo de tecido ao longo do tempo, percebe-se que embora sua formulação permita a solução numérica sem o cálculo direto da velocidade, sendo ela calculada indiretamente pela variação das posições da superfície em instantes diferentes, a aceleração de cada ponto no modelo é obtida com a equação 3.37, que utiliza a velocidade. Sendo assim, esta vantagem de não calcular a velocidade é descartada na adaptação do método de Verlet para o modelo de tecido, calculando ela ao final de cada iteração do procedimento, que é descrito pelos passos:

1. Cálculo da aceleração resultante

$$\mathbf{a}_{k,l}(t) = \frac{\mathbf{f}_{e,k,l}(t) - \varrho \mathbf{v}_{k,l}(t) - \mathbf{f}_{i,k,l}(t)}{\mu}, \quad (3.55)$$

2. Atualização das posições

$$\mathbf{r}_{k,l}(t + \Delta t) = 2 * \mathbf{r}_{k,l}(t) - \mathbf{r}_{k,l}(t - \Delta t) + \mathbf{a}_{k,l}(t) \Delta t^2, \quad (3.56)$$

3. Atualização das velocidades

$$\mathbf{v}(t) = \frac{\mathbf{r}_{k,l}(t + \Delta t) - \mathbf{r}_{k,l}(t - \Delta t)}{2\Delta t}. \quad (3.57)$$

3.3 Trabalhos Relacionados

No capítulo 2 apresentou-se um panorama sobre os trabalhos relacionados à simulação de tecidos destacando o tipo de modelagem de tecido em conjunto com a solução numérica usada para obter a nova geometria do tecido em cada instante de tempo, visando encaixar a proposta desta dissertação em uma lacuna existente na literatura. Esta seção resume os trabalhos que estão relacionados à simulação de tecidos e utilizam os métodos numéricos explícitos apresentados na seção 3.2. Assim, após explicar as adaptações desses métodos para serem usados com o modelo proposto por de Melo [4], os trabalhos que os utilizaram para simular tecidos com outras abordagens de modelagem são discutidos nesta seção para salientar as diferenças.

O método explícito de Euler, o mais simples entre os métodos explícitos, foi utilizado em 1995 por Provot na versão em que a velocidade é atualizada antes da posição para ser usada no cálculo da nova posição de cada ponto da superfície do tecido [2]. Neste trabalho o autor considera que a aceleração resultante de cada ponto seja dada pela lei de Newton $\mathbf{F} = m\mathbf{a}$, que corresponde a aceleração do instante $t + \Delta t$ conforme o esquema abaixo

$$\begin{aligned} \mathbf{a}_{i,j}(t + \Delta t) &= \frac{1}{\mu} \mathbf{F}_{i,j}, \\ \mathbf{v}_{i,j}(t + \Delta t) &= \mathbf{v}_{i,j}(t) + \mathbf{a}_{i,j}(t + \Delta t) \Delta t, \\ \mathbf{r}_{i,j}(t + \Delta t) &= \mathbf{r}_{i,j}(t) + \mathbf{v}_{i,j}(t + \Delta t) \Delta t, \end{aligned}$$

em que μ é a massa de cada ponto e o termo $\mathbf{F}_{i,j}$ refere-se às forças resultantes entre as externas e internas em cada partícula (i, j) do modelo massa-mola. As forças externas podem ser adicionadas para todos os pontos, como a força gravitacional e a força fluido, ou isoladamente, como uma força pontual aplicada somente a um ponto. Já as forças internas são calculadas com base na Lei de Hooke $\mathbf{F} = k\mathbf{x}$, usando os deslocamentos das molas que conectam cada ponto aos seus pontos adjacentes.

O principal problema na dinâmica de tecidos proposta por Provot é a instabilidade numérica, causada por diferentes motivos. O primeiro deles é que esta abordagem não suporta grandes mudanças no valor dos elementos do modelo entre um instante e outro de tempo, a menos que o passo de tempo seja suficientemente pequeno. Em outras palavras, se a aceleração e velocidade crescem de forma brusca é preciso reduzir significativamente o passo de tempo para garantir a estabilidade numérica da simulação. Um outro fator de instabilidade no trabalho de Provot é quando uma partícula fica rapidamente muito longe das demais, fazendo com que a força interna nela seja brevemente alta, que resulta em uma aceleração alta somente em um pequeno instante de tempo. Porém, a velocidade é atualizada com base nesta alta aceleração e aplicada a todo o passo de tempo, ao invés de ser usada somente no intervalo de tempo que ela realmente existiu.

No ano seguinte ao trabalho de Provot, Volino et al. publicaram um modelo baseado na mecânica de partículas que utilizava o método explícito do ponto intermediário para simular tecidos [29]. Neste trabalho os autores descreveram um modelo versátil de superfície deformável que aborda a colisão com outros objetos e auto-colisões, usando o método numérico do ponto intermediário, que além de ter um bom desempenho computacional, é uma ordem de precisão maior e apresenta menores problemas de instabilidade em relação ao método de Euler. Neste trabalho os autores também utilizam a segunda lei de Newton $\mathbf{F} = m\mathbf{a}$ para encontrar a aceleração resultante e a lei de Hooke $\mathbf{F} = k\mathbf{x}$ em adição às expressões angulares para encontrar a força interna de cada partícula.

Contudo, mesmo com maior estabilidade numérica em relação ao método de Euler, o método do ponto intermediário no trabalho de Volino et al. ainda possuía problemas de instabilidade, que eram resolvidos de uma forma artificial, diminuindo a precisão física do modelo. Os autores monitoram constantemente as variações da energia mecânica local para medir a instabilidade do sistema e, uma vez detectado seu aumento na região vizinha de alguma partícula, uma distribuição artificial de energia cinemática é feita nos elementos envolvidos, acelerando a propagação da perturbação. Outra dificuldade deste trabalho é para controlar os parâmetros mecânicos do material do tecido, que são definidos com a atribuição de valores para os coeficientes de rigidez das molas. Como a idéia era desenvolver um modelo de superfície versátil, que pudesse também ser usado para outros objetos fora os tecidos, Volino et al. propuseram utilizar coeficientes de Poisson, módulos Young e densidades para descrever as propriedades elásticas do objeto deformável [29].

Um ano mais tarde, Eberhardt et al. utilizaram o método explícito de Runge-Kutta de quarta ordem sobre um modelo de partículas, baseado no modelo de Breen et al. [26], para encontrar a nova geometria da superfície do tecido em cada instante de tempo [14]. Neste trabalho foi implementado o método de Runge-Kutta com passos de tempo adaptativos e também o método de Burlish-Stoer, chegando à conclusão que o método de Runge-Kutta é muito mais eficiente computacionalmente. Para o modelo de tecido, os autores estenderam o modelo de Breen et al. introduzindo técnicas para medir as forças internas com maior precisão e considerando propriedades específicas de tecidos, como comportamento anisotrópico e histerese, além de incrementar o modelo para produzir a dinâmica de tecidos considerando a resistência do ar, vento e fricção de superfície.

Com o destaque das vantagens do método de Runge-Kutta no trabalho de Eberhardt [14], Volino e Thalmann [30], em 1997, apresentaram um trabalho utilizando o método de Runge-Kutta descrito em [28]. Este método é uma adaptação do método de Runge-Kutta de quarta ordem que atinge um estado cinemático de todos os vértices do objeto simulado com uma estimativa de erro cometido neste estado. Esta estimativa é então usada como um fator de amortecimento que coloca o estado do modelo na menor deformação possível dentro do intervalo de erro considerado. Além disso, os autores realizaram uma correção da posição e velocidade dos vértices, após o processo de cálculo da

dinâmica, que é feita para cada aresta comparando e corrigindo a posição de seus vértices extremos. Como resultado, os autores obtiveram um sistema robusto de simulação de tecidos, que previne o aumento da energia de deformação para tratar imprecisões numéricas, fazendo com o que modelo raramente resulte em uma situação de instabilidade numérica, na qual os pontos do tecido apresentam enormes deslocamentos e deixam de formar uma superfície regular.

Após o trabalho de Barraf e Witkin [13] em 1998, o uso de métodos implícitos para simular tecidos ganharam grande destaque, fazendo com que o emprego dos métodos explícitos para este propósito diminuísse. Contudo, em 2001, Oshita e Makinouchi apresentaram um trabalho utilizando o método explícito de Runge-Kutta de quarta ordem juntamente com um modelo massa-mola para produzir simulações de tecidos em tempo real [31]. Neste trabalho os autores utilizavam uma malha esparsa de partículas, que tinha no máximo cem pontos, e dividiram a dinâmica de tecidos em duas etapas: a parte de geração da superfície, que é feita com o método de Runge-Kutta, e a parte de suavização, a qual é processada pela combinação da modelagem das forças internas com técnicas geométricas de subdivisão em triângulos, fazendo com que a superfície final tenha uma resolução em torno de cinco vezes maior e apresente algumas dobras criadas nesta segunda etapa.

Também em 2001, Volino e Thalmann publicaram um trabalho que comparava a eficiência de diferentes métodos numéricos para realizar a simulação de tecidos [32]. Eles consideraram dois métodos explícitos: o do ponto intermediário e do Runge-Kutta de quarta ordem, e dois implícitos: o de Euler para trás e o método de Rosenbrook, o qual corresponde a uma versão implícita do método de Runge-Kutta de quarta ordem. Estes métodos foram implementados sobre um sistema que permitia usar duas representações mecânicas para os tecidos: um modelo massa-mola simplificado e um modelo de superfície elástica baseada na mecânica de partículas. Segundo os autores, esta última representação do tecido tratava-se de uma abordagem completa de superfície elástica que permitia a simulação de elasticidade anisotrópica usando módulos Young, coeficientes de Poisson e viscosidade na modelagem do entrelaçamento dos fios do tecido.

Nesse último trabalho, Volino e Thalmann propuseram comparar as implementações dos diferentes métodos numéricos focando quatro características: o tempo de computação gasto por uma iteração do sistema, o valor do passo de tempo necessário para atingir uma dada precisão ou estabilidade numérica, a precisão desejada com relação a resolução espacial da malha discreta, e a estabilidade numérica do método. Em praticamente todos os quesitos e, principalmente no de tempo computacional, o método de Euler para trás em conjunto com o modelo massa-mola foi superior aos demais. Este método numérico era combinado com algumas iterações do método do gradiente conjugado para obter uma estimativa mais eficiente. Contudo, neste trabalho os autores não apresentaram as formulações dos modelos de tecidos usados nem indicaram modelos equivalentes na literatura, descrevendo apenas as duas representações sucintamente e de forma literal.

Adicionalmente, na descrição do modelo elástico os autores afirmaram que o curvamento do modelo, embora este tenha sido implementado, não é levado em consideração em seu trabalho, como também consideraram que as leis de elasticidade sejam computadas como interações entre cada três vértices de um triângulo que forma a superfície, refletindo um comportamento mecânico assumido como linear [32]. Desta forma, o modelo elástico considerado utiliza muitas simplificações, sendo possivelmente similar ao descrito pelos mesmos autores em 1995 [29], que baseados na mecânica de partículas não possuem a mesma precisão física dos modelos baseados na mecânica de contínuos, como o proposto por de Melo [4], utilizado neste trabalho.

Ainda em 2001, Jakobsen publicou na Internet um *site* que descreve os algoritmos desenvolvidos para o jogo *Hitman Codename 47*, da empresa *IO Interactive's Game* [33]. Neste trabalho o autor utiliza o esquema de integração de Verlet em conjunto com um modelo de partículas para processar o movimento de tecidos durante o jogo. O método modela corpos rígidos como partículas com restrições e conta com algoritmos que tratam colisões e penetrações por projeções, usando um método de relaxação para equilibrar o sistema. O resultado é um simulador de tecidos muito eficiente computacionalmente, pelo uso do método numérico de Verlet, e estável numericamente, em função da relaxação das forças internas para garantir um estado de equilíbrio. Estes dois pontos, bom desempenho e estabilidade, são cruciais em uma simulação de tecidos dentro de um jogo que, na prática, proporciona-lhe um maior realismo. Também voltado aos jogos, uma outra implementação do esquema de Verlet que considera passos de tempo adaptativos foi descrita em [34].

Em 2003, Bridson et al. [10] propuseram um esquema de integração temporal que mistura formulações implícitas com explícitas em conjunto com um modelo de tecido sofisticado, baseado na mecânica de partículas, para simular a dinâmica de tecidos. Eles utilizaram a integração explícita no cálculo das forças elásticas, as quais independem da velocidade, e a integração implícita na obtenção das forças de amortecimento, que dependem da velocidade. Desta forma, os autores combinaram o método explícito de *Leapfrog*, cuja precisão numérica é de segunda ordem, para atualizar as posições dos pontos da superfície do tecido, enquanto um método implícito de regra trapezoidal, também de precisão de segunda ordem, era usado para a atualização das velocidades dos pontos.

Também em 2003, Hauth publicou um trabalho voltado para a investigação de métodos numéricos aplicados especificamente para a simulação de tecidos [35]. Neste trabalho o autor descreve inicialmente diferentes tipos de discretização espacial e assume um modelo de tecido do tipo massa-mola, para então apresentar diversos métodos numéricos comparando os resultados de cada método com a solução analítica para funções paramétricas. Entre os métodos investigados, estão os métodos explícitos de Euler, do ponto intermediário, de Runge-Kutta de quarta ordem e o método chamado de Verlet, que segundo Hauth é também chamado de *Leapfrog*, embora este método que o autor se referiu corresponda ao método de *Leapfrog* apresentado na subseção 3.2.4, e não ao de Verlet apresentado

na subseção 3.2.5. Além disso, o autor também compara as versões implícitas dos métodos de Euler, ponto intermediário e Runge-Kutta de quarta ordem, além de métodos de alta ordem, como os métodos de Adams e métodos BDF (*Backward Differentiation Formulas*).

Apesar de focado para o uso de métodos numéricos aplicados em simulação de tecidos, Hauth não chegou a exemplificar ou comparar esses métodos com imagens de simulações que utilizam diferentes implementações de soluções numéricas sobre o modelo massa-mola considerado. Após a explicação e a comparação dos métodos numéricos com as soluções analíticas, Hauth analisa a estabilidade, eficiência e precisão de cada método, chegando a conclusão de que é praticamente impossível afirmar qual é o melhor método para uma dada aplicação sem que todos esses métodos sejam testados para o objetivo desejado [35]. O autor estende a mesma conclusão para a eficiência dos métodos investigados, pois enquanto geralmente os métodos implícitos requerem um maior tempo computacional por iteração, com eles é possível utilizar um passo de tempo mais largo em relação aos explícitos. Por fim, ele apresenta um digrama para auxiliar a tomada de decisão sobre qual método usar em determinadas aplicações, as dividindo em sistemas lineares e não lineares.

Um ano mais tarde, em 2004, Natsupakpong divulgou através de um *site* um estudo sobre métodos de integração numérica aplicados a simulação de objetos deformáveis, utilizando a modelagem massa-mola [36]. Neste trabalho a autora compara os métodos explícitos de Euler, do ponto intermediário e o de Runge-Kutta de quarta ordem, considerando também o método semi-implícito de Euler (usado por Provot [2]) e o método implícito de Euler. A comparação feita consistia em encontrar os maiores passos de tempo para cada método que resultava em uma simulação estável, bem como comparar os tempos totais gastos para processar determinadas simulações. O método semi-implícito de Euler foi que atingiu os melhores resultados neste estudo.

Finalizado este resumo sobre os principais trabalhos relacionados à simulação de tecidos que utilizam os métodos explícitos apresentados na seção 3.2, é possível mais uma vez destacar a lacuna existente neste ramo de pesquisa, que consiste em unir métodos explícitos com um modelo baseado na mecânica de contínuos. Um dos principais motivos da existência desta lacuna deve-se ao sucesso da combinação oposta: a de usar integrações implícitas ou semi-implícitas com modelos baseados na mecânica de partículas. Este sucesso foi impulsionado ao longo da última década por trabalhos que salientaram a vantagem em unir as duas abordagens e, especialmente nos trabalhos mais recentes como [9, 10, 5], os modelos de tecido baseados na mecânica de partículas são agregados a sofisticados esquemas numéricos que misturam formulações explícitas e implícitas e ainda algoritmos avançados de tratamento de colisões e auto-colisões. Contudo, alguns dos problemas presentes nos trabalhos comentados nesta seção não estão necessariamente ligados ao método numérico empregado, mas ao modelo de tecido considerado. Desta forma, esta dissertação investiga estes mesmos métodos explícitos aplicados ao modelo de tecido proposto por de Melo [4].

3.4 Resultados

Nesta seção são mostrados os resultados da implementação dos métodos apresentados na seção 3.2 sobre o modelo de tecido proposto por de Melo [4]. Os resultados da implementação dos métodos numéricos são apresentados em termos de estabilidade, eficiência e precisão. Para medir estes quesitos é usado um conjunto de seis simulações de tecidos, que possuem parâmetros de material e restrições diferentes e atingem determinadas formas em um estado de equilíbrio considerado, conforme explicado na subseção 3.4.1. As subseções seguintes apresentam os resultados de cada método separadamente, na mesma ordem em que eles foram apresentados na seção 3.2. Nestes resultados, primeiro o valor do passo de tempo é testado até que se atinja o valor máximo capaz de alcançar de forma estável o estado de equilíbrio desejado e em cada simulação, sendo encontrado também o número de iterações necessária para isso. Em seguida, este passo de tempo máximo é usado para medir a eficiência computacional: de uma forma relativa, que consiste no tempo gasto para processar as iterações necessárias para atingir o estado considerado; e de uma forma absoluta, tomando o tempo que o método realiza uma única iteração. Por fim, as imagens do estado de equilíbrio das simulações são mostradas para que na próxima seção a qualidade visual obtida seja discutida.

3.4.1 Simulações de Teste

As seis simulações de tecidos usadas para obter os resultados da implementação de cada método numérico serão chamadas de S_1 até S_6 , estando elas em ordem crescente de número de pontos que formam a superfície. A simulação S_1 é uma malha quadrada 10×10 suspensa pelos seus dois vértices extremos horizontais e superiores, posicionada na horizontal no instante t_0 e caindo de acordo com a força da gravidade ao longo da evolução do tempo. A simulação S_2 é similar à S_1 , diferindo por ser uma malha 20×20 e por ter um ponto extremo fixo ao invés de dois. S_3 é uma malha 30×30 que cai horizontalmente de acordo com a gravidade, simulando o caimento de um tecido sobre um banco de praça. A simulação S_4 é uma malha 40×40 , suspensa pelos seus dois pontos extremos laterais, sendo deformada pela força do vento como uma bandeira. S_5 é uma malha 50×50 que, começando na horizontal e agindo de acordo com a gravidade cai sobre uma poltrona. Por último, S_6 é uma malha 60×60 deformada por forças de compressão tangenciais ao plano da superfície inicial do tecido.

Em todas essas simulações é considerado um estado de equilíbrio, seja ele quando não ocorrem mais alterações na forma do tecido, como no estado final de S_1 , que é um pano preso em um varal, ou quando o tecido apresenta um comportamento que repete os mesmos tipos de deformações, como as dobras que se formam e desaparecem em uma bandeira hasteada ao vento, simulada em S_4 . Com exceção da simulação S_6 , a qual consiste em uma deformação de um pano somente com forças tangenciais de compressão, todas as demais utilizam a ação da gravidade no sentido $(0.0, -9.8, 0.0)$.

Assim, a partir de um plano que representa o tecido, os parâmetros e as restrições das simulações de teste determinaram o estado final de equilíbrio do tecido. Para a definição destes estados são consideradas situações reais, que foram fotografadas e estão ilustradas na figura 3.9. As superfícies iniciais de todas as simulações, com exceção de S_4 , estão posicionadas no plano xz , como ilustrado na figura 3.8(a), possuindo apenas áreas e resoluções espaciais diferentes. A simulação S_4 parte de uma superfície inicial posicionada no plano xy , conforme ilustrado em 3.8(b).

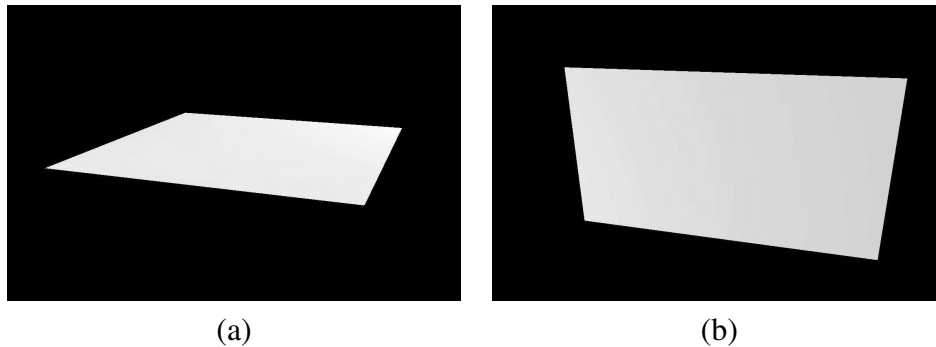


Figura 3.8: Superfície inicial das simulações: (a) S_1 , S_2 , S_3 , S_5 e S_6 ; e (b) S_4 .

A simulação S_1 , a partir do seu plano inicial, faz o tecido suspenso por seus dois pontos superiores extremos, cair na direção vertical de acordo com a força gravitacional, estabilizando na posição da figura 3.9(a). S_2 cai da mesma forma que S_1 , porém com apenas um ponto fixo, sendo considerado como estado de equilíbrio a foto da figura 3.9(b). A simulação S_3 também cai de acordo com a gravidade e atinge no estado de equilíbrio o formato de um tecido cobrindo um banco de praça, conforme a figura 3.9(c). A simulação S_4 é presa por dois pontos extremos laterais e sofre a força do vento, conforme a foto da figura 3.9(d), que foi tirada com o auxílio de dois ventiladores em direções opostas. Como nesta situação o tecido não para de se movimentar, o formato desta foto é considerado como estado de equilíbrio porque o tecido, após cair com a gravidade e sofrer a força do vento, consegue se manter na mesma linha horizontal, perpendicular a haste em que ele está preso. O tecido de S_5 cai de acordo com a gravidade definindo a forma de uma poltrona, conforme a figura 3.9(e). Por fim, a simulação S_6 , sem a presença da força de gravidade, deforma um pedaço de pano com forças de compressão no plano xz , provocando diversas dobras no tecido, como mostra a figura 3.9(f).

Desta forma, esses seis estados de equilíbrio das fotografias são usados como referência para as simulações de teste, objetivando medir a estabilidade numérica, a eficiência computacional e a qualidade das imagens produzidas pelos diferentes métodos numéricos. A palavra estabilidade é usada no contexto de simulações computacionais com diferentes interpretações. Neste trabalho, ela refere-se à estabilidade numérica associada à capacidade do método em produzir uma simulação de forma estável, evitando o chamado problema de alguns pontos da superfície sofrem deslocamentos muito grandes e deixarem de formar uma superfície regular. Este problema está diretamente ligado

ao tamanho do passo de tempo usado na integração ao longo do tempo. Quanto maior é o passo de tempo, maior a possibilidade de ocorrer tal fenômeno. Devido as diferentes formulações dos métodos apresentados, é possível que haja diferença entre os valores máximos suportados por cada método para manter as simulações de teste estáveis.

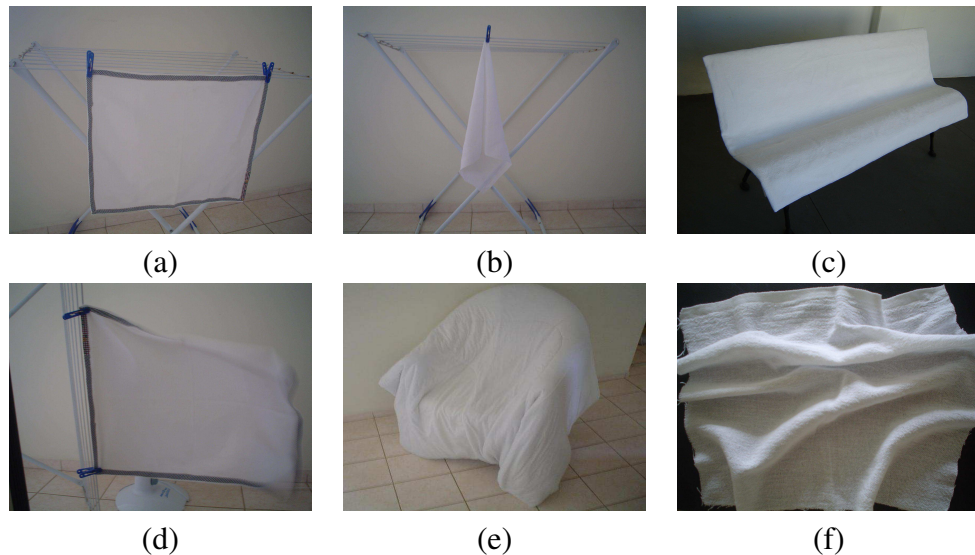


Figura 3.9: Fotos dos estados de equilíbrio.

Diante desta possível diferença no valor máximo do passo de tempo entre os métodos implementados, a eficiência computacional deve ser medida de diferentes maneiras. O primeiro modo é avaliando qual dos métodos processa o mesmo tempo de simulação da forma mais rápida, comparando os tempos obtidos. A segunda forma é medindo quanto tempo cada método demora para atingir os estados de equilíbrio citados, verificando quantas iterações são necessárias para isso. Neste teste o valor máximo do passo de tempo influencia diretamente o número de iterações necessárias para atingir o equilíbrio, permitindo, por exemplo, que o método de Runge-Kutta possa chegar em um estado de equilíbrio em menos tempo e iterações que Euler, mesmo com Euler processando cada iteração da dinâmica de tecidos de forma muito mais rápida. Para avaliar a precisão do método são utilizados os resultados visuais obtidos, comparando-os com as fotos da figura 3.9.

Desta forma, para cada método, o primeiro teste consiste em verificar qual é o maior passo de tempo que ele consegue alcançar esses estados de equilíbrio de maneira estável para cada simulação separadamente. Este primeiro resultado é usado para discutir a estabilidade numérica do método e também sua eficiência de uma forma relativa, analisando quantas iterações são necessárias e qual é o tempo gasto para atingir esses estados. Em seguida, a eficiência é medida de forma absoluta, que consiste em coletar os tempos que cada método processa uma única iteração. Esta etapa pode ser feita com qualquer passo de tempo que mantenha a simulação estável, pois o intuito é fixar um valor

de referência para que os tempos de cada método possam ser comparados. Por último, as imagens dos estados de equilíbrio obtidas com cada método são mostradas para que se verifique quais são as diferenças no aspecto visual.

3.4.2 Método de Euler

Como na subseção 3.2.1 foram apresentadas duas versões do método de Euler, foram feitos testes preliminares para decidir qual versão usar e, neste sentido, prevaleceu a versão usada por Provot [2], que é muitas vezes chamada de método semi-explícito de Euler pelo fato de atualizar as novas posições dos pontos já com a velocidade atualizada. Assim, todos os resultados desta subseção referem-se a esta versão do método. É interessante iniciar a exposição dos resultados pelo método de Euler, pois ele é um excelente parâmetro de referência para as comparações, afinal na teoria é o que possui maior eficiência computacional e a menor estabilidade e precisão numérica. Todos os tempos de processamento desta seção foram obtidos em um processador AMD 64 de 3.2 GHz equipado com 2 GB de memória RAM e com uma placa de vídeo 7800GS.

O primeiro teste executado é referente à verificação do maior passo de tempo que este método suporta para alcançar de maneira estável os estados de equilíbrio de cada simulação, mostrados na seção 3.4.1. Começando com um passo de tempo pequeno, $\Delta t = 0.001$, e o incrementando sempre com este mesmo valor, é possível encontrar o limiar da estabilidade numérica em cada simulação. Uma vez encontrado este valor, ele é usado para atingir os estados de equilíbrio e verificar quantas iterações foram necessárias para isto. A tabela 3.2 sintetiza os resultados.

Simulação	Passo de tempo	Número de iterações
S_1	0.012	1150
S_2	0.004	2050
S_3	0.003	3650
S_4	0.002	1850
S_5	0.002	10000
S_6	0.002	2500

Tabela 3.1: Passos de tempo máximos e iterações necessárias do método de Euler.

Com a tabela 3.1 pode-ser notar que os valores do passo de tempo obtidos são próximos, com exceção de S_1 , que consegue utilizar um passo de tempo aproximadamente uma ordem maior que os demais. Isto ocorre porque as simulações de testes foram criadas de acordo com o tipo de dobra e variação que a superfície sofre. Assim, a simulação S_1 é a que possui menor criação de dobras e maior suavidade na variação de sua curvatura, conseguindo utilizar um passo de tempo maior com estabilidade. O número de iterações necessárias na terceira coluna desta tabela mostra quantas vezes a solução numérica precisa ser repetida, com o maior passo de tempo possível encontrado, para que

a superfície inicial do tecido avance até o estado de equilíbrio considerado. Este valor é usado para medir o tempo que o método demora a chegar no estado de equilíbrio, fixando o número de iterações necessárias e o valor máximo do passo de tempo, que são informados na tabela 3.1.

Simulação	Número de Vértices	T_i	T_e
S_1	100	0.0018s	2s
S_2	400	0.0072s	15s
S_3	900	0.0165s	59s
S_4	1600	0.0301s	59s
S_5	2500	0.0493s	8 : 13s
S_6	3600	0.071s	2 : 58s

Tabela 3.2: Tempos de processamento do método de Euler.

A terceira coluna da tabela 3.2, rotulada como T_i , corresponde ao tempo gasto para executar uma iteração de cada simulação, com qualquer passo de tempo estável. A quarta coluna desta tabela, chamada de T_e , corresponde ao tempo que o método gastou para atingir o equilíbrio, utilizando em cada simulação o número de iterações necessárias com o passo de tempo máximo, cujos valores são mostrados na tabela 3.1. Desta forma, têm-se medidas de tempo absolutas, que correspondem aos tempos da terceira coluna, tomados sempre durante o intervalo de uma única iteração; e medidas de tempo relativas, que correspondem aos valores da quarta coluna, gerados com um número variável de iterações para se atingir o estado de equilíbrio em cada simulação. Portanto, em outras palavras, o tempo T_e equivale ao tempo T_i multiplicado pelo número de iterações necessárias da tabela 3.1. As imagens obtidas pelo método de Euler são ilustradas na figura 3.10.

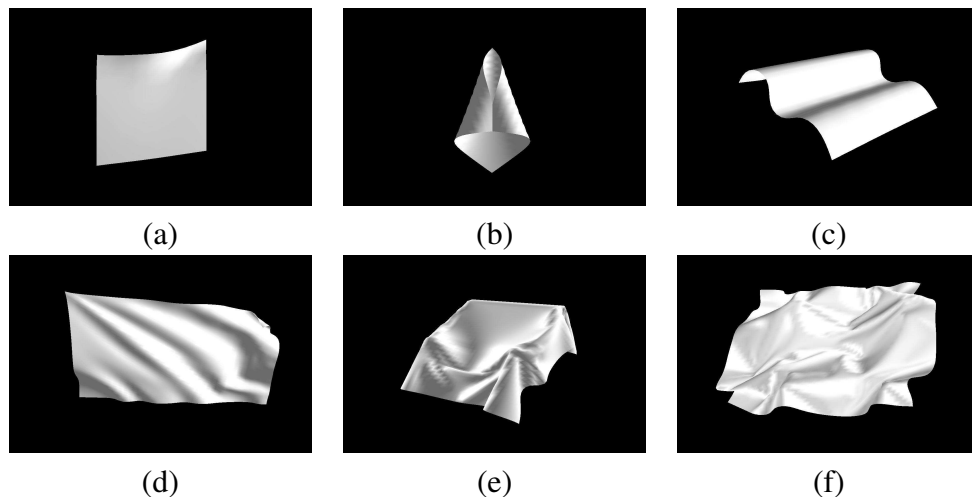


Figura 3.10: Resultados visuais do método de Euler.

3.4.3 Método do Ponto Intermediário

Da mesma forma que os resultados do método de Euler, o primeiro teste feito com a implementação do método do ponto intermediário é verificar os maiores passos de tempo que atingem os estados de equilíbrio de maneira estável e quantas iterações são necessárias para isto. A tabela 3.3 mostra os resultados obtidos. O segundo teste consiste em medir os tempos de processamento gastos nas simulações de um número fixo de iterações, que é o tempo total, e nas simulações até atingir o equilíbrio, que corresponde ao tempo relativo da tabela 3.4. A figura 3.11 ilustra os resultados visuais.

Simulação	Passo de tempo	Número de iterações
S_1	0.012	1500
S_2	0.004	2350
S_3	0.003	3800
S_4	0.002	2000
S_5	0.002	11000
S_6	0.002	4550

Tabela 3.3: Passos de tempo máximos e iterações necessárias do método do ponto intermediário.

Simulação	Número de Vértices	T_i	T_e
S_1	100	0.0036s	6s
S_2	400	0.0158s	37s
S_3	900	0.0343s	2 : 12s
S_4	1600	0.0651s	2 : 17s
S_5	2500	0.1083s	19 : 47s
S_6	3600	0.1571s	12 : 43s

Tabela 3.4: Tempos de processamento do método do ponto intermediário.

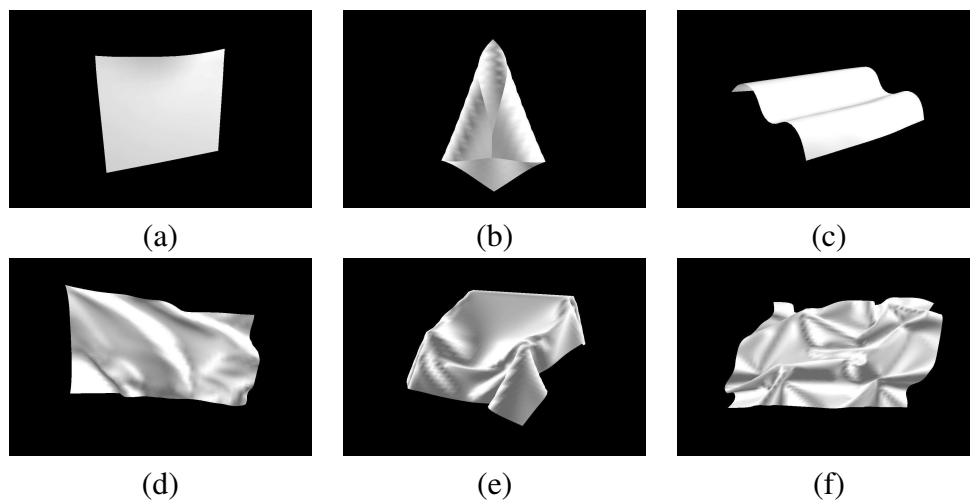


Figura 3.11: Resultados visuais do método do ponto intermediário.

3.4.4 Método de Runge-Kutta

A implementação do método de Runge-Kutta de quarta ordem é a que possui o maior número de cálculos, sendo executadas três iterações da dinâmica interna do tecido em cada iteração da solução numérica. Desta forma, é esperado que os tempos totais sejam superiores aos demais métodos. A tabela 3.5 apresenta os valores máximos do passo de tempo e o número de iterações necessárias para atingir o equilíbrio em cada simulação. A tabela 3.6 mostra os resultados do teste de desempenho computacional e a figura 3.12 ilustra os resultados visuais obtidos.

Simulação	Passo de tempo	Número de iterações
S_1	0.012	5000
S_2	0.004	3050
S_3	0.003	4100
S_4	0.002	2500
S_5	0.002	20000
S_6	0.002	6350

Tabela 3.5: Passos de tempo máximos e iterações necessárias do método de Runge-Kutta.

Simulação	Número de Vértices	T_i	T_e
S_1	100	0.0072s	34s
S_2	400	0.0303s	1 : 35s
S_3	900	0.0664	4 : 29s
S_4	1600	0.1182s	4 : 37s
S_5	2500	0.1856s	1 : 00 : 24s
S_6	3600	0.2659s	28 : 10s

Tabela 3.6: Tempos de processamento do método de Runge-Kutta.

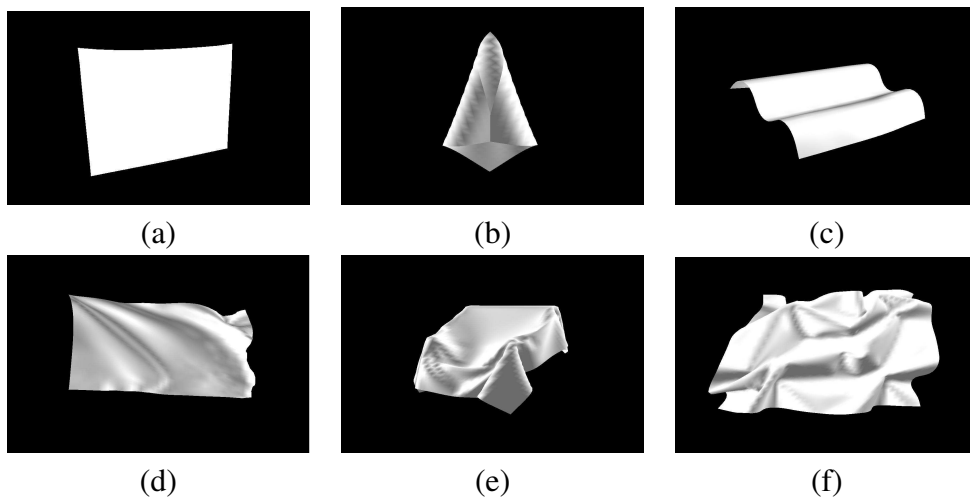


Figura 3.12: Resultados visuais do método de Runge-Kutta.

3.4.5 Método *Leapfrog*

A implementação do método *Leapfrog* é muito similar a de Euler, diferindo por uma estimativa da velocidade em $t_0 + \frac{\Delta t}{2}$ de cada ponto da superfície. Esta estimativa é feita pelo método do ponto intermediário e, partir dela, o método *Leapfrog* atualiza posições e velocidades da mesma forma explícita usada em Euler, porém com as posições e velocidades sendo atualizadas em instantes espaçados de $\frac{\Delta t}{2}$. A tabela 3.7 exibe os testes do passo de tempo e número de iterações, enquanto 3.8 mostra os resultados de desempenho e a figura 3.13 as imagens das simulações.

Simulação	Passo de tempo	Número de iterações
S_1	0.012	1200
S_2	0.004	2100
S_3	0.003	3650
S_4	0.002	1850
S_5	0.002	10000
S_6	0.002	5000

Tabela 3.7: Passos de tempo máximos e iterações necessárias do método *Leapfrog*.

Simulação	Número de Vértices	T_i	T_e
S_1	100	0.0019s	2s
S_2	400	0.0078s	16s
S_3	900	0.0184s	1 : 04s
S_4	1600	0.0331s	59s
S_5	2500	0.0523s	8 : 43s
S_6	3600	0.0776s	6 : 46s

Tabela 3.8: Tempos de processamento do método *Leapfrog*.

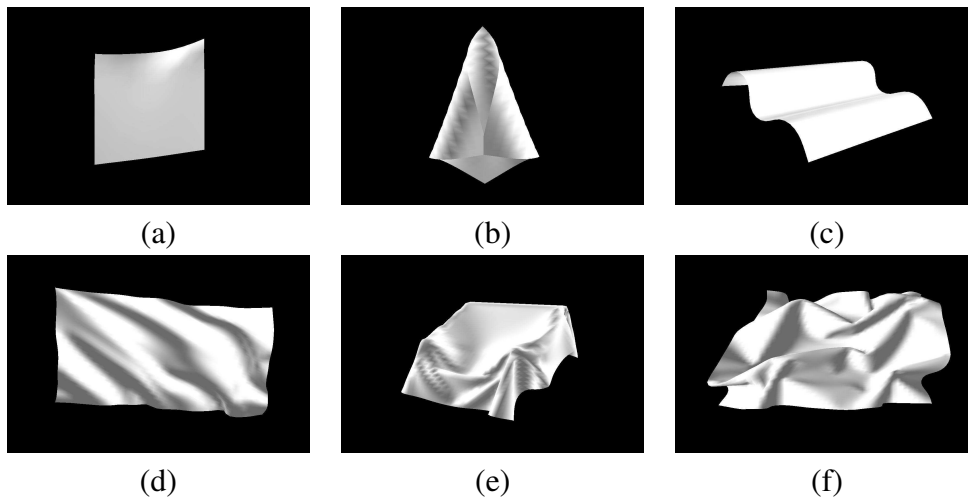


Figura 3.13: Resultados visuais do método *Leapfrog*.

3.4.6 Método de Verlet

A implementação do método de Verlet difere das demais feitas até aqui porque precisa de duas configurações iniciais da superfície do tecido. Como as forças externas são as responsáveis por produzir acelerações que deformaram o tecido, nesta implementação usou-se a mesma superfície como a inicial do instantes t_0 e t_1 , pois esta consideração não afeta a dinâmica do tecido. A tabela 3.9 apresenta os resultados de estabilidade numérica, a tabela 3.10 os resultados de eficiência computacional e a figura 3.14 os resultados visuais.

Simulação	Passo de tempo	Número de iterações
S_1	0.012	1150
S_2	0.004	1250
S_3	0.003	3200
S_4	0.002	1600
S_5	0.002	10000
S_6	0.002	2500

Tabela 3.9: Passos de tempo máximos e iterações necessárias do método de Verlet.

Simulação	Número de Vértices	T_i	T_e
S_1	100	0.0018s	2s
S_2	400	0.0076s	9s
S_3	900	0.0173s	55s
S_4	1600	0.0326s	59s
S_5	2500	0.0508s	8 : 38s
S_6	3600	0.0727s	3 : 03s

Tabela 3.10: Tempos de processamento do método de Verlet.

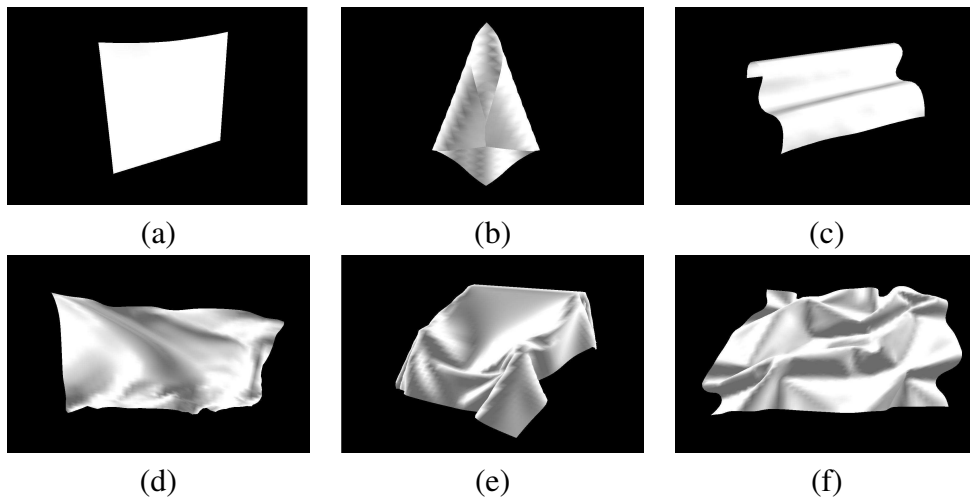


Figura 3.14: Resultados visuais do método de Verlet.

3.5 Análises

Nesta seção são analisados os resultados numéricos e visuais apresentados na seção 3.4. Visando organizar esta análise de acordo com os objetivos relacionados com as simulações de tecidos, ela é dividida em termos de estabilidade numérica, eficiência computacional e qualidade das imagens. A análise da estabilidade numérica utiliza os resultados obtidos nos testes do valor máximo do passo de tempo para cada método e é feita na subseção 3.5.1. A análise da eficiência computacional usa os resultados dos tempos de processamento absoluto e relativo para comparar o desempenho dos métodos, como discutido na subseção 3.5.2. Para analisar a qualidade das imagens obtidas em cada método são utilizadas as imagens dos estados de equilíbrio das simulações de teste, como apresentado na subseção 3.5.3. A escolha do método que melhor se adaptou ao modelo proposto por de Melo [4] é justificada na subseção 3.5.4.

3.5.1 Estabilidade Numérica

Como se viu na seção 3.4, os valores máximos do passo de tempo de cada simulação são iguais para todos os métodos, sendo este um resultado não previsto. Pela teoria, a expectativa era que pelo menos os métodos do ponto intermediário e o de Runge-Kutta de quarta ordem suportassem um passo de tempo maior que os demais, afinal estes métodos possuem novas iterações do modelo durante cada solução numérica, as quais aumentam a precisão da estimativa das velocidades dentro do intervalo entre dois passos subsequentes. Na verdade, existe uma diferença entre estes valores, porém ela é tão pequena que pode ser desprezada. Ao tentar simular a simulação de teste S_1 , por exemplo, com o valor 0.013, ao invés do valor máximo 0.012, é possível verificar quantas iterações conseguem ser executadas antes do estouro em cada método. Neste sentido, o método semi-explícito de Euler consegue processar 44 iterações, enquanto o método de *Leapfrog* faz 45 iterações, Verlet 46, o método do ponto intermediário 64 e Runge-Kutta de quarta ordem 91. Outra forma de verificar esta diferença é usando uma casa decimal a mais, que resulta em 0.0124 para os métodos do ponto intermediário e o de Runge-Kutta de quarta ordem e 0.0123 para os demais.

Investigando este comportamento similar dos métodos implementados em relação ao tamanho máximo do passo de tempo foi possível perceber que existe uma imposição de suavidade na variação dos elementos geométricos na modelagem do tecido que limita o método numérico utilizado. Esta imposição ocorre com a definição de uma superfície composta de um material de alta resistência à métrica e baixa resistência à curvatura, assumindo que as forças internas de cada ponto variam suavemente entre um instante de tempo e o seguinte. Desta forma, durante o processo de aumentar o valor do passo de tempo chega-se a um valor limite no qual as forças internas da superfície resultam em pequenos deslocamentos dentro do plano da superfície. Devido a planaridade destes deslocamentos,

eles estão associados à métrica da superfície, a qual, por sua vez, está relacionada com os coeficientes do tensor métrico, que são valores grandes devido a alta resistência à métrica. Estes pequenos deslocamentos então, em conjunto com os altos valores do tensor métrico, acabam propagando movimentos não desejados pelos vértices da superfície no sentido oposto mas com maior força. Não demora muito até que as deformações saiam do plano e a simulação resulte em uma situação de instabilidade.

Para melhor explicar essa situação em que a simulação mostra-se instável com determinado valor do passo de tempo, a figura 3.15 apresenta seqüências de imagens da simulação de teste S_1 em modo de visualização da malha sem iluminação para que suas faces fiquem nítidas. As figuras 3.15(a)-(d) mostram quatro imagens da simulação S_1 associadas a iteração i em que foram geradas pelo método semi-implícito de Euler com o passo de tempo 0.012. Já as figuras 3.15(e)-(h) mostram as quatro respectivas imagens geradas pelo mesmo método porém com o passo de tempo 0.013.

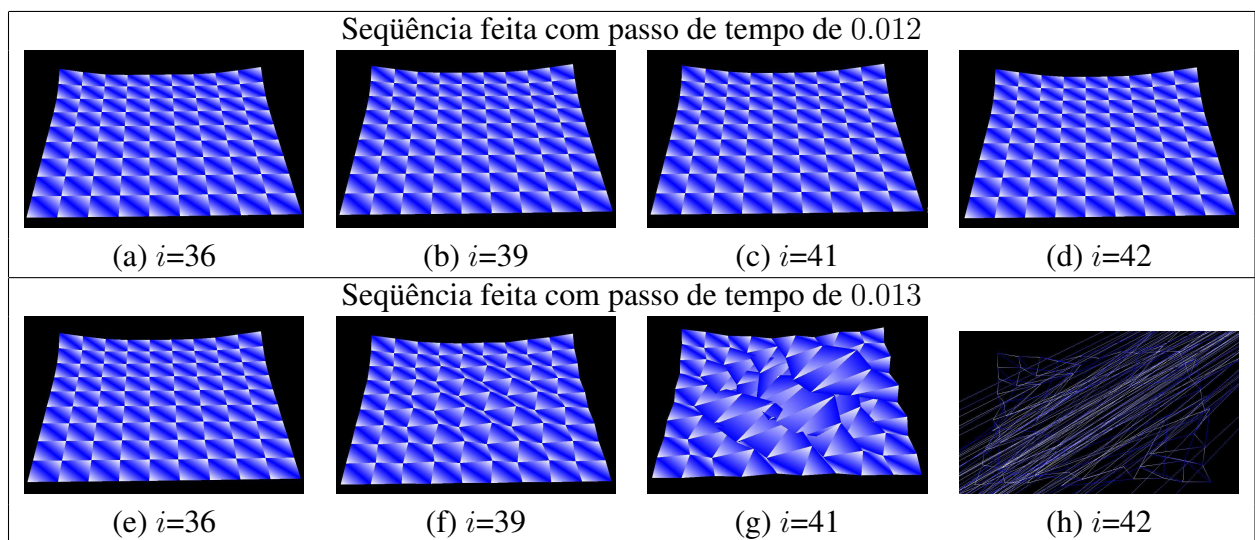


Figura 3.15: Instabilidade numérica causada por altos valores de Δt .

Na figura 3.15(a) é possível notar que os pontos de cada coluna e linha da superfície estão bem alinhados. Já na figura 3.15(e) os pontos apresentam deslocamentos que formam um serrilhado bem sutil ao longo das direções horizontais e verticais, quase imperceptíveis visualmente. Contudo, estes pequenos deslocamentos se propagam rapidamente e ficam mais evidentes, como mostra a imagem da figura 3.15(f). Apenas duas iterações depois as faces da superfície do tecido já estão todas comprometidas, embora os deslocamentos não desejados permaneçam contidos no mesmo plano. Uma iteração depois a superfície se torna caótica, resultando, duas iterações mais tarde, em uma situação de estouro, na qual é retornado ao usuário, normalmente, um erro de estouro na memória destinada a um número de ponto flutuante. A imagem da figura 3.15(h) mostra a superfície no modo de visualização *wireframe* para salientar que os deslocamentos, a partir desta iteração, ocorrem fora do plano

também. Enquanto isso, as imagens da figura 3.15(b)-(d) continuam mantendo seus pontos alinhados nas duas direções.

Desta forma, aumentando o passo de tempo em cada simulação chega-se ao momento em que o alto valor dos coeficientes do tensor métrico contribui para que pequenos deslocamentos inesperados sejam propagados e a simulação se torne instável. Apesar do exemplo fornecido ser somente da simulação S_1 com o método semi-explícito de Euler, as demais situações de instabilidade neste teste do valor máximo do passo de tempo apresentam o mesmo padrão, ou seja, iniciam com pequenos deslocamentos, depois eles são propagados com maior intensidade de forma planar e, por fim, saem do plano com deslocamentos enormes, resultando em uma superfície não regular.

Esse comportamento mostra que o modelo de tecido, ao utilizar valores dos coeficientes elásticos apropriados para os tecidos, requer uma suavidade entre as variações dos vetores normais, pois ao utilizar os valores de t para estimar os elementos de $t + \Delta t$ ocorre uma imposição de passo de tempo que torne esta variação suave, a qual possui maior influência sobre o método numérico em relação ao limite do passo de tempo. Uma forma simples de visualizar esta afirmação é fazer o mesmo teste alterando os valores do material do tecido de maneira a reduzir os altos valores do tensor métrico. Para tanto, considerou-se novamente a simulação S_1 e, ao invés de utilizar o coeficiente elástico $\zeta_{\alpha\beta} = 100$, como feito no teste original, atribui-se $\zeta_{\alpha\beta} = 5$. A tabela 3.11 mostra as diferenças.

Método	Passo de tempo Máximo
Semi-explícito de Euler	0.038s
Ponto Intermediário	0.039s
Runge-Kutta	0.040s
<i>Leapfrog</i>	0.038s
Verlet	0.037s

Tabela 3.11: Passos de tempo máximos de S_1 com $\zeta_{\alpha\beta} = 5$.

Assim, além dos valores máximos do passo de tempo serem maiores em relação ao valor 0.012, existe diferença entre os valores limites de cada método. Contudo, devido aos valores dos coeficientes $\zeta_{\alpha\beta}$ serem muito baixos, as simulações desse teste resultam em uma grande distensão da área da superfície, apresentando um comportamento que não é apropriado para os tecidos. A figura 3.16 ilustra uma imagem da simulação em um estado de equilíbrio que não existe mais movimento. Pode-se notar que, embora a massa da superfície seja a mesma do teste original, o baixo valor do tensor métrico fez com que a borda entre os pontos fixos da superfície fosse alongada como um queijo derretido.

É interessante notar que a superfície da figura 3.16 possui como restrição dois pontos fixos nos três eixos, portanto a distância entre eles é a mesma no instante inicial quanto no estado de equilíbrio mostrado, evidenciando que houve uma distensão não compatível com um tecido. Para que um tecido

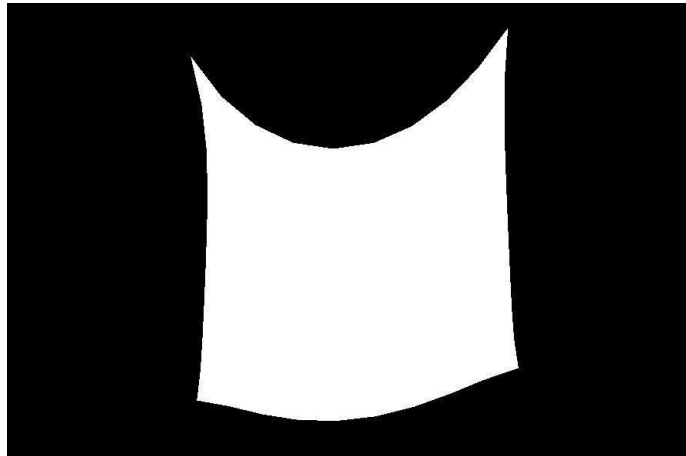


Figura 3.16: Superfície da simulação S_1 com $\zeta_{\alpha\beta} = 5$.

pudesse atingir uma geometria similar, ele deveria ser configurado como sendo composto de elastano e forças externas precisariam ser aplicadas para puxá-lo para baixo, contudo, não foi o que ocorreu nesse teste, em que somente a força da gravidade provocou esta deformação.

Finalmente, um último fator associado ao material do tecido que influencia a estabilidade numérica é a sua massa. Na simulação S_1 , por exemplo, o limite de 0.012 pode ser superado por 0.013 mantendo $\zeta_{\alpha\beta} = 100$, porém alterando sua massa de 10 para 12. Na verdade, existe uma relação de dependência entre os coeficientes do tensor métrico e sua massa no modelo deformável, no sentido que quanto mais pesado é o material maior devem ser os valores dos coeficientes do tensor métrico para garantir que não ocorram distensões não realistas nos tecidos. Assim, ao manter $\zeta_{\alpha\beta} = 100$, mas aumentar a massa, não se considera esta relação, fazendo com que o tensor métrico suporte um passo de tempo ligeiramente maior. Entretanto, as massas dos tecidos também seguem padrões específicos. Uma calça *jeans* de um tamanho convencional para adultos, por exemplo, apesar de ser mais pesada que vários tecidos, não pode pesar algo em torno de 5kg.

Desta forma, calibrando a massa e os coeficientes elásticos do material do tecido de forma condizente com o comportamento real, não é possível destacar diferenças no valor máximo do passo de tempo entre os métodos implementados. Viu-se que isto ocorre devido à imposição feita pela forma como o modelo é aproximado na solução numérica.

3.5.2 Eficiência Computacional

Na prática, viu-se que os passos de tempo máximos de cada método são iguais para cada simulação de teste, conforme explicado na subseção 3.5.1. Isto faz com que os tempos T_e obtidos sirvam apenas como uma informação adicional sobre o desempenho computacional, embora o número de iterações necessárias para o equilíbrio ainda seja um dado interessante para analisar qual método

avança com mais suavidade nos movimentos. Desta forma, a análise da eficiência computacional dos métodos implementados é feita primeiro com o tempo de processamento de uma iteração, para em seguida avaliar o número de iterações necessárias para o equilíbrio. Em relação a estes tempos, eles são multiplicados por 10000 iterações para exibi-los de uma forma mais clara em um gráfico de tempo por resolução espacial da malha, como mostrado na figura 3.17.

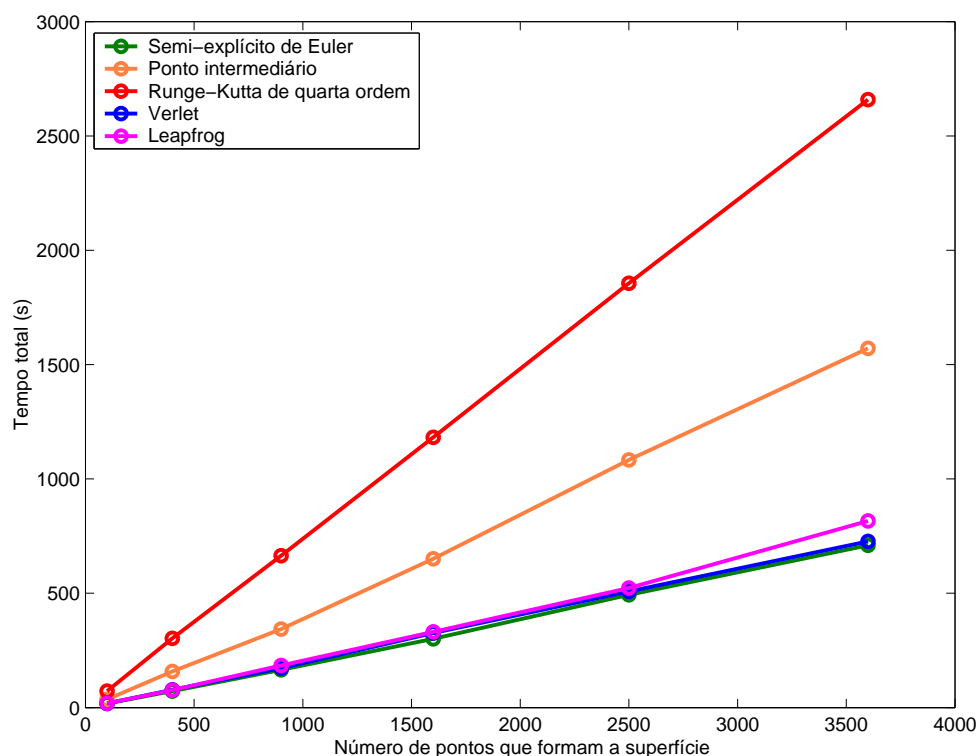


Figura 3.17: Tempos totais de processamento para executar 10000 iterações.

Com o gráfico da figura 3.17 nota-se que todos os tempos de processamento de cada método possuem um comportamento aproximadamente linear. Como em cada simulação de teste o número de restrições muda, existe uma pequena diferença de tempo para ser totalmente linear. Por exemplo, a simulação S_1 apresenta somente duas restrições de ponto fixo, enquanto a poltrona da simulação S_5 possui 360. Mesmo assim, a resolução espacial da malha é um fator predominante no tempo em relação ao número de restrições. Considerando como retas o resultado de cada método no gráfico, pode-se notar que a diferença entre eles é a inclinação da reta. O método de Runge-Kutta de quarta ordem, conforme previsto, é o que demanda maior tempo de processamento, afinal, ele tem três iterações da dinâmica interna do tecido dentro de cada iteração numérica. Em seguida, vem o método do ponto intermediário, que possui uma iteração adicional do modelo deformável em cada solução numérica. Depois destes dois métodos, os demais são muito próximos em termos de tempo de processamento, com leve vantagem para o método de Euler.

Em relação ao número de iterações necessárias para atingir o estado de equilíbrio, nota-se que os maiores são, em ordem, novamente os métodos de Runge-Kutta de quarta ordem e do ponto intermediário. Este resultado praticamente inviabiliza o uso destes dois métodos em conjunto com o modelo proposto por de Melo [4] quando uma das prioridades for o desempenho computacional, pois além deles demandarem um grande tempo de processamento ainda demoram mais para chegar aos estados de equilíbrio. Esta maior demora é explicada pela formulação dos métodos, que faz novas estimativas da velocidade no intervalo entre dois passos subseqüentes e com isso provocam deslocamentos mais suaves na superfície, demandando mais iterações. Em relação aos demais métodos, o método de Verlet praticamente acompanha o de Euler no número de iterações necessárias, enquanto no de *Leapfrog* este número é um pouco maior.

Desta forma, pode-se concluir que em termos de eficiência computacional, os métodos de Runge-Kutta de quarta ordem e o do ponto intermediário são bem inferiores do que os demais, principalmente por necessitarem de novas iterações do modelo de tecido. Já entre os outros três métodos: Euler, *Leapfrog* e Verlet, é possível notar que eles são muito próximos em termos de tempo de processamento e também no número de iterações necessárias para chegar ao equilíbrio. Contudo, em uma análise mais rigorosa, o método de *Leapfrog* fica um pouco abaixo em relação ao método de Euler e Verlet, os quais foram os mais eficientes computacionalmente neste estudo.

3.5.3 Qualidade das Imagens

A análise dos resultados visuais deve ser feita em termos da qualidade das imagens geradas nas simulações de teste para cada método. Contudo, como se viu na seção 3.4, todas as imagens obtidas nas simulações são muito similares visualmente em relação a proximidade com as fotografias da figura 3.9. Assim, as diferentes precisões numéricas dos métodos implementados não refletem grandes diferenças na geometria das superfícies no estado de equilíbrio. Dentro de cada grupo de imagens associadas a uma simulação de teste, porém gerados com diferentes métodos, pode-se notar mais diferenças visuais na simulação S_4 , da bandeira ao vento, quando feita à comparação, por exemplo, da bandeira gerada pelo método semi-explícito de Euler com a gerada pelo método do ponto intermediário. Neste caso é interessante lembrar que a bandeira é a única das simulações de teste em que o estado final de equilíbrio considerado não corresponde a um estado estacionário. Assim, é considerado como estado de equilíbrio o momento que a base inferior da bandeira fica perpendicular ao poste que ela está fixada, o que não necessariamente implica na mesma geometria da superfície apresentada para esta simulação, embora as duas geometrias possam ser equivalentes em alguns instantes posteriores, dado o comportamento oscilatório da bandeira.

As geometrias das superfícies mostradas nas imagens das outras simulações de teste são muito parecidas, sendo preciso reparar a formação de pequenas dobras para perceber as diferenças visuais.

Analisar estas pequenas diferenças e decidir qual apresenta melhor qualidade visual é uma tarefa subjetiva, pois enquanto uma pessoa pode achar que os resultados de um determinado método possuem mais qualidade por formarem mais dobras irregulares, outra pode achar melhor os resultados em que o tecido se deforma de maneira mais suave durante sua dinâmica. A dinâmica de tecidos é um fator importante para a análise da qualidade dos resultados visuais, porém ela não pode ser visualizada com imagens, sendo necessária a visualização das animações geradas em cada simulação de teste. Estas animações estão disponíveis em [37] e com elas pode-se concluir que os movimentos gerados pelos métodos de Runge-Kutta de quarta ordem e pelo método do ponto intermediário são mais suaves em relação aos outros três, embora isso não afete a qualidade das deformações produzidas na dinâmica. Desta forma, pode-se concluir que em termos de resultados visuais os métodos são todos equivalentes.

3.5.4 Escolha do Método Numérico

Para apresentar os resultados da simulação de tecidos com a abordagem proposta, este trabalho escolhe um dos métodos numéricos implementados para ser usado como padrão nos resultados do capítulo 5. Como nesta dissertação é proposta uma alternativa que visa agregar realismo e eficiência computacional para simular tecidos, os métodos de Runge-Kutta de quarta ordem e o do ponto intermediário são descartados pelo alto custo computacional que eles possuem. Com relação aos três métodos restantes, que são os métodos de Euler, *Leapfrog* e Verlet, viu-se que existe uma pequena diferença negativa do método *Leapfrog* tanto na performance quanto no número de iterações necessárias para atingir o equilíbrio nas simulações de teste em relação aos outros dois. Assim, o método de Euler, que é o método semi-implícito de Euler utilizado por Provot [2], e o método de Verlet, são os melhores métodos deste estudo que podem integrar ao longo do tempo o modelo proposto por de Melo [4].

Devido a sua boa eficiência e precisão numérica, o método de Verlet é muito usado atualmente para o desenvolvimento de aplicações que necessitam de bom desempenho, como em *softwares* em tempo real e jogos. O software gráfico 3D Studio Max utiliza este método de integração juntamente com um modelo de tecido massa-mola, visando gastar o menor tempo possível na dinâmica de tecidos, para que com o tratamento de colisões a simulação não fique muito pesada. Este método também é estável em relação ao uso de passos de tempo adaptativos, fundamental para as técnicas recentes de tratamento de colisões [24, 10] e muito usado em jogos para controlar dinamicamente a relação entre velocidade e precisão, de forma que quando a precisão não for uma prioridade, o passo de tempo pode ser incrementado para um movimento mais rápido, e caso contrário, ele pode ser reduzido para gerar maior precisão. Existe uma versão simples do método de Verlet, disponível em [34], que o adapta para o uso de passos de tempo variáveis usando uma taxa $\frac{\Delta t(t_n)}{\Delta t(t_{n-1})}$ do lado direito de 3.56.

Além disso, embora não se tenham grandes diferenças visuais entre os métodos de Euler e Verlet,

suas formulações teóricas apresentam uma diferença na precisão numérica, sendo Euler de precisão de primeira ordem enquanto Verlet é de terceira. Também em Verlet o erro das soluções diminui por aproximar a velocidade do próximo instante $t + \Delta t$ pelo intervalo $[t - \Delta t, t + \Delta t]$ ao invés de $[t, t + \Delta t]$, como na integração de Euler. Em relação ao consumo de memória, Verlet é superior ao método de Euler nesta implementação específica, pois utiliza um vetor de velocidades e dois de posições. Mesmo assim, levando-se em conta as diferenças, optou-se por utilizar o método de Verlet como o método padrão do simulador.

3.6 Considerações Finais

Neste capítulo foram apresentados os principais métodos explícitos que podem integrar ao longo do tempo o modelo de tecido proposto por de Melo [4]. Os métodos de Euler, do ponto intermediário, de Runge-Kutta de quarta ordem, *Leapfrog* e de Verlet tiveram suas formulações explicadas na seção 3.2 e foram adaptados para encontrar diferentes superfícies do tecido ao longo do tempo. Um resumo dos trabalhos relacionados com a simulação de tecidos e que utilizam os métodos explícitos discutidos foi apresentado na seção 3.3. Para investigar qual o melhor método que se adapta ao modelo de tecido [4], a seção 3.4 apresentou os resultados numéricos e visuais de cada método sobre um conjunto pré-determinado de simulações de testes. Com estes resultados foi possível analisar as diferenças entre os métodos em termos de estabilidade numérica, eficiência computacional e qualidade visual, conforme feito na seção 3.5.

Com a análise feita sobre os resultados visuais e numéricos foi possível constatar que as diferentes ordens de precisão numérica dos métodos explícitos não necessariamente implicam em efeitos visuais distintos. No caso da utilização destes métodos junto com um modelo baseado na mecânica de contínuos, como a formulação em diferenças finitas do modelo proposto por de Melo [4], não foi possível notar grandes diferenças visuais, contudo, os tempos de processamento mostraram uma grande diferença de eficiência computacional entre eles. Os métodos de Runge-Kutta de quarta ordem e o do ponto intermediário precisam de novas iterações da dinâmica do modelo em cada iteração de solução numérica, fazendo com que os tempos gastos sejam muito superiores em relação aos outros. Com relação aos métodos de Euler, *Leapfrog* e Verlet, todos eles apresentam desempenhos parecidos, porém em termos do número de iterações necessárias para atingir o estado de equilíbrio das simulações de teste, o método de *Leapfrog* é um pouco inferior ao de Euler e ao de Verlet.

Analisando as diferenças entre os métodos de Verlet e o semi-implícito de Euler, optou-se por utilizar o método de Verlet no restante dos resultados deste trabalho por ele possuir na teoria uma precisão numérica maior e por ser bastante usado atualmente em programas comerciais. Além disso, pensando na futura incorporação de algoritmos que façam o tratamento de colisões e auto-colisões, o

método de Verlet já possui algumas versões que atendem bem este propósito. Existe também outra versão do método de Verlet muito usada para produzir efeitos de dissipação em um ambiente, como forças de fricção do ar, que pode ser formulada a partir da equação 3.56 da seguinte maneira

$$\mathbf{r}_{k,l}(t + \Delta t) = (2 - f) * \mathbf{r}_{k,l}(t) - (1 - f)\mathbf{r}_{k,l}(t - \Delta t) + \mathbf{a}_{k,l}(t)\Delta t^2,$$

na qual f é um número entre 0 e 1 que representa a fração da velocidade que é perdida em cada atualização por algum tipo de força de fricção.

Com as soluções numéricas apresentadas, analisadas e, dentre elas, escolhida a mais apropriada para integrar o modelo de tecido proposto por de Melo [4], o capítulo 4 aborda todas as implementações feitas neste trabalho e no capítulo 5 são apresentados os resultados das simulações de tecidos usando o método de Verlet.

Capítulo 4

Implementação

Este capítulo apresenta as implementações feitas neste trabalho, que podem ser divididas na construção de uma interface lógica de programação e no projeto de uma interface gráfica para simular tecidos. Como este trabalho visa analisar a dinâmica de tecidos produzida com uma nova implementação que torne o modelo de tecidos selecionado mais eficiente computacionalmente, é fundamental reestruturar os códigos já existentes para se adequarem às necessidades dos métodos explícitos. Além disso, o desenvolvimento de uma interface lógica de programação robusta e modular permitirão que futuros incrementos, como algoritmos de tratamento de colisões, possam ser facilmente adicionados sem grandes modificações. Para tanto, usou-se um projeto orientado a objetos, separando o processo referente à dinâmica de tecidos do de visualização da simulação, conforme explicado na seção 4.1.

O simulador desenvolvido neste trabalho tem o objetivo de construir uma interface gráfica de fácil controle ao usuário, mas que permita criar animações complexas definindo apenas um pequeno conjunto de parâmetros de superfície e simulação. Os principais *softwares* gráficos comerciais, como o 3D Studio e Maya, possuem *plugins* com simuladores de tecidos acoplados, que são muitas vezes usados para produzir animações de excelente qualidade, entretanto, o manuseio nestes softwares requer uma certa experiência e familiaridade com vários conceitos de simulações físicas complicados até para um usuário acostumado a usar o *software* para outras tarefas. Assim, a interface gráfica proposta prioriza evidenciar uma das grandes vantagens em se usar o modelo de tecido proposto por de Melo [4], que é a intuitividade dos parâmetros para definir as propriedades físicas do material, bem como os termos de força externa e propriedades do ambiente, aumentando a usabilidade do simulador. Uma interface gráfica é desenvolvida para a plataforma Linux e outra para a Windows, as quais são detalhadas na seção 4.2, juntamente com exemplos de simulações.

4.1 Modelagem de dados

O primeiro passo da implementação computacional do simulador composto do modelo de tecido proposto por de Melo [4] com os métodos explícitos apresentados neste trabalho é a definição do fluxo de dados que produz a dinâmica de tecidos, como mostrado na subseção 4.1.1. Em seguida, este fluxo de dados é apresentado de maneira mais detalhada em forma de algoritmo na subseção 4.1.2, para então na subseção 4.1.3 mostrar a análise de requisitos. O último passo consiste em modelar os dados envolvidos no simulador para criar uma interface lógica, usando um projeto orientado a objetos. Nesta modelagem, o simulador proposto é separado em dois módulos principais: o módulo da dinâmica de tecido, que estabelece as novas posições dos pontos da superfície em cada iteração, e o módulo de visualização, que mostrará ao usuário a animação criada com os parâmetros entrados através de uma interface gráfica. Desta forma, definem-se duas classes, a classe *malha*, responsável pela dinâmica de tecidos, e a classe *render*, responsável pela visualização da simulação, que são explicadas nas subseções 4.1.4 e 4.1.5, respectivamente.

4.1.1 Fluxo de Dados

É possível descrever a dinâmica de tecidos em termos do fluxo de dados que acontece durante toda a simulação. Considerando que o usuário informa os dados iniciais ao simulador e obtém uma animação de saída, cada iteração feita pode servir como um quadro da animação. Convencionando i como um contador crescente que representa a iteração atual em que o sistema se encontra e T como o número total de iterações que o usuário especifica, o fluxograma da figura 4.1 ilustra a entrada, a saída, os dados, o teste de parada e os processos envolvidos no simulador.

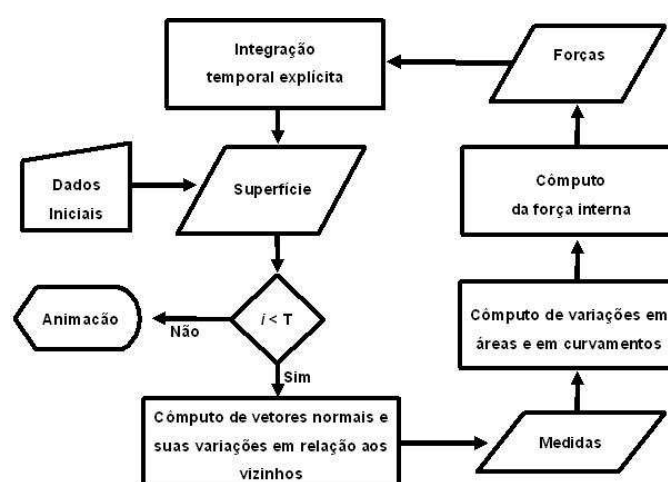


Figura 4.1: Fluxograma da dinâmica de tecidos neste trabalho.

Desta forma, o usuário informa um conjunto de dados para que o simulador construa a superfície inicial do tecido, que neste simulador é sempre um plano. Em seguida, os elementos da geometria diferencial são calculados com base nos pontos que formam a superfície, obtendo os vetores que formam o referencial local a cada ponto, como também seus tensores e sua área. Com as medidas geométricas calculadas, o processo seguinte consiste em calcular as variações da forma da superfície atual em relação à inicial, obtendo as medidas de deformação. Estas são usadas no cálculo das forças internas, que são balanceadas com as forças externas, definidas pelo usuário, na equação de equilíbrio do modelo, solucionada por um dos métodos de integração explícita apresentados neste trabalho. O processo final de uma iteração gera uma nova geometria da superfície do tecido. Este ciclo é feito até que o número de iterações feitas pelo sistema atinja o valor especificado pelo o usuário. A saída do sistema é uma animação formada pelas geometrias da superfície encontradas em cada iteração. Na prática, a construção da animação de saída é feita tomando-se iterações devidamente espaçadas.

4.1.2 Algoritmos

Conforme se frisou ao longo deste trabalho, a simulação de tecidos pode ser encarada como um modelo de tecido, definido por um modelo de superfície deformável adequado, em conjunto com uma integração temporal, que encontra diferentes geometrias da superfície ao longo do tempo. Assim, pode-se descrever o sistema dividindo-o em dois algoritmos: um para encontrar todos os elementos do modelo de tecido e obter as acelerações resultantes; e outro para obter a geometria da superfície em cada iteração a partir destas acelerações. O algoritmo 2 apresenta as instruções da função **Deforma**, responsável por receber os dados iniciais da simulação, calcular na ordem correta os elementos do modelo deformável e chamar a função que realiza a integração temporal explícita ao seu final [3]. Já o algoritmo 1 apresenta as instruções da função **Integra**, que corresponde ao método de Verlet.

Função Integra (var \mathbf{r}_{i-1} : **Ponto**[], var $\mathbf{v}_{i-1}, \mathbf{f}_{i-1}, \mathbf{N}^\alpha|_\alpha$: **Vetor**[], var $\varrho, \mu, \Delta t$: **Real**)

```

01  Var
02     $\mathbf{a}_i, \mathbf{v}_i, \mathbf{r}_i, \mathbf{r}_{i-1}, \mathbf{aux}$  : Vetor[];
03  Início
04    Para cada ponto  $(k, l)$  da malha discreta faça
05       $\mathbf{a}_i(k, l) \leftarrow \frac{\mathbf{f}_{i-1}(k, l) - \varrho \mathbf{v}_{i-1}(k, l) - \mathbf{N}^\alpha|_\alpha(k, l)}{\mu}$ ;
06       $\mathbf{aux}(k, l) \leftarrow \mathbf{r}_i(k, l)$ ;
07       $\mathbf{r}_i(k, l) \leftarrow 2 * \mathbf{r}_{i-1}(k, l) - \mathbf{r}_{i-1}(k, l) + \Delta t^2 \mathbf{a}_i(k, l)$ ;
08       $\mathbf{v}_i(k, l) \leftarrow \frac{\mathbf{r}_i(k, l) - \mathbf{r}_{i-1}(k, l)}{2 * \Delta t}$ ;
09       $\mathbf{r}_{i-1}(k, l) \leftarrow \mathbf{aux}$ ;
10    Fim para
11     $\mathbf{Integra} \leftarrow \mathbf{r}_i$ 
12  Fim

```

Algoritmo 1: Algoritmo de integração explícita.

Função Deforma (**var** $\mathbf{v}_{inicial}, \mathbf{a}_{inicial}, \mathbf{f}_{externas} : \mathbf{Vetor}[]$; **var** $\zeta_{\alpha\beta}, \xi_{\alpha\beta}, \phi_{\alpha\beta}, massa, \rho, \Delta t : \mathbf{Real}$;
var $comp_x, comp_y, comp_z : \mathbf{Real}$; **var** $m, n, passos : \mathbf{Inteiro}$)

```

01 Var
02    $\mathbf{r}, \mathbf{r}_{antigo}, \mathbf{v}, \mathbf{f}, \mathbf{EstadoDeformado} : \mathbf{Vetor}[]$ ;
03    $S_0, S, \mu, fator : \mathbf{Real}$ ;
04    $i : \mathbf{Inteiro}$ ;
05 Início
06   Calcule a área inicial  $S_0$  com base em  $comp_x, comp_y$  e  $comp_z$ ;
07   Discretize  $S_0$  usando  $m$  e  $n$  de forma a criar espaçamentos iguais  $\Delta x^1$  e  $\Delta x^2$ , e preencher o vetor  $\mathbf{r}$ ;
08   Para cada ponto  $(k, l)$  da malha discreta faça
09      $\mathbf{v} \leftarrow \mathbf{v}_{inicial}$ ;
10      $\mathbf{f} \leftarrow \mathbf{f}_{externas}$ ;
11     Calcule os vetores de base  $\mathbf{a}_\alpha$  e vetor normal  $\mathbf{n}$ ;
12     Calcule os valores iniciais dos tensores métrico  $A_{\alpha\beta}$ , métrico recíproco  $A^{\alpha\beta}$  e de curvatura  $B_{\alpha\beta}$ ;
13     Calcule os parâmetros do material  $\Phi^{\alpha\beta}, \Psi^{\alpha\beta}, \Theta^{\alpha\beta}$ , usando  $\zeta_{\alpha\beta}, \xi_{\alpha\beta}, \phi_{\alpha\beta}$  e  $A^{\alpha\beta}$ ;
14   Fim para
15    $i \leftarrow 0$ ;
16   Enquanto  $i$  for menor que  $passos$  faça
17     Determinar a área atual  $S$  com base nos pontos  $\mathbf{r}$ ;
18      $\mu \leftarrow \frac{massa}{S}$ ;
19      $fator \leftarrow \frac{S_0}{S}$ ;
20     Para cada ponto  $(k, l)$  da malha discreta faça
21       Calcule os vetores de base  $\mathbf{a}_\alpha$  e vetor normal  $\mathbf{n}$ ;
22       Calcule os valores atuais dos tensores métrico  $a_{\alpha\beta}$ , métrico recíproco  $a^{\alpha\beta}$  e de curvatura  $b_{\alpha\beta}$ ;
23       Calcule a medida de deformação  $\varepsilon_{\alpha\beta}$  usando o valor do tensor métrico inicial  $A_{\alpha\beta}$  e atual  $a_{\alpha\beta}$ ;
24       Calcule a medida de deformação  $\kappa_{\alpha\beta}$  usando o valor do tensor de curvatura inicial  $B_{\alpha\beta}$  e atual  $b_{\alpha\beta}$ ;
25       Calcule os símbolos de Christoffel  $\Gamma_{\alpha\beta}^\gamma$ ;
26       Calcule os termos de Weingarten  $b_\alpha^\beta$ ;
27       Calcule os componentes da força de curva  $\mathbf{N}^{\alpha\beta}$ ;
28       Calcule os componentes da força diretora  $\mathbf{M}^{\alpha\beta}$ ;
29       Calcule a força elástica  $\mathbf{N}^\alpha|_\alpha$ ;
30       Calcule a força externa resultante usando  $\mathbf{f}$  e  $\mathbf{a}_{inicial}$ ;
31     Fim para
32      $\mathbf{r}_{antigo} \leftarrow \mathbf{r}$ ;
33      $\mathbf{r} \leftarrow \text{Integra}(\mathbf{r}, \mathbf{v}, \mathbf{f}, \mathbf{N}^\alpha|_\alpha, \rho, \mu, \Delta t)$ ;
34      $\mathbf{EstadoDeformado}[i] \leftarrow \mathbf{r}$ ;
35     Para cada ponto  $(k, l)$  da malha discreta faça
36        $\mathbf{v} \leftarrow \frac{\mathbf{r} - \mathbf{r}_{antigo}}{\Delta t}$ ;
37     Fim para
38      $i \leftarrow i + 1$ ;
39   Fim enquanto
40    $\mathbf{Deforma} \leftarrow \mathbf{EstadoDeformado}$ ;
41 Fim

```

Algoritmo 2: Algoritmo de deformação.

Assim, o algoritmo 1 é chamado quantas vezes for preciso até que o número de iterações especificado pelo usuário seja atingido na função **Deforma** do algoritmo 2. Esta última função é responsável por calcular todos os elementos do modelo na ordem correta e passar os referentes à integração explícita para a função **Integra**. Elementos como os parâmetros de material são calculados para cada ponto da malha discreta somente na primeira iteração, de forma que o mesmo valor é usado nas demais até o fim. A cada iteração da dinâmica do simulador de tecidos é gerada uma geometria da superfície, representada pelo vetor de pontos **r**, o qual contém as posições tridimensionais de cada ponto que são guardadas no vetor chamado **EstadoDeformado**. Ao final do laço principal da função, este vetor de estados é o retorno da função, sendo usado para criar a animação.

4.1.3 Análise de Requisitos

Observando os algoritmos 1 e 2, pode-se avaliar quais são os dados constantes, os dados variáveis, as estruturas de dados e as funções necessárias para a implementação do simulador de tecidos. Um primeiro conjunto de dados que se pode destacar são os dados informados pelo usuário, de forma que todos eles precisam ser armazenados durante a simulação. Estes dados são listados a seguir por uma simples descrição, relacionando-os com os respectivos tipos de dados.

- Para definir o plano que forma a superfície inicial do tecido são necessários três valores reais, que correspondem aos comprimentos nos eixos x , y e z .
- Para armazenar os pontos tridimensionais da superfície é preciso um vetor cuja dimensão depende da resolução espacial, que é definida por dois valores inteiros e correspondem as dimensões da grade discreta bidimensional.
- A velocidade inicial é informada em um único vetor, de forma que todos os pontos terão esta velocidade na superfície inicial e, todos os pontos possuem uma velocidade atual que precisam ser armazenadas em um vetor de dimensão igual ao vetor de posições.
- A aceleração inicial funciona da mesma forma que a velocidade, também possuindo um único vetor inicial de entrada e outro de dimensão igual ao de posições para cada iteração.
- As forças externas que agem sobre o tecido podem ser especificadas sobre cada ponto da grade bidimensional, de forma que é preciso também de um vetor de dimensão igual aos demais.
- Para definir o material do tecido é preciso informar os coeficientes elásticos da superfície, que correspondem aos quatro valores reais de cada um dos três coeficientes.
- Também relacionado ao material do tecido, sua massa é um valor real.
- O coeficiente de amortecimento é um valor real.
- O passo de tempo que separa as iterações é um valor real, mas o número total de iterações é um número inteiro.

Além destes dados informados pelo usuário é preciso também modelar os dados variáveis que são usados para produzir a dinâmica no simulador. Estes dados são listados a seguir [3].

- Cada ponto da grade discreta possui dois vetores de base e um vetor normal, além de uma área da vizinhança local ao ponto, a qual é representada por um número real.
- Para cada ponto também estão definidos os tensores métrico, métrico recíproco e o de curvatura, de forma que cada um deles possui quatro valores reais.
- Com operações sobre estes tensores e os elementos que formam o referencial local a cada ponto, são obtidas as medidas de deformação, que são dois valores reais, os símbolos de Christoffel, que são seis valores reais, e os termos de Weingarten, que são quatro valores reais, por ponto.
- A força interna é calculada com base nos componentes da força de curva e força diretora, com estas grandezas representadas em um vetor de dimensão igual ao de posições.

É importante salientar que todos os vetores citados na modelagem desses dados são estruturas de dados com posições tridimensionais, como o da posição, velocidade e aceleração. Desta forma, os elementos utilizados na modelagem do simulador foram descritos e associados aos seus tipos, de maneira a incorporá-los na implementação. Feito isto, o próximo passo é verificar as estruturas e funções que podem ser definidas com base nas necessidades da dinâmica de tecidos. Observando a descrição dos requisitos do sistema, três estruturas podem ser destacadas: vetores, tensores e dados pertencentes a um único ponto da grade bidimensional. Quanto às funções, além das descritas nos algoritmos 1 e 2, é preciso de duas que calculem os valores aproximados das derivadas de primeira e segunda ordem do modelo, além de outras auxiliares ao cálculo dos elementos.

Utilizando a programação orientada a objetos para modelar os requisitos exibidos nesta subseção, é possível apontar duas classes que gerenciaram a dinâmica de tecidos. A classe *malha* é responsável pelo fluxo de dados da figura 4.1, com exceção da saída, de forma a incorporar a implementação dos algoritmos 1 e 2. A classe *render* é responsável por coletar as diferentes configurações das posições dos pontos do tecido e transformar estes dados em imagens e animações, que devem ser vistas através de uma interface gráfica desenvolvida para o usuário. Considerando todas as estruturas que precisam ser definidas como objetos, as variáveis são modeladas como atributos e as funções como métodos da classe que instancia um determinado objeto. É considerado de fundamental importância que estas duas classes sejam projetadas como módulos independentes, de forma que a parte de visualização seja separada da dinâmica para tornar o módulo referente à classe *malha* portátil, bem como permitir que a dinâmica interna de tecidos seja bem separada para a adição de um tratamento de colisões no futuro.

4.1.4 Classe *malha*

A classe *malha* é responsável por realizar a dinâmica de tecidos, de forma independente da classe *render*, de modo que seja possível portá-la entre diferentes plataformas e só mudar a forma de visualização. Além disto, esta classe provê uma interface lógica robusta para que futuras incorporações sobre ela, como um sistema de tratamento de colisões, sejam facilmente implementadas. Analisando os algoritmos 1 e 2, pode-se dizer que ambas as funções descritas neles formam a dinâmica de tecidos e, portanto, devem ser modeladas na classe *malha*, de maneira que seus atributos são os elementos do modelo de tecido e seus métodos as rotinas que calculam estes elementos.

O ponto de partida para os atributos da classe *malha* são os parâmetros de superfície e simulação fornecidos pelo usuário, que definem as características geométricas e físicas do objeto deformável. Os parâmetros geométricos consistem nas dimensões do objeto de acordo com o seu tipo, como por exemplo, três comprimentos nos planos x , y e z para definir um objeto do tipo plano, um raio e um comprimento para definir um cilindro, um raio maior e um menor para um torus e assim por diante. Apesar da implementação deste trabalho só considerar a deformação de planos em sua discretização espacial, o simulador proposto é planejado para dar suporte à criação de vários tipos de objetos, para que no futuro seja preciso mudar apenas a definição de vizinhanças em novos objetos.

Como o objeto deformável que representa a superfície inicial do tecido pode ser de vários tipos, é interessante modelá-lo como uma sub-classe de *malha*, chamada de *objeto*, de forma que cada tipo de objeto forme uma nova sub-classe desta última, facilitando a incorporações de novos tipos. A convergência de todos estes tipos ocorre no momento em que os pontos tridimensionais da superfície inicial são armazenados no vetor \mathbf{r} , independente da forma em que foram gerados. As dimensões e valores do vetor posição \mathbf{r} só podem ser determinadas com os parâmetros de simulação referentes à resolução $m \times n$ da malha discreta, pois a área da superfície é subdividida em intervalos iguais Δx^1 e Δx^2 em cada eixo. Desta forma, os atributos da classe *malha* são ilustrados na figura 4.2.

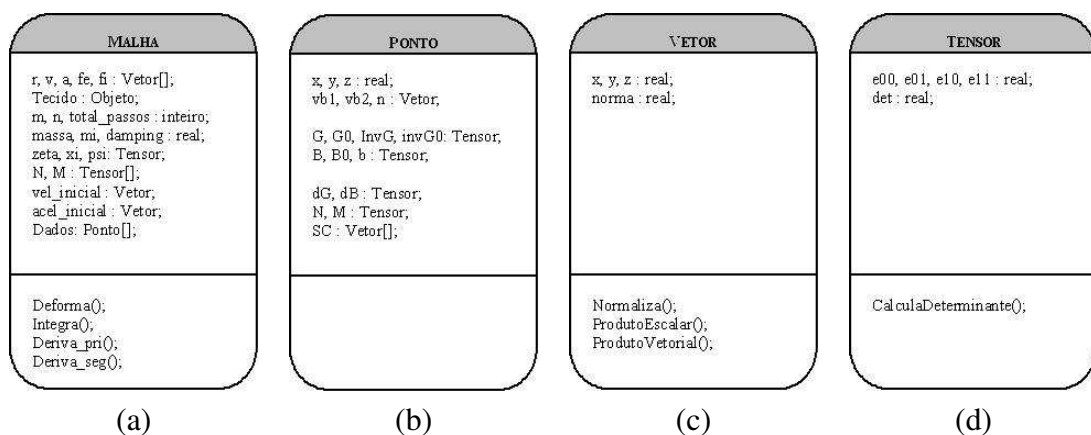


Figura 4.2: Classe *malha* e suas sub-classes.

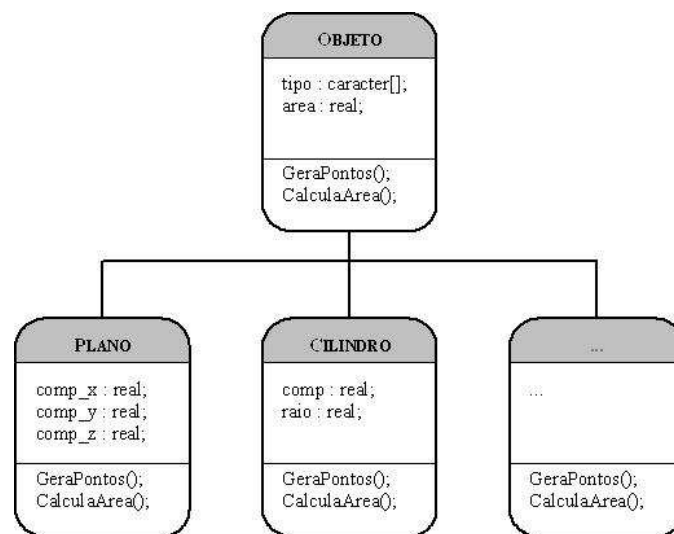


Figura 4.3: Modelagem da classe *objeto*.

A figura 4.2(a) ilustra a classe malha considerando os nomes de tipos minúsculos como os básicos das principais linguagens de programação e os maiúsculos como outras classes. A classe *Ponto*, ilustrada na figura 4.2(b) encapsula todos os dados do modelo de tecido que estão associados a cada ponto da malha discreta, por isso um vetor deste objeto chamado *Dados* faz parte da classe *malha*. As classes de suporte a tipos básicos, como vetores e tensores, são mostradas nas figuras 4.2(c) e 4.2(d), respectivamente. Por fim, a classe *objeto* é considerada como classe virtual, de forma que ela impõe que seus métodos sejam implementados pelas classes derivadas, como a classe *plano* usada no simulador, ou a classe *cilindro* como um exemplo de outros objetos que podem ser incorporados, conforme ilustrado a figura 4.3.

4.1.5 Classe *render*

A classe *render* é responsável pela visualização de imagens e animações produzidas pela dinâmica de tecidos. Como ela deve ser feita de forma independente da classe *malha*, a única comunicação entre elas é feita com a passagem dos pontos de cada estado do tecido, que corresponde as linhas 34 e 40 do algoritmo 2. Quando o usuário informa o número de iterações que serão feitas em uma simulação, junto com o atributo *intervalo*, é possível estabelecer quantos quadros a animação possuirá, que corresponde a divisão $quadros = \frac{número\ de\ iterações}{intervalo}$. Com o tempo total da animação, informada pelo usuário em segundos, e sabendo o número de quadros, é utilizado um temporizador que atualiza cada quadro após instantes de tempo iguais. Este tempo da animação não está vinculado ao montante de tempo simulado. O tempo simulado é obtido multiplicando o número de iterações pelo valor de Δt , contudo, este tempo pode ser visualizado em uma duração diferente, que corresponde ao tempo de

animação. Desta forma, a classe *render* possui como atributos uma lista de posições tridimensionais referentes a cada nova geometria da superfície do tecido e um valor inteiro que corresponde o intervalo no qual as iterações produzidas pela dinâmica de tecidos tornam-se quadros da animação gerada para a saída da simulação.

Além desses atributos, que são especificados pelo usuário, a classe *render* possui métodos que geram e gerenciam as animações produzidas nas simulações. A geração consiste em armazenar as geometrias da superfície encontradas nos intervalos determinados e desenhá-las em uma cena tridimensional. Para este desenho escolheu-se usar a biblioteca gráfica OpenGL. Assim, cada geometria é desenhada em uma lista de visualização que é chamada durante a execução da animação. Contudo, para aumentar a usabilidade do simulador, é interessante gerar estas animações de duas formas: uma de maneira dinâmica, em que os quadros são exibidos à medida que são gerados, conhecido como modo *on-the-fly*; e de maneira estática, em que todos os quadros são gerados primeiro para depois a animação ser visualizada. Nas duas formas é fundamental construir rotinas para carregar os quadros em *display lists* para a visualização e também rotinas para salvá-los em arquivo para que não seja preciso rodar novamente uma simulação somente para assistir a animação gerada.

Existem também recursos auxiliares ao processo de animação que são interessantes de serem incorporados na classe *render*. A visualização da grade da superfície do tecido, conhecida como modo de visualização *wireframe*, é muito importante principalmente quando o intuito é localizar visualmente certo ponto da malha discreta para fins de depuração. A aplicação de texturas e fontes de luz na cena também contribuem para o aumento da qualidade da animação, incrementando também o realismo na visualização do tecido. Rotinas de movimentação da cena, como as de translação e rotação, bem como as de escala, também são consideradas. A figura 4.4 ilustra os principais atributos e métodos da classe *render*.

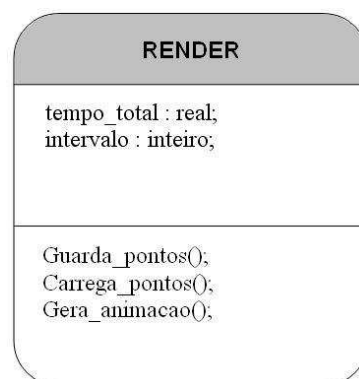


Figura 4.4: Modelagem da classe *render*.

4.2 Interface Gráfica

A interface gráfica do simulador de tecidos foi desenvolvida com a linguagem de programação C++ em conjunto com a biblioteca gráfica OpenGL. A implementação do sistema foi feita para as plataformas Linux, detalhada na subseção 4.2.1, e também para plataforma Windows, descrita na subseção 4.2.2. Em ambas as plataformas seguiram-se às exigências analisadas na seção 4.1, de maneira a separar a dinâmica de tecidos da visualização de imagens e animações. Por fim, a subseção 4.2.3 dá exemplos de como utilizar o simulador para obter alguns resultados básicos.

4.2.1 Plataforma Linux

A interface gráfica na plataforma Linux foi feita utilizando o editor Qt Designer 3.3 e o compilador g++. Ela foi desenvolvida sobre uma janela principal que contém um componente com quatro abas, sendo elas divididas em grupos de diferentes informações, como mostrado na figura 4.6(a). É nesta janela que todas as rotinas de entrada e saída das simulações de tecidos são feitas. O único botão do menu principal tem como tarefa gerenciar os arquivos que contém os dados de entrada e salvar novas configurações em arquivo. Este arquivo de entrada é um arquivo texto com os valores de definição dos parâmetros da simulação e ele, após carregado, preenche a maioria dos campos dentro das quatro abas da janela principal. A figura 4.5 mostra um exemplo deste arquivo, que corresponde à simulação S_1 do capítulo 3. Pode-se também preencher todos os campos na interface e, ao salvar os valores preenchidos, é criado um arquivo texto com o mesmo formato dos dados. Na aba referente aos dados de simulação encontram-se as informações sobre o passo de tempo, tempo de animação, número de iterações e intervalo considerado entre as iterações para armazenar os quadros da animação. Nesta aba o usuário deve selecionar o tipo de saída das imagens, o método de integração utilizado e qual tipo de visualização da animação ele deseja.

```
% Este script foi criado automaticamente pelo Desmo 1.0
delta_t = 0.001
n_passos = 10000
pula = 50
tempo_animacao = 10
grid : 10 10
aceleracao : x = 0.0 y = -9.8 z = 0.0
velocidade : x = 0.0 y = 0.0 z = 0.0
eta : 100.0 100.0 100.0 100.0
psi : 0.3 0.3 0.3 0.3
psi : 0.001 0.001 0.001 0.001
massa = 10.0
gama = 0.2
obj_deformavel : plano 10.0 0.0 10.0
n_restricoes = 2
restricoes = {
0 0 ponto_fixo inicio=0 fim=20001 x=0 y=0 z=0
0 9 ponto_fixo inicio=0 fim=20001 x=0 y=0 z=0
}
```

Figura 4.5: Exemplo de um arquivo de entrada com todos os dados de uma simulação.

A aba da figura 4.6(b) é referente aos valores do ambiente em que a superfície se encontra. Nesta aba ocorre a definição da aceleração do ambiente, da velocidade inicial dos pontos da superfície, do coeficiente de amortecimento expresso pelo rótulo Gama e também do módulo e direção da força fluido. É interessante lembrar que alguns parâmetros são opcionais, como a aceleração do ambiente, velocidade inicial, força fluido e outros. A aba seguinte é referente aos dados da malha discreta que forma o tecido, em que se definem as resoluções espaciais da grade e os parâmetros do material do tecido, como sua massa e seus coeficientes elásticos, como mostra a figura 4.6(c). Nesta figura, existe uma caixa de opção sobre o tipo de material do tecido, porém sua função é apenas definir a textura utilizada sobre o tecido na visualização. A última aba é referente aos dados do objeto deformável, que corresponde à superfície do tecido. Nesta aba pode-se especificar as dimensões do plano inicial a ser deformado, em termos das coordenadas dos eixos, conforme ilustra a figura 4.6(d). Nesta aba há também o botão que abre a janela das restrições, mostrado na figura 4.7. O botão com o rótulo fronteiras é usado atualmente somente para depurações, assim como o botão de rótulo Gera Dados.

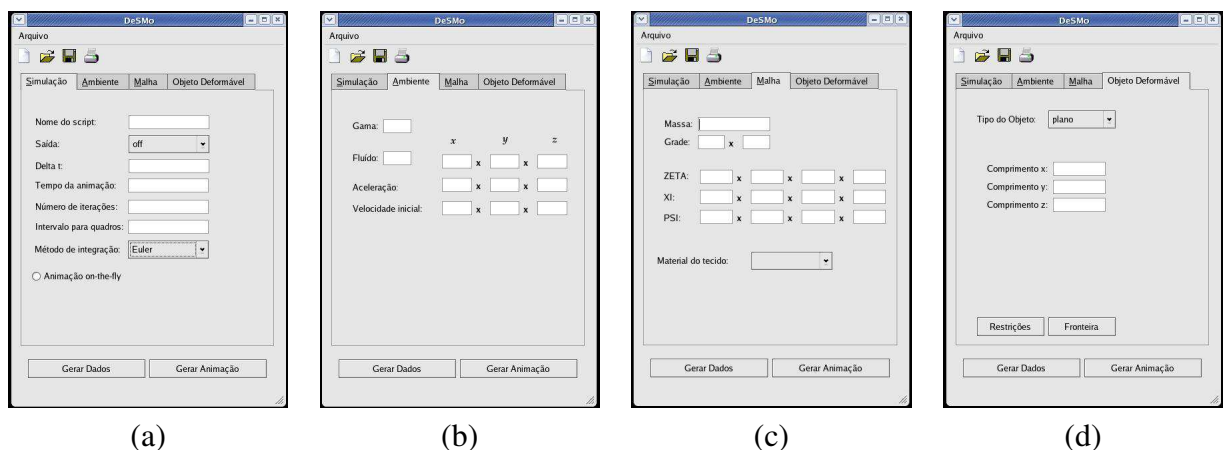


Figura 4.6: Interface gráfica para a plataforma Linux.



Figura 4.7: Janela das restrições da simulação.

Com todos os dados da janela principal preenchidos ou carregados, a simulação pode ser processada acionando o botão com o rótulo Gerar Animação. Ao apertar este botão a dinâmica de tecidos começa a ser calculada e a animação gerada é exibida em uma nova janela que contém um contexto de OpenGL. Este contexto é criado somente com a superfície inicial do tecido, porém apertando teclas de atalho é possível visualizar informações sobre o quadro que está sendo visualizado, conforme ilustrado na figura 4.8(a). O movimento do tecido nesta janela é gerenciado pelos botões do *mouse*. O botão esquerdo alterna entre os modos de reprodução ou pausa da animação, enquanto o direito reinicia a partir do primeiro quadro. À medida que a animação é reproduzida pode-se visualizar mais informações, como feito na figura 4.8(b), ou usar os recursos auxiliares, como adicionar e retirar iluminação, texturas, preenchimento das faces, ou realizar operações de escala e movimentação na superfície do tecido.

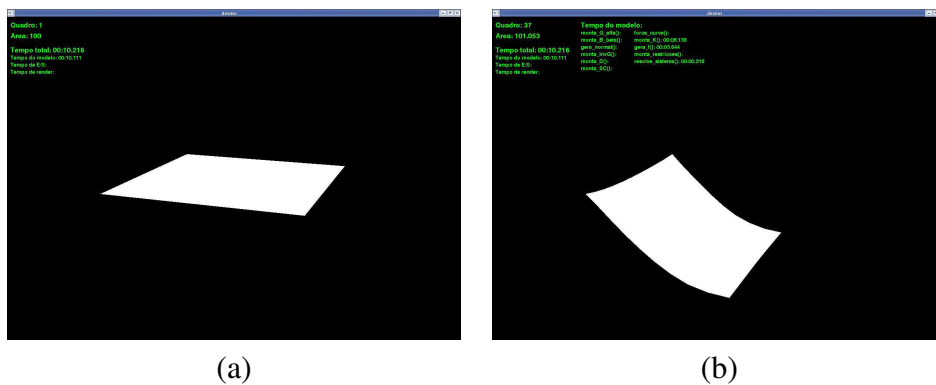
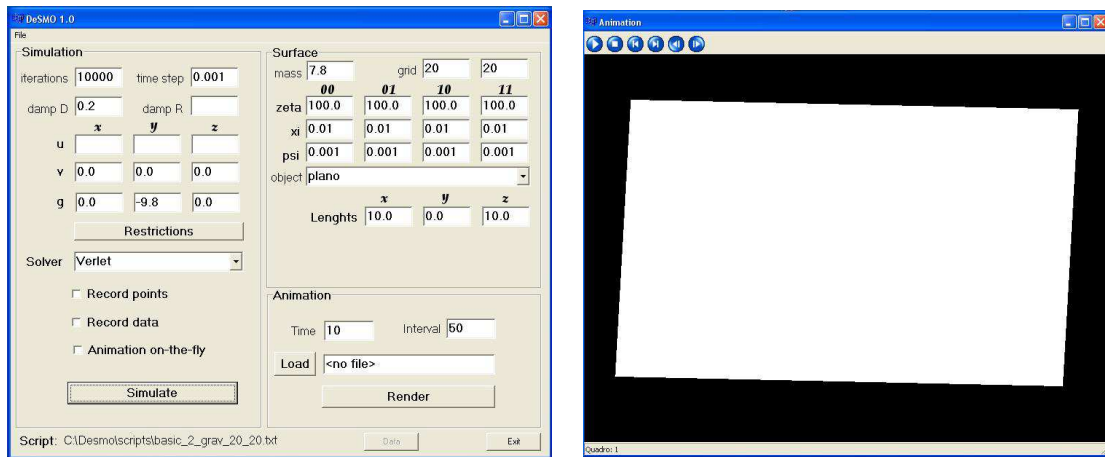


Figura 4.8: Janela de visualização da animação gerada.

4.2.2 Plataforma Windows

A interface gráfica desenvolvida para a plataforma Windows possui um formato diferente, dispondo todas as informações da simulação na janela principal, sem o uso de abas, como mostra a figura 4.9(a). Esta interface foi feita no compilador C++ Borland Builder 6, usando a biblioteca gráfica *vcl* do Windows. No começo deste trabalho apenas a interface para a plataforma Linux estava disponível, porém para testar a portabilidade do módulo malha e também para tornar o simulador mais acessível, a interface gráfica foi feita para o sistema Windows com os rótulos em inglês. De forma similar à desenvolvida para o Linux, o botão *File* na barra do menu gerencia os arquivos de entrada e saída que preenchem a maioria dos parâmetros da simulação. Dentro desta janela principal os dados são divididos entre os grupos de simulação, de superfície e de animação. Os parâmetros de animação são o tempo de animação e o intervalo entre interações usado para armazenar os quadros. Os parâmetros de superfície são todos os parâmetros relacionados com a superfície do tecido, que são equivalentes

aos das abas de malha e objeto deformável na interface gráfica do Linux. Todos os demais parâmetros são considerados parâmetros de simulação na interface do Windows.



(a)

(b)

Figura 4.9: Janelas de (a) simulação; e (b) visualização da animação gerada.

Um procedimento recomendado para simular tecidos que possuam muitos pontos é utilizar o modo de animação *on-the-fly* até calibrar os parâmetros da simulação de maneira estável para garantir que não ocorra uma situação de instabilidade numérica. Quando a simulação estiver corretamente definida, pode-se armazenar arquivos contendo todos os seus pontos e dados para assistir a animação sempre que se desejar sem necessitar gerar os quadros novamente. Contudo, no modo *on-the-fly*, no qual os quadros da animação são calculados e armazenados em listas de visualização na memória volátil do computador, pode-se salvar individualmente em arquivo cada quadro separadamente, escolhendo-se sua extensão. Nesta implementação também estão disponíveis os mesmos recursos auxiliares acionados por teclas de atalho, que controlam as rotinas de iluminação, preenchimento, texturização, rotação, escala e translação.

A janela de visualização da animação ilustrada na figura 4.9(b) difere um pouco da janela feita para a plataforma Linux, sendo ela de usabilidade mais convencional em relação à reprodução de animações. Utilizando botões de reprodução, pausa, retorno ao início, avanço ao final e navegação quadro a quadro, os quais correspondem, respectivamente, aos seis botões no alto da janela da figura 4.9(b), o fluxo da animação pode ser facilmente controlado. O quadro atual da animação é mostrado no rodapé da janela, enquanto os valores de qualquer elemento do modelo, como a área do tecido, podem ser obtidos ligando a opção rotulada como *Record data*. Por fim, o tempo de processamento gasto em uma simulação é exibido em uma mensagem de alerta ao usuário no final da execução das iterações solicitadas.

4.2.3 Exemplos de Simulação

Nesta subseção são dados alguns exemplos de simulações de tecidos para mostrar a usabilidade do simulador implementado. Para tanto, são explicadas como foram feitas algumas das simulações de teste do capítulo 3, evidenciando os parâmetros de simulação. A simulação S_1 , por exemplo, cujos dados estão ilustrados na figura 4.5, é um tecido pequeno que cai de acordo com a força da gravidade preso por suas duas extremidades superiores. Esta simulação é feita com uma superfície de resolução espacial 10×10 , com massa 10 e coeficientes $\zeta_{\alpha\beta} = 100$, $\xi_{\alpha\beta} = 0.3$ e $\theta_{\alpha\beta} = 0.001$. A força da gravidade $\mathbf{g} = (0, -9.8, 0)$ age sobre todos da superfície. Suas únicas restrições são dois pontos fixos que atuam desde a primeira iteração da dinâmica até a iteração 20000, como mostra a figura 4.10(a). O estado final desta simulação está ilustrado na figura 4.11(a).

Fazendo agora uma pequena alteração para mostrar como funciona a definição das restrições no simulador, deseja-se que o tecido caia de forma semelhante durante a iteração 1 até a 10000 e, a partir da iteração 10001 até 15000, só o ponto da direita seja fixo, enquanto o da esquerda possa se movimentar, mas somente ao longo dos eixo xz , ou seja, continuando fixo em y . Da iteração 15001 até 20000, o ponto da esquerda, que até chegar neste intervalo já sofreu um deslocamento, permanecerá fixo em todos os eixos, liberando o ponto fixo da direita, que estava fixo desde a iteração 1 até a 15000. A definição das novas restrições é mostrada na figura 4.10(b). Já os resultados da simulação estão ilustrados na figura 4.11, associando as imagens com as iterações.

Restrictions									
	U	V	Tipo	Inicio	Fim	X	Y	Z	
0	0	0	ponto_fixo	0	20001	0	0	0	
1	0	9	ponto_fixo	0	20001	0	0	0	

(a)

Restrictions									
	U	V	Tipo	Inicio	Fim	X	Y	Z	
0	0	0	ponto_fixo	0	10000	0	0	0	
1	0	9	ponto_fixo	0	15000	0	0	0	
2	0	0	ponto_fixo	10001	15000	1	0	1	
3	0	0	ponto_fixo	15001	20000	0	0	0	

(b)

Figura 4.10: Janela de restrições.

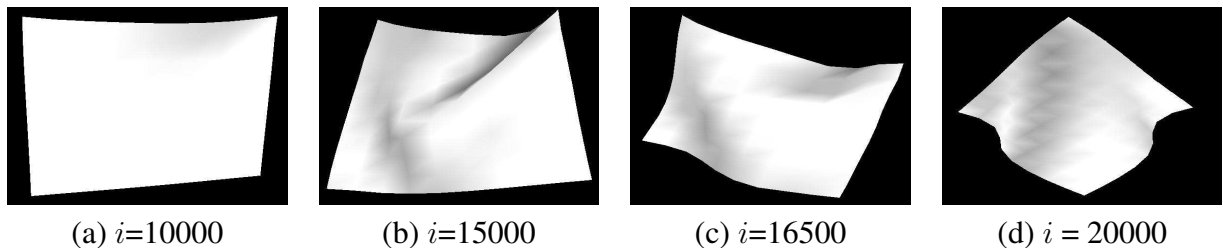
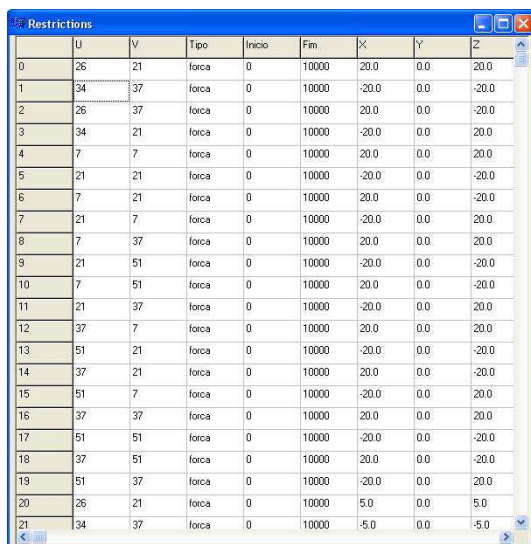


Figura 4.11: Imagens de simulação.

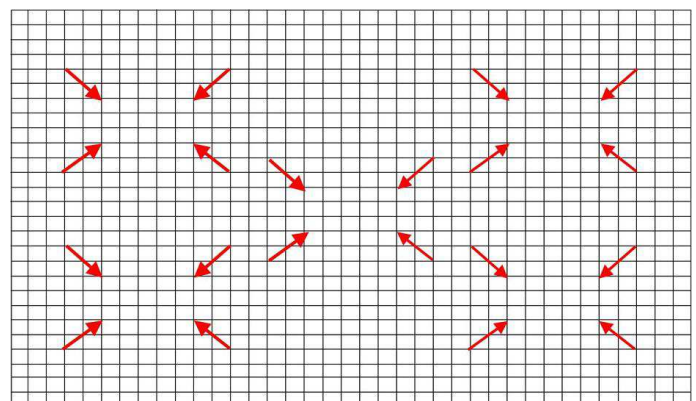
Desta forma, o controle das restrições no simulador implementado é feito com a definição de poucos parâmetros, obtendo efeitos interessantes por simplesmente deixar a força da gravidade agir com a alternância de pontos fixos em diferentes instantes e eixos. A simulação S_2 do capítulo 3 possui as restrições definidas de forma parecida com as de S_1 , com a única diferença de se ter um ponto fixo ao invés de dois, fazendo com que a força da gravidade sobre o tecido suspenso forme a geometria de uma capa. Como esta implementação não possui algoritmos que tratem colisões, os caimentos de tecidos sobre objetos não podem ser reproduzidos de forma correta sob o ponto de vista físico. Entretanto, estes efeitos podem ser aproximados com a definição de pontos fixos que conforme seus parâmetros, como em quais eixos são fixos e sua duração, formam a geometria do objeto de colisão. Isto foi feito nas simulações S_3 , com o caimento de um tecido sobre um banco de praça, e em S_5 , com a forma de um tecido sobre uma poltrona, mostradas nas figuras 3.10(c) e 3.10(e).

É possível também definir como restrições da simulação forças pontuais, as quais são aplicadas separadamente a cada ponto da malha discreta com um módulo e sentido determinado. Estas forças são muito úteis para dar um acabamento ao tecido simulando pequenos ajustes que ocorrem com a mão na realidade, ou até modelando totalmente o comportamento do tecido. A simulação S_6 , por exemplo, consiste de uma superfície em um ambiente sem gravidade sobre a qual são aplicadas forças de compressão tangenciais ao seu plano inicial (figura 3.10(f)). Ao todo, vinte pontos da malha sofrem forças pontuais, aplicadas até a iteração 10000 com um módulo apropriado para evitar auto-colisões. Em seguida, estes mesmos pontos sofrem uma força menor somente para manter as dobras já formadas. A figura 4.12(a) mostra algumas das quarentas restrições e 4.12(b) ilustra a localização e o sentido das forças na malha discreta.



	U	V	Tipo	Inicio	Fim	X	Y	Z
0	26	21	força	0	10000	20.0	0.0	20.0
1	34	37	força	0	10000	-20.0	0.0	-20.0
2	26	37	força	0	10000	20.0	0.0	-20.0
3	34	21	força	0	10000	-20.0	0.0	20.0
4	7	7	força	0	10000	20.0	0.0	20.0
5	21	21	força	0	10000	-20.0	0.0	-20.0
6	7	21	força	0	10000	20.0	0.0	-20.0
7	21	7	força	0	10000	-20.0	0.0	20.0
8	7	37	força	0	10000	20.0	0.0	20.0
9	21	51	força	0	10000	-20.0	0.0	-20.0
10	7	51	força	0	10000	20.0	0.0	-20.0
11	21	37	força	0	10000	-20.0	0.0	20.0
12	37	7	força	0	10000	20.0	0.0	20.0
13	51	21	força	0	10000	-20.0	0.0	-20.0
14	37	21	força	0	10000	20.0	0.0	-20.0
15	51	7	força	0	10000	-20.0	0.0	20.0
16	37	37	força	0	10000	20.0	0.0	20.0
17	51	51	força	0	10000	-20.0	0.0	-20.0
18	37	51	força	0	10000	20.0	0.0	-20.0
19	51	37	força	0	10000	-20.0	0.0	20.0
20	26	21	força	0	10000	5.0	0.0	5.0
21	34	37	força	0	10000	-5.0	0.0	-5.0

(a)



(b)

Figura 4.12: Definição e ilustração das forças pontuais.

Naturalmente, os dois tipos de restrições, que são os pontos fixos e as forças pontuais, podem ser utilizados em uma simulação ao mesmo tempo. Em relação às suas definições, a diferença ocorre somente na atribuição dos valores para os parâmetros X , Y e Z . Para pontos fixos, o valor 1 em uma destas coordenadas quer dizer que o ponto pode andar naquele eixo, ou seja, se um ponto tiver $X=Y=Z=1$ a restrição não terá validade, pois o ponto se deslocará nos três eixos, enquanto $X=Y=1$ e $Z=0$ indica que o ponto só pode andar no plano xy , como foi feito na imagem da figura 4.11(b). Já quanto às forças pontuais, os valores são definidos com o módulo das forças em cada eixo, com o sinal determinando o sentido da força. Por exemplo, as forças da figura 4.12(b) que estão no sentido da esquerda para direita têm os módulos no eixo x positivos, conforme as definições da figura 4.12(a).

Contudo, além das forças pontuais e da gravidade atuando sobre pontos fixos, é possível aplicar a força fluido para deformar o tecido. Esta força é definida com uma constante de viscosidade do fluido e com seu módulo e direção. A bandeira da simulação S_4 foi feita usando este tipo de força, que provoca um comportamento oscilatório, como se o tecido mergulhasse em um líquido como a água (figura 3.10(d)). Neste caso, seria usado o coeficiente de viscosidade da água e o módulo e força do fluido seria equivalente a uma correnteza dentro da água. Para a simulação S_4 foi utilizado um coeficiente de viscosidade de $\varrho_R = 0.032$ e a direção de fluido $\mathbf{u} = (100, 0, 0)$, com a superfície inicial posicionada no plano yz . Estes valores são então calculados com a velocidade \mathbf{v} e o vetor normal \mathbf{n} de cada ponto da superfície, resultando em uma força \mathbf{F}_R tal que

$$\mathbf{F}_R = \varrho_R(\mathbf{n} \cdot (\mathbf{u} - \mathbf{v})\mathbf{n}.$$

O coeficiente de viscosidade do fluido ϱ_R está rotulado como *damp R* na interface para Windows ilustrada na figura 4.9(a). Este campo só é preenchido quando se desejar este efeito oscilatório no tecido, que no caso da bandeira foi feito para aproximar o vento. O coeficiente ao lado deste na interface, rotulado de *damp D*, é o coeficiente de amortecimento do meio ϱ , que aparece na equação 1.1. Ele é multiplicado pela velocidade atual de cada ponto da superfície, amortecendo sua energia potencial interna. Esta parcela corresponde ao montante de força acumulada na superfície que é dissipada pelo meio, sendo fundamental para produzir efeitos apropriados. O efeito visual deste coeficiente é ilustrado na figura 4.13. Para todas as simulações mostradas nesta dissertação, tanto as de teste quanto as dos resultados do capítulo 5, foi considerado $\varrho_D = 0.2$.

Por fim, os demais parâmetros referentes ao ambiente no qual a superfície do tecido se encontra, como a aceleração e a velocidade inicial dos pontos, são especificados com a definição de vetores na janela principal do simulador e estes valores são atribuídos para cada ponto da superfície, com a diferença que a aceleração é somada para cada ponto em cada iteração, enquanto os valores da velocidade inicial são usados somente na primeira iteração, sendo ela atualizada depois em função da aceleração resultante obtida pela equação de equilíbrio.

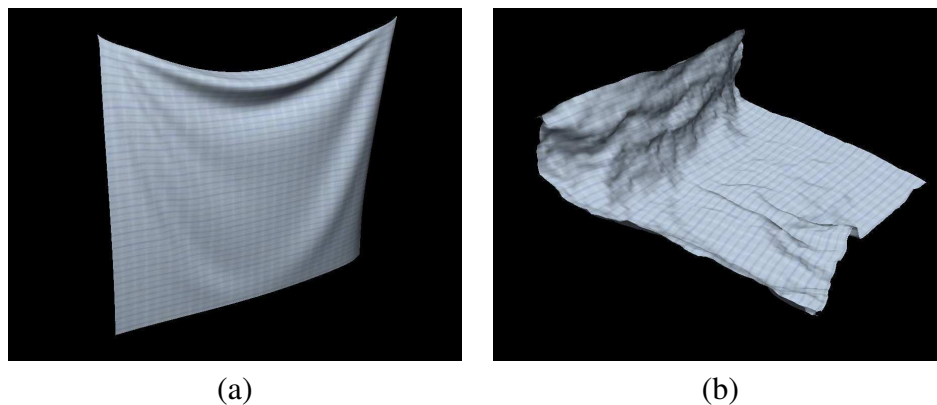


Figura 4.13: Efeitos do coeficiente ϱ_D : (a) com ele e (b) sem ele.

4.3 Considerações Finais

Neste capítulo foi apresentada a implementação da proposta deste trabalho, que consiste em integrar o modelo de tecido proposto por de Melo [4] ao longo do tempo utilizando métodos explícitos. Para tanto, primeiro a modelagem dos dados envolvidos foi discutida na seção 4.1, destacando o fluxo de dados existente na dinâmica de tecidos. Esta parte inicial da implementação diz respeito à construção de uma interface lógica que visa separar a simulação de tecidos em dois módulos independentes, um para produzir a dinâmica do tecido, chamado de *malha*, e outro para gerenciar a visualização das animações produzidas, chamado de *render*. Estes dois módulos também foram explicados nesta mesma seção. Em seguida, a seção 4.2 apresentou a interface gráfica do simulador desenvolvido para as plataformas Linux e Windows, fornecendo alguns exemplos de simulações de tecidos nesta mesma seção para mostrar a usabilidade do *software*.

Contudo, este capítulo apresentou a parte implementacional deste trabalho usando alguns recursos de projeto orientado a objetos, deixando as demais informações disponíveis em [38], como os códigos fonte e relatórios técnicos. Além disso, a apresentação da interface gráfica permitiu a identificação dos parâmetros de simulação que são manipulados para obter distintos comportamentos de tecidos, como nos exemplos de simulação apresentados neste capítulo, que são muito importantes para entender como os resultados do capítulo 5 foram gerados. O caimento de tecidos sobre mesas, por exemplo, pode ser feito com o simulador desenvolvido mesmo sem que ele possua algoritmos que façam o tratamento de colisões. Este tipo de simulação pode ser feita com a definição de vários pontos fixos, nos três eixos, da parte central da malha discreta, formando um quadrado ou um círculo. No capítulo 5 são mostradas imagens de animações de caimento e de outros tipos de simulação.

Capítulo 5

Resultados

Este capítulo apresenta os resultados do simulador implementado, destacando as o realismo nas imagens das animações geradas, a precisão física e o desempenho computacional. O objetivo é validar praticamente o modelo proposto por Wu e de Melo [4], mostrando os resultados visuais das principais características do modelo com simulações geradas em uma complexidade computacional linear. Com as imagens de animações é possível constatar o realismo alcançado, enquanto a análise do tempo computacional das simulações mostra o desempenho. A precisão física é evidenciada de acordo com a resposta dos diferentes tecidos às forças externas que são definidas como restrições das simulações e também com seqüências de imagens que tentam mostrar a dinâmica de tecidos. Além disso, o controle intuitivo dos parâmetros do material do tecido faz com que se obtenha com facilidade diversos resultados de diferentes materiais simulados sobre as mesmas restrições.

Contudo, antes de mostrar efetivamente os melhores resultados visuais, a seção 5.1 aborda dois procedimentos empíricos de calibração de valores usados na produção de simulações. A primeira delas refere-se à especificação dos parâmetros de material para atingir comportamentos específicos, como, por exemplo, valores que diferem um tecido de *jeans* de um de algodão. Já a segunda calibração é relativa às forças externas que são definidas como restrições pelo usuário, de forma que elas podem alterar o estado do tecido. Como este simulador não possui o tratamento de colisões, a correta manipulação destas forças é fundamental para a obtenção dos resultados desejados.

Os resultados são apresentados na seção 5.2 e divididos em três subseções. Elas correspondem aos resultados visuais obtidos com simulações que utilizam deformações bem conhecidas, como: *bucklings*, destacando a formação de dobras sob ação de forças externas; *drapings*, mostrando caimentos de tecidos sobre objetos; e *hangings*, a suspensão de tecidos por alguns pontos sob ação da gravidade. A seção 5.3 faz análises sobre os resultados destacando o realismo obtido através da diversidade de dobras e também a precisão física analisando a dinâmica dos tecidos nas simulações com seqüências de imagens ao longo do tempo. A seção 5.4 tece comentários finais.

5.1 Metodologia

Antes de exibir os resultados do simulador de tecidos é preciso salientar a forma em que um determinado comportamento de tecido é atingido. Para simular um material particular que compõe um tecido é necessário informar sua massa e seus coeficientes elásticos, sendo estes valores manipulados para produzir diferentes tipos de reação a forças externas. Estas reações estão relacionadas com a resistência ao curvamento e esticamento e a facilidade de formação de dobras, que são os parâmetros de material do tecido. A manipulação destes parâmetros para se obter diferentes materiais é explicada na subseção 5.1.1. Já as forças externas são forças que causam a deformação da superfície do tecido, sendo elas definidas pelo usuário da mesma forma que os parâmetros de material. Estas forças são especificadas com base no tipo de deformação que se deseja obter e dependem muito do material do tecido. A manipulação das forças externas é detalhada na subseção 5.1.2.

5.1.1 Calibração de Parâmetros de Material

O tipo de material que forma um tecido é definido por sua massa e por seus coeficientes elásticos $\zeta_{\alpha\beta}$, $\xi_{\alpha\beta}$ e $\phi_{\alpha\beta}$. Estes últimos valores são usados para obter os parâmetros de tecido $\Phi_{\alpha\beta}$, $\Psi_{\alpha\beta}$ e $\Theta_{\alpha\beta}$, que são calculados uma única vez na primeira iteração da dinâmica para todos os pontos da malha discreta, enquanto a massa é usada no cálculo da densidade de massa μ com base na área da superfície. Enquanto a massa do tecido é um valor constante, a densidade de massa é um valor variável em cada iteração, pois a superfície pode sofrer mudanças de área, ocasionadas por compressões ou distensões, ao longo do tempo. Os coeficientes elásticos são valores constantes que associam o material do tecido a determinadas características geométricas, como a resistência à mudança de curvatura, resistência à mudança de métrica e facilidade de formação de dobras. Desta forma, o usuário define o material de um tecido informando ao simulador sua massa e seus coeficientes elásticos.

O coeficiente elástico ζ afeta predominante a métrica da superfície, de forma que baixos valores indicam uma facilidade em se esticar, enquanto altos representam uma grande resistência a distensões, que é o comportamento mais adequado para tecidos. Já o coeficiente elástico ξ influi na curvatura da superfície, de modo que valores baixos determinam facilidade para se curvar e altos dificuldade. O parâmetro ϕ , chamado de fator de *buckling*, estabelece a facilidade do material em criar dobras, de forma que valores pequenos apontam pequeno acoplamento entre variações de área e de curvatura e valores altos grande acoplamento. Este último parâmetro pode ser encarado como o quanto a métrica e a curvatura se relacionam na produção de deformações para fora do plano. Para tornar mais claro o significado geométrico de cada coeficiente elástico, as combinações de valores baixos e altos para cada um deles são associados aos efeitos visuais resultantes, ilustrados na figura 5.1, com um plano fixado em todas as suas bordas sob a ação da força gravitacional $g = 9.8m/s^2$.

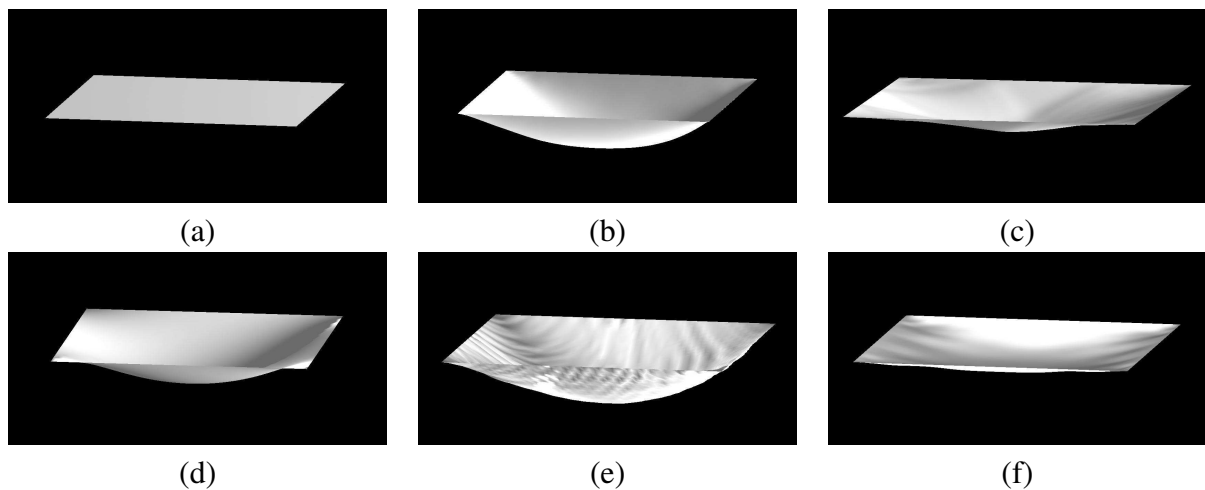


Figura 5.1: Influência dos coeficientes elásticos.

Considerando a superfície inicial, ilustrada na figura 5.1(a), com massa igual a 80 e orientada no plano xz , de forma que a gravidade atue na direção $(0, -9.8, 0)$, as simulações foram obtidas variando-se isoladamente os valores de cada coeficiente elástico da superfície, os quais seguem a convenção de que $\zeta = \zeta_{\alpha\beta}$, da mesma forma que ξ e ϕ .

O parâmetro ζ assume um valor de resistência métrica igual a 10 na figura 5.1(b) e igual a 250 na figura 5.1(c), ambas com ξ e ϕ iguais a 0. Pode-se notar que o coeficiente ζ está associado à resistência métrica da superfície, provocando diferentes graus de elasticidade quando ela, presa em suas quatro bordas, é deformada pela ação da gravidade. Considerando possível que um tecido varie 10% de sua área [10], os valores adequados para ζ variam entre 80, como em tecidos de elastano, até 500, como em lonas bem rígidas. Como em geral os tecidos são compostos de muitos materiais, qualquer combinação dentro deste alcance resulta em uma resistência métrica similar a de um tecido.

Já o coeficiente elástico ξ está associado a suavidade das curvas da superfície. A figura 5.1(c) possui o mesmo valor de ζ que a figura 5.1(b), só diferindo o valor de ξ , que ao invés de 0 é alterado para 10. Pode-se perceber que a curva formada pela região esticada na imagem 5.1(c) não está tão acentuada quanto em 5.1(b), de forma que com o mesmo valor de $\zeta = 10$, a elevada resistência à mudança de curvatura faz com que haja menor deformação da métrica da superfície. O coeficiente ξ possui um alcance diferente de ζ para simular tecidos, sendo mais sensível numericamente devido ao fato da resistência da métrica ser muito maior do que a da curvatura. Os valores devem variar entre 0 e 10, pois a partir deste último valor as superfícies possuem o comportamento de lâminas plásticas.

Por fim, o parâmetro ϕ está relacionado com a frequência de formação de dobras que acontecem nos tecidos. Este parâmetro difere de ξ porque não descreve a suavidade ou a resistência das dobras, mas a capacidade do material em formar as dobras após sofrer deformações na direção exterior à superfície do tecido. Esta relação entre a métrica e a curvatura é descrita com ϕ , como mostrado

na figura 5.1(d), que usa os mesmos valores de 5.1(b), mas com $\phi = 0.5$. Pode-se notar que a mudança do valor de ϕ fez com que várias dobras se formassem ao longo de todo o tecido. Devido ao coeficiente ϕ associar duas grandezas de comportamentos opostos, seu alcance é ainda mais sensível numericamente que ξ , variando de 0 a 1 para simular tecidos. Valores acima de 1 costumam parecer com oscilações dentro do plano da superfície do tecido, muito comum nos modelos do tipo massa-mola, nos quais a rigidez das molas provoca estes movimentos indesejados.

Considerando que os tecido em geral possuem o tipo de deformação relacionado ao coeficiente elástico ϕ , é sempre interessante simular um tecido com um valor não nulo para ϕ , mesmo que da ordem de 0.01, para se obter um comportamento físico mais real. A figura 5.1(e) ilustra a composição de todos os coeficientes elásticos usados em conjunto dentro dos alcances citados, sendo $\zeta = 100$, $\xi = 0.5$ e $\phi = 0.01$. Pode-se perceber nesta imagem que, embora a simulação use um ζ menor do que 5.1(c), a curva formada na região inferior às bordas fixadas apresentam uma suavidade maior em relação à figura 5.1(c), que possui $\xi = 0$. Além disso, como $\phi = 0.01$ na figura 5.1(e), a imagem apresenta mais dobras na lateral do que a figura 5.1(c), que tem $\phi = 0$.

Uma forma de mostrar a relação entre as grandezas referentes à métrica e curvatura é através da simulação da figura 5.1(b), na qual uma superfície de baixa resistência métrica, fixada em toda sua fronteira, sofre a ação da gravidade nos seus pontos internos e se estica, criando área e mudando a curvatura original da superfície. Outra maneira de se ressaltar esta dependência é no modo com que as figuras 5.1(c) e 5.1(d) diminuem a inclinação da curva formada em 5.1(b). Na figura 5.1(c) a resistência à métrica é maior, logo a distensão da área é menor e a curva é menor. Na figura 5.1(c) a resistência à métrica é a mesma de 5.1(b), mas a resistência à curvatura é maior, fazendo com que a medida que a área aumente encontre a resistência das curvas que se formam, resultando também em uma curva final menor que 5.1(b). A figura 5.2 ilustra a influência do coeficiente elástico ϕ , na qual uma superfície inicial recebe a ação de forças externas totalmente tangenciais com a direção contrária, sendo (a) com $\phi = 0$ e (b) com $\phi = 0.01$.



Figura 5.2: Influência de ϕ .

5.1.2 Calibração de Forças Externas

As deformações que acontecem nos tecidos ocorrem devido à ação de forças externas, que são definidas pelo usuário em cada simulação. Na implementação feita considera-se que as forças externas podem ser definidas de três maneiras. A primeira é definindo o módulo e a direção de uma força pontual como uma restrição da simulação. Esta força então é aplicada individualmente ao ponto da malha discreta durante o intervalo determinado, informado em termos das iterações. Uma outra maneira é definindo a aceleração do ambiente, que resulta na mesma força para todos os pontos da superfície. Este tipo de força é muito usado para definir a gravidade do ambiente. A última maneira é definindo uma força de fluido, informando para isso a constante de densidade do fluido e sua direção. Este tipo de força é usado frequentemente para definir a força do vento.

As forças externas pontuais desempenham um papel fundamental no simulador de tecidos. Com elas é possível manipular a malha livremente, adicionando forças somente em um ponto ou em um conjunto de pontos. Por exemplo, para simular um pano de prato sobre uma pia sendo amassado por uma pessoa é necessário adicionar forças pontuais precisas, que simulam os movimentos feitos com as mãos da pessoa. Este tipo de deformação não pode ser produzido pelas forças externas da aceleração do ambiente ou da força fluido, pois cada ponto sofre forças externas bem específicas que definem a forma da superfície do tecido. Enquanto a direção da força é mais intuitiva de ser especificada, seu módulo é mais complicado, pois depende da massa do objeto que recebe a força.

Desta forma, uma consideração importante é utilizar medidas apropriadas para definir o material dos tecidos. Independente da resolução da grade da malha discreta, cada superfície tem uma área, medida pelo seu comprimento e sua largura, que estão diretamente ligadas a quanto o tecido deve pesar. Estas associações de área e peso são feitas observando informações dos fabricantes de tecidos, para saber quanto pesa um metro quadrado de algodão, jeans, linho, veludo etc. Estas medidas então são usadas para aproximar de uma maneira um pouco mais real a massa dos tecidos simulados e, com isso, conseguir restringir os valores usados para calibrar as forças externas.

As imagens da figura 5.3 ilustram duas aplicações de forças pontuais no simulador. A figura 5.3(a) corresponde a um tecido que originalmente possui a forma da figura 5.1(a) e, após receber uma força $\mathbf{f} = (0, -9.8, 0)$ exatamente no seu ponto central, em um ambiente sem a existência da gravidade, cai levando o restante do tecido. Já a figura 5.3(b) mostra um tecido esticado por forças posicionadas nos seus quatro cantos. Forças de módulos altos têm grandes chances de levar a simulação a uma situação de instabilidade. Um bom valor de referência para as forças externas é o valor da gravidade -9.8 . Forças com módulos em torno de 10 a 100 normalmente são adequadas para a maioria das simulações de tecidos. Estes módulos também dependem da duração das forças, pois pode-se provocar um mesmo deslocamento de forma brusca ou suave. Uma cortina, por exemplo, pode ser aberta com um rápido movimento ou de maneira lenta, porém resultando no mesmo estado final.

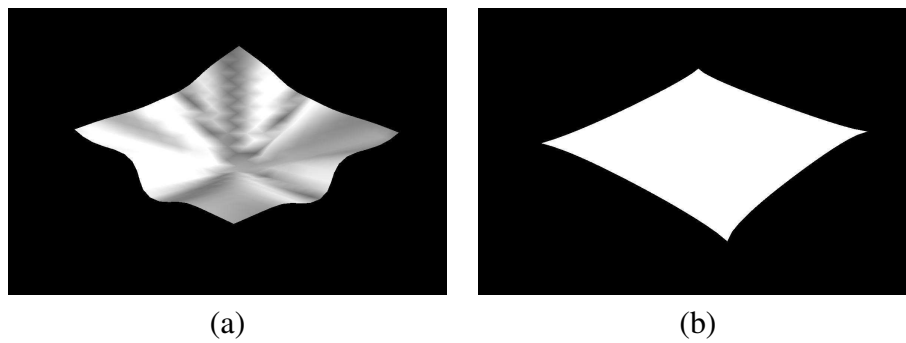


Figura 5.3: Atuação de forças externas pontuais.

Com relação às forças externas da aceleração do ambiente, a definição é feita com a atribuição de um único vetor na interface gráfica do simulador. Assim, a aceleração preenchida na interface é adicionada para cada ponto da superfície, que provoca um comportamento uniforme na mesma direção dos pontos da superfície em resposta a estas forças. Nas simulações de teste do capítulo 3 a maioria das simulações utiliza a aceleração do ambiente para definir a força de gravidade, como na formação do banco de praça e da poltrona, definindo pontos fixos e deixando a gravidade agir. Outra forma de adicionar forças externas é com a força fluido, a qual foi usada, por exemplo, na simulação de teste da bandeira também no capítulo 3.

Devido a diversidade de forças externas que podem ser adicionadas, pode-se atingir os mesmos efeitos com diferentes parâmetros de simulação do modelo de tecido. Contudo, com base na análise feita sobre estes parâmetros, é possível propor um procedimento composto por cinco passos para obter estes parâmetros de maneira apropriada, conforme descrito pelos seguintes passos:

1. A densidade de massa do tecido é estimada com base nas especificações técnicas fornecidas pelos fabricantes.
2. Um conjunto de forças é definido em analogia a situações físicas reais.
3. Considerando $\xi_{\alpha\beta} = \phi_{\alpha\beta} = 0$, varia-se o valor de $\zeta_{\alpha\beta}$ até que se obtenha uma superfície visualmente sem esticamentos. Outra forma de fazer este passo é monitorando a mudança da área que é informada na visualização da animação, tomando como limite superior uma variação de 10% em relação à área inicial. Na maioria dos casos usou-se o valor $\zeta_{\alpha\beta} = 100.0$, entretanto, notou-se que este valor é dependente da massa da superfície. Quanto maior é a massa, maior tem que ser o valor de $\zeta_{\alpha\beta}$.
4. Fixando $\zeta_{\alpha\beta} = 100.0$ e mantendo $\phi_{\alpha\beta} = 0$, testa-se empiricamente alguns valores para dois tipos de material, até que os comportamentos visuais possam ser comparados com situações reais. Estes dois tipos de material de tecido correspondem à seda, que é um material muito

leve e macio, e à lona, que é um material pesado e rígido. A partir destes dois extremos, é possível estimar intuitivamente outros materiais com base no grau de rigidez que algum material apresenta dentro deste intervalo.

5. Estabelecendo os valores de $\zeta_{\alpha\beta}$ e $\xi_{\alpha\beta}$, determina-se o valor de $\phi_{\alpha\beta}$ aplicando forças de compressão e distensão e comparando os efeitos visuais com os reais. Quanto maior é este valor, maior o número de dobras formadas.

Este procedimento consegue encontrar uma configuração apropriada para definir um material de um tecido. A ordem dos passos é muito importante, pois ela parte primeiro da massa do tecido, a qual é definida da forma mais precisa possível considerando valores reais, para então estabelecer forças externas que são valores dependentes desta massa. A seguir, os coeficientes elásticos do tecido são calibrados um a um de forma empírica, sendo primeiro ζ referente à resistência à mudança de métrica, depois ξ referente à resistência à mudança de curvatura e, por último, ϕ correspondente à facilidade de formação de dobras. Embora seja um procedimento que encontre alguns valores empiricamente, a realimentação existente com resultados visuais facilita a convergência da simulação para o comportamento esperado. Para a execução das simulações necessárias até obter o resultado desejado recomenda-se usar um passo de tempo em torno de 0.002 e, somente após calibrar corretamente os parâmetros para uma simulação estável, aumentar este valor atualizando as restrições se necessário.

5.2 Simulações

Nesta seção são apresentadas as simulações geradas com o modelo de tecido proposto por de Melo [4] integrado ao longo do tempo com o método de Verlet. Estas simulações estão divididas de acordo com o tipo de deformação que o tecido sofre. A ação de amassar, enrugar ou criar dobras em um tecido é conhecida como *buckling*, sendo esta deformação muito pesquisada na literatura [12, 11, 21]. As simulações de *bucklings* são mostradas com deformações causadas por forças pontuais sem a atuação da força de gravidade, conforme detalhado na subseção 5.2.1. Outro tipo de deformação comum são os caimentos de tecidos sobre objetos, conhecidos como *drapings* e apresentados na subseção 5.2.2. Como neste trabalho não existe o tratamento de colisões, as simulações deste tipo são feitas com a definição de pontos fixos ao longo do tempo que simulam a geometria de objetos de colisão. Este tipo de deformação é muito comum para simular toalhas caindo sobre mesas com a ação da gravidade, como fez Breen et al. [26]. Algumas vezes, a suspensão de tecidos também é conhecida como *drapings*, mas neste trabalho elas serão chamadas de *hangings* e ilustradas na subseção 5.2.3. Elas correspondem a simulações de cortinas e bandeiras, por exemplo. Por fim, a subseção 5.3.1 apresenta a dinâmica de tecidos mostrando seqüências de imagens geradas ao longo do tempo.

5.2.1 *Bucklings*

As simulações de *buckling* podem ser dividida de duas formas. As que utilizam somente forças de compressão e distensão tangentes ao plano da superfície, e as demais, que utilizam forças pontuais em qualquer direção. Uma das contribuições do modelo proposto por de Melo [4] é exatamente a de conseguir produzir efeitos de *buckling* somente com forças de compressão. As imagens da figura 5.4 ilustram quatro tecidos submetidos as mesmas forças de compressão, porém variando o material do tecido. A figura 5.4(a) corresponde a um tecido de seda, o qual é muito permissivo a formação de dobras, como pode-se notar. O material do tecido da figura 5.4(b) é o algodão, sendo possível notar que existem menos dobras em relação ao de seda. De forma semelhante, o tecido de jeans na figura 5.4(c) e o de lona na figura 5.4(d) também possuem quantidades de dobras diferentes.

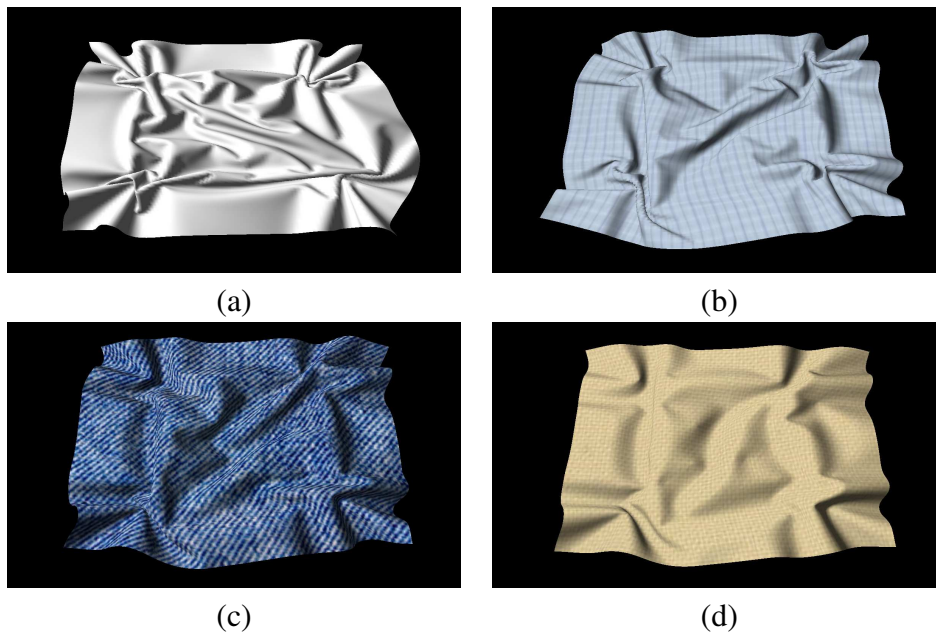


Figura 5.4: *Bucklings* de diferentes materiais de tecido.

O diferencial deste tipo de resultado é que as respostas da superfície na direção para fora do plano acontecem naturalmente, sem a imposição de forças artificiais nestas direções quando a superfície é submetida apenas a forças de compressão. Estas situações acontecem quando se aperta um algum tipo de tecido no ar. Naturalmente, ao se comprimir um tecido, ele se curva na direção perpendicular à superfície, como o comportamento das imagens da figura 5.4. As diferenças entre os parâmetros usados para obter estas simulações são os valores de massa e principalmente os coeficientes elásticos $\xi_{\alpha\beta}$, o qual está relacionado com a dificuldade do material se curvar. Além disso, os coeficientes $\phi_{\alpha\beta}$ não são nulos, pois quando ocorrem somente forças de compressão o coeficiente de *buckling* é fundamental para obter resultados corretos.

Um segundo tipo de simulação de *buckling* que pode ser feito é usando forças pontuais que não sejam estritamente tangenciais ao plano da superfície. Desta forma, uma maneira de produzir uma simulação interessante deste modo é aplicando algumas forças de compressão e depois ir incrementando forças para fazer pequenos ajustes na superfície do tecido. As imagens da figura 5.5 apresentam duas gravatas borboletas, sendo 5.5(a) uma gravata de material mais flexível (seda) e 5.5(b) mais rígido (veludo). Estas imagens foram obtidas adicionando forças de compressão em alguns pontos centrais da superfície do tecido na direção vertical, as dividindo em sentidos opostos de cada lado para evitar auto-colisões. Depois da formação das dobras no meio do tecido, que correspondem a parte achatada da gravata borboleta, forças para fora do plano são adicionadas para fazer pequenos ajustes na gravata. Estes ajustes correspondem ao que ocorre na realidade, quando ao vestir uma gravata deste tipo puxa-se com as mãos suas extremidades para que seus lados fiquem iguais. Isto acontece porque como o tecido é muito flexível, ao fazer forças de compressão ele pode tanto se curvar para um lado quanto para o outro e, portanto, o ajuste ocorre no sentido de deixar as dobras para o mesmo lado.

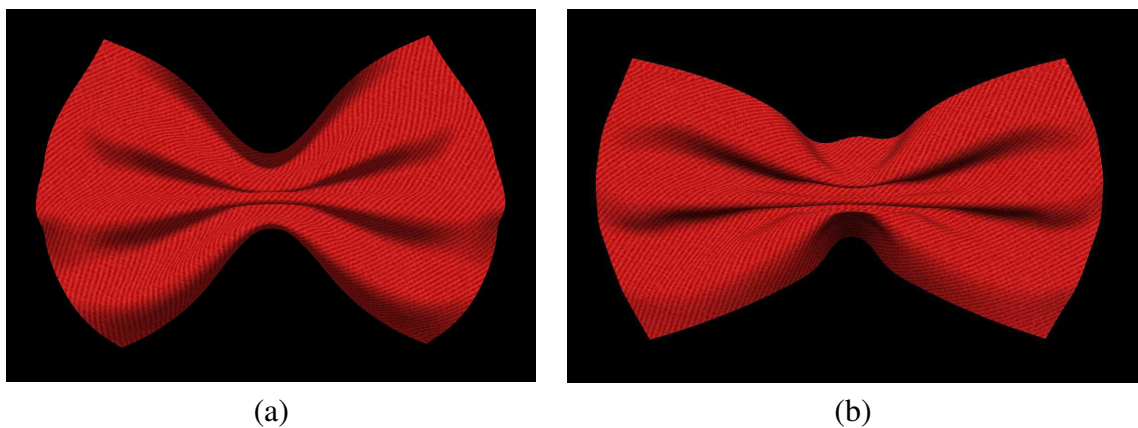


Figura 5.5: Formação de uma gravata borboleta (a) flexível e (b) rígida.

Pode-se notar que a figura 5.5(a) apresenta os contornos mais arredondados do que a figura 5.5(b). Esta diferença, ao contrário das imagens da figura 5.4, ocorrem simplesmente por valores dos coeficientes elásticos $\xi_{\alpha\beta}$ distintos, pois as duas gravatas borboletas estão definidas com a mesma massa e com os mesmo coeficientes elásticos $\zeta_{\alpha\beta}$ e $\phi_{\alpha\beta}$. Com relação aos ajustes feitos por forças pontuais não tangentes a superfície, é possível notar que na figura 5.5(a) os lados da gravata achatados estão puxados para cima, na direção saindo do papel desta dissertação, enquanto os da figura 5.5(b) estão no sentido para baixo, apenas para ilustrar o resultado provocado por este tipo de ajuste.

5.2.2 *Drapings*

As simulações de tecidos que consistem de caimentos da superfície do tecido sobre objetos são conhecidas como *drappings*. Como este trabalho não utiliza algoritmos que façam o tratamento de colisões, os resultados mostrados nesta subseção utilizam a definição de pontos fixos como restrições da simulação para simular a geometria dos objetos de colisão. Assim, um caimento de uma toalha sobre uma mesa redonda ou quadrada é feita definindo o formato da mesa com pontos fixos sobre a malha discreta. A figura 5.6 ilustra os pontos fixos como bolas vermelhas sobre um reticulado de pontos que representa a malha discreta. A duração dos pontos fixos é definida em termos das iterações da simulação, fazendo com que cada ponto seja fixo nos três eixos a partir de uma certa iteração até a iteração final, deixando o caimento acontecer com a força da gravidade. Este caimento termina quando a superfície entra em um estado de equilíbrio, que no caso de um caimento sobre uma mesa corresponde ao momento em que a área do tecido que não está sobre a mesa não apresenta mais movimentos entre um instante de tempo e o seguinte.

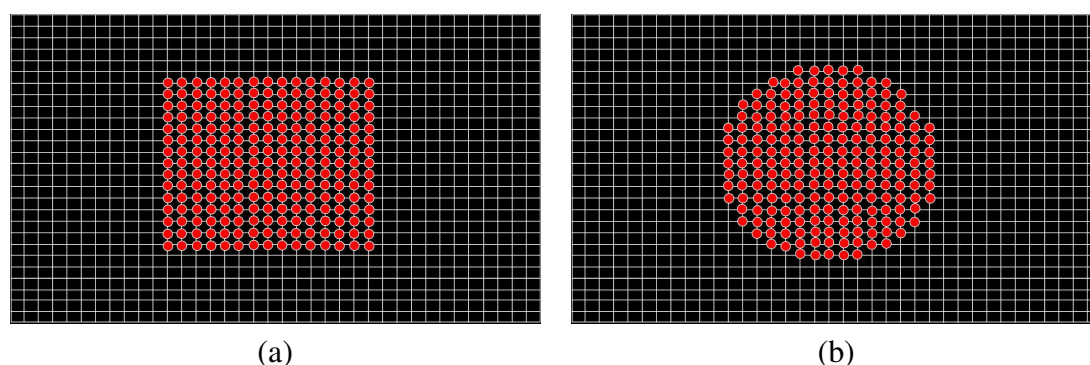


Figura 5.6: Pontos fixos na malha discreta para formar uma mesa (a) quadrada e (b) redonda.

As resoluções espaciais das malhas discretas da figura 5.6 são apenas ilustrativas, pois pode-se definir a mesma simulação de *draping* com resoluções diferentes, ou até outras simulações similares deixando mais ou menos pontos para a definição do plano da mesa. Uma vez definida as restrições de pontos fixos sobre uma malha discreta pode-se usá-las variando apenas o material do tecido e obter diferentes efeitos. Por exemplo, a figura 5.7(b) ilustra o caimento de um tecido de cetim sobre uma mesa redonda, já a figura 5.7(d) apresenta a imagem de uma simulação com as mesmas restrições porém com um tecido de linho. Ambas as imagens são comparadas com um foto real da situação, conforme as fotos da figura 5.7(a) e 5.7(c). Desta forma, é possível ter uma referência do comportamento do tecido durante o procedimento de calibração do material explicado na seção 5.1, para que a cada tentativa de configuração dos parâmetros a imagem final do estado de equilíbrio atingido possa ser comparada com o caimento da foto real. Esta é uma forma interessante de avaliar o realismo obtido nas imagens, embora na prática nem sempre é possível ter uma foto de referência.

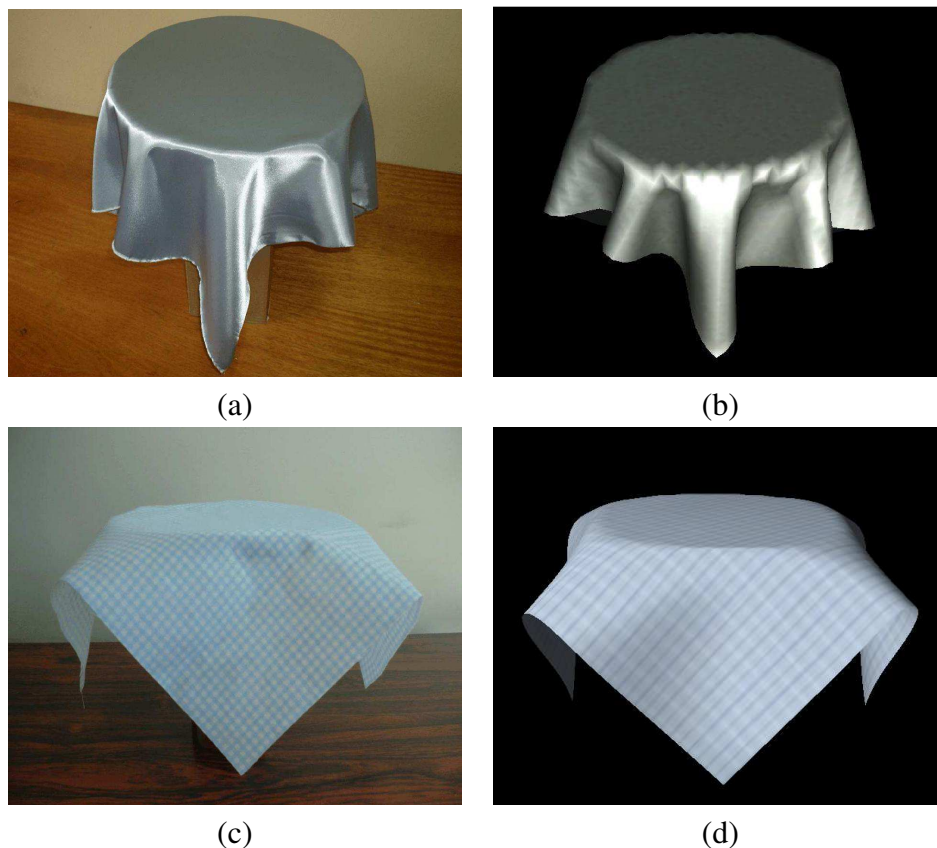


Figura 5.7: Caimentos de tecidos sobre mesas redondas.

Nestas imagens os parâmetros do material do tecido usados para o cetim são muito próximos dos valores usados para o linho, variando apenas a massa, sendo o linho mais pesado que o cetim, e também o valor do coeficiente elástico $\xi_{\alpha\beta}$, pois o linho tem maior resistência à mudança de curvatura do que o cetim. Mais um exemplo da facilidade em aproveitar as restrições e definições dos parâmetros da simulação e mudar apenas um valor para obter resultados bem diferentes é ilustrado na figura 5.8(a), na qual usou-se a mesma massa do tecido de cetim mas com uma resistência à curvatura bem menor, como se fosse um algodão.

Com relação ao caimento de um tecido sobre uma mesa quadrada a simulação é feita de forma muito semelhante em relação a uma mesa redonda. Usando uma malha com os pontos fixos configurados de acordo com a figura 5.6(a), a figura 5.8(b) ilustra o resultado de um tecido usado para cobrir mesas de cozinha ou sala, que normalmente são de algodão mas contém uma camada de plástico na superfície, sendo mais rígido que o algodão porém tão leve quanto. Desta forma, as dobras formadas nos cantos do tecido são bem armadas e com dobras bem definidas. Este tipo de dobra formada na região da fronteira do tecido era um problema nas imagens obtidas no trabalho proposto por de Melo [4], pois como vários elementos do modelo eram zerados nestes pontos as dobras eram mal

formadas. Com a discretização espacial apresentada na seção 3.1 deste trabalho, foi possível obter melhores resultados neste aspecto e assim aumentar o realismo das imagens obtidas.

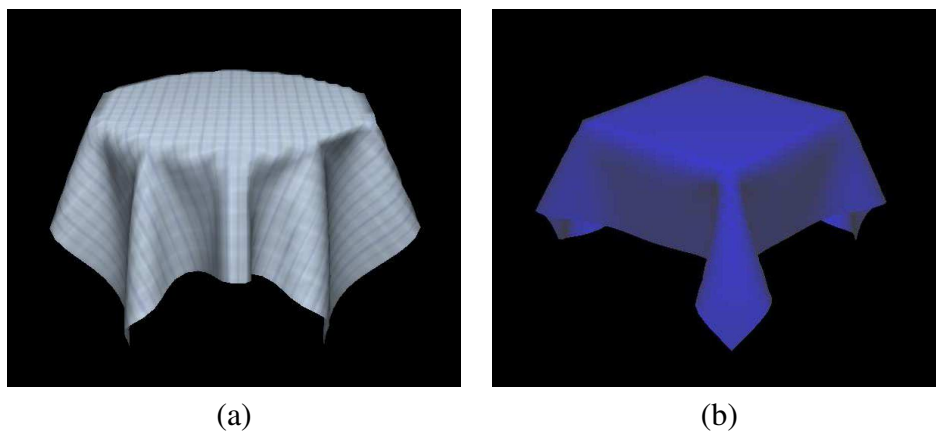


Figura 5.8: Caimento de tecido sobre uma mesa (a) redonda e (b) quadrada.

As simulações de *draping* também podem ser feitas com objetos mais complexos que mesas, demandando para isso um maior esforço na definição das restrições por falta de tratamento de colisões e auto-colisões. Para definir um objeto mais detalhado, como uma poltrona, é preciso definir pontos fixos em diferentes instantes de tempo que formarão as partes da poltrona de forma isolada. Desta forma, à medida que a superfície inicial do tecido cai de acordo com a gravidade, primeiro os pontos do encosto da poltrona são fixados para depois serem fixados os braços da poltrona e por último o assento. A figura 5.9 ilustra o estado final de equilíbrio no caimento de um tecido de veludo sobre uma poltrona.



Figura 5.9: Caimento de um tecido sobre uma poltrona.

5.2.3 Hangings

Nas simulações de *hangings* o tecido é suspenso por alguns pontos fixos e a ação da gravidade provoca as deformações nele. Os exemplos mais comuns deste tipo de simulação são cortinas presas em trilhos e bandeiras hasteadas em um poste. Em relação às cortinas, as simulações podem ser feitas definindo pontos fixos em uma linha de pontos da malha discreta próxima da fronteira superior do tecido, de forma a permitir que estes pontos se desloquem somente no eixo horizontal para que a cortina possa ser aberta e fechada. As imagens da figura 5.10 ilustram quatro simulações de cortinas com materiais de tecidos diferentes. A cortina da figura 5.10(a) ilustra um tecido bem flexível, como uma cortina de algodão. A figura 5.10(b) mostra uma cortina mais rígida, como uma cortina de sala relativamente pesada, em que as dobras são armadas. Já a figura 5.10(c) exibe uma cortina de banheiro, que possui um material leve e plastificado, permitindo a formação de dobras armadas também. Todas estas simulações são feitas partindo de uma superfície lisa formada no plano xy e, após a definição dos pontos fixos que formam o trilho da cortina, forças pontuais são inseridas nas duas extremidades da cortina para simular o movimento de abri-la até certo ponto. Após algumas iterações, os pontos fixos do trilho são restritos em todos os eixos, inclusive no eixo horizontal, e o restante do tecido demanda ainda um número considerável de iterações até que suas áreas nas extremidades laterais caiam de forma a ficar o mais perpendicular possível ao trilho da cortina.

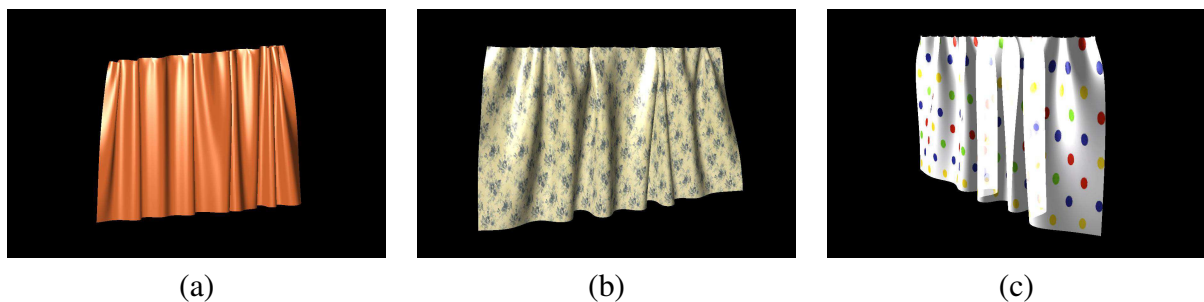


Figura 5.10: Cortinas suspensas de diferentes materiais: (a) seda; (b) algodão; e (c) plastificada.

Outro exemplo de uma simulação de *hanging* é uma bandeira balançando de acordo com a força do vento. Neste caso, dois pontos extremos da lateral da superfície do tecido são definidos como fixos em um eixo, simulando os pontos em que a bandeira está amarrada a um poste. Para obter o efeito da força do vento pode-se usar a força fluido definindo uma direção e uma densidade de fluido apropriada. Assim, a partir da superfície inicial definida no plano yz , conforme mostra a figura 5.11(a), a força de fluido é especificada com a direção $\mathbf{f} = (100, 0, 0)$ e com uma densidade 0.032, fazendo com que a bandeira inicialmente caia de acordo com a gravidade ao mesmo tempo que se desloca em direção da força de fluido. Após algumas iterações, a bandeira que estava no plano yz agora encontra-se no plano xy e, uma vez atingido este plano, a superfície começa a sofrer

deformações que, embora não sejam iguais, repetem um certo padrão após algumas iterações em relação a formação de dobras. A imagem da figura 5.11(b) ilustra a bandeira já deformada pela ação da força fluido. Este tipo de deformação pode ser interpretado como um mergulho da superfície em algum líquido ou meio, os quais podem ter uma força associada.

Em relação ao estado de equilíbrio final desta simulação, diferentemente das outras apresentadas até agora, a bandeira não atinge um estado em que não existam mais movimentos entre dois passos subsequentes, a não ser que a força fluido seja interrompida. No capítulo 3, durante a execução das simulações de teste para descobrir qual o melhor método explícito que poderia ser aplicado ao modelo proposto por de Melo [4], foi considerado como estado de equilíbrio da bandeira o momento em que a superfície consegue pela primeira vez se manter em uma linha horizontal, mesmo com a ação da gravidade. A imagem da figura 5.11(b) foi escolhida pensando neste equilíbrio, mas também visando mostrar o maior número de dobras produzidas por este tipo de deformação.

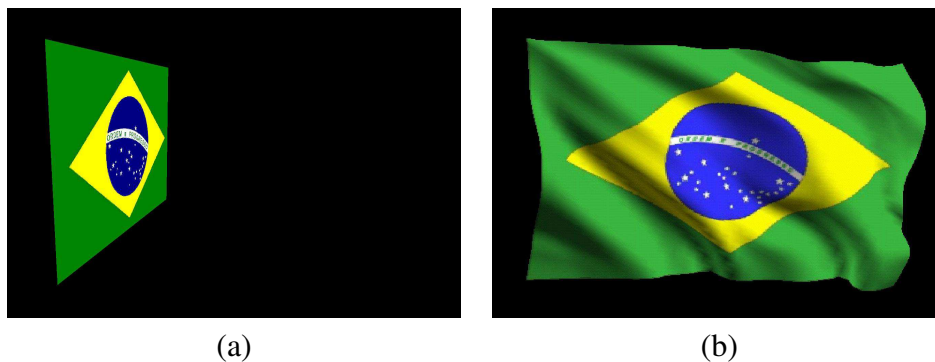


Figura 5.11: Bandeira no estado (a) inicial e (b) deformado.

5.3 Análises

Esta seção analisa os resultados mostrados na seção anterior, abordando o realismo nas imagens geradas e a precisão física na diversidade de dobras formadas e também no movimento de tecidos até os estados de equilíbrio. A dinâmica dos tecidos nas simulações feitas são analisadas com seqüências de imagens na subseção 5.3.1, mostrando as variações da superfície inicial até a superfície deformada em estado de equilíbrio. O realismo das imagens é analisado em termos da qualidade e diversidade das dobras formadas, discutidas na subseção 5.3.2. Nesta subseção é apresentada uma tabela com os valores dos principais parâmetros do tecido e da simulação utilizados nas simulações deste capítulo. A análise do tempo de processamento das simulações não é feito devido ao fato de se utilizar o método de Verlet, o qual foi mostrado no capítulo 3 que apresenta complexidade computacional linear. Contudo, os tempos por iteração de cada simulação são mostrados na tabela 5.1 da subseção 5.3.2.

5.3.1 Dinâmica

O objetivo desta subseção é mostrar como a dinâmica de tecidos é feita no simulador implementado. A melhor forma de fazer isto é com a visualização de animações, para que se perceba efetivamente a continuidade no movimento de tecidos e avalie-se o realismo atingido. Todos as imagens mostradas ao longo desta seção são quadros de animações, que podem ser encontradas em [37]. Contudo, na tentativa de transmitir esta dinâmica de tecidos sem o recurso da animação, esta subseção apresenta seqüências de imagens para mostrar as diferentes geometrias da superfície ao longo do tempo. Para os principais resultados, são usados quatro quadros diferentes associados ao tempo em que eles foram obtidos. O primeiro quadro corresponde a superfície inicial e o último ao estado de equilíbrio ou a um determinado comportamento esperado. A figura 5.12 apresenta as seqüências de imagens do processo de *buckling* dos quatro panos exibidos na figura 5.4.

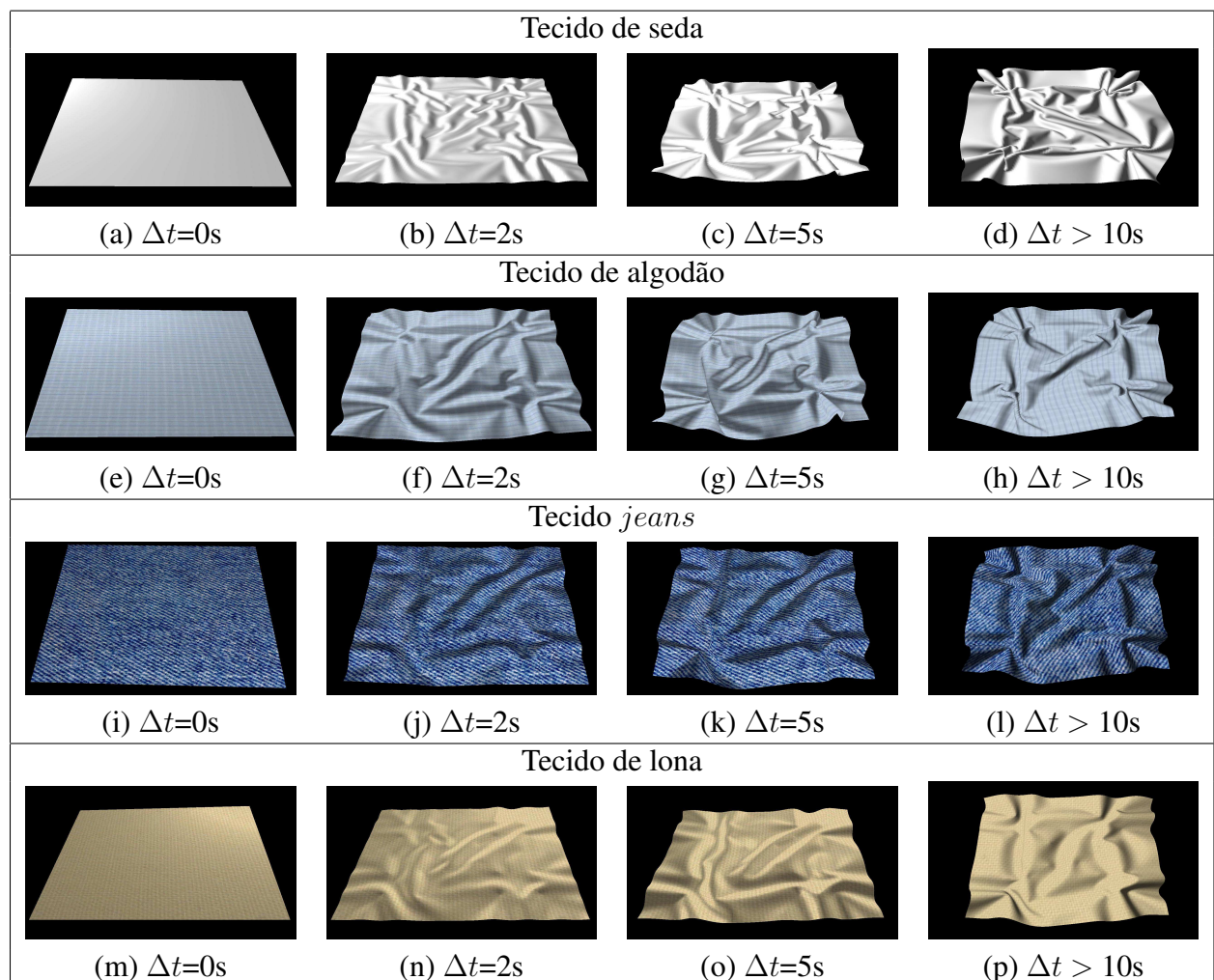


Figura 5.12: Dinâmica de *bucklings*.

Na imagem 5.12 pode-se notar que as geometrias das quatro superfícies iniciais são iguais, embora sejam compostas de materiais diferentes e resultem em diferentes deformações. Como os materiais são diferentes, as massas dos tecidos são diferentes, portanto, é preciso atualizar os módulos das forças externas que atuam sobre o tecido para cada simulação. Isto é feito para que as deformações aconteçam de forma proporcional, destacando a formação de dobras de diferentes materiais após um determinado tempo de referência. Ao se aplicar uma força muito pequena em um tecido de seda, por exemplo, ele se deformará mesmo que lentamente, contudo, um tecido de lona pode permanecer imóvel com a aplicação da mesma força. Ao todo são aplicadas vinte forças externas de mesmo módulo de forma distribuída, fazendo o tecido se amassar em torno de seu centro (figura 4.12(b)). Estas forças são exercidas até o tecido apresentar a maior deformação possível sem que ocorram auto-colisões, quando então o módulo das forças é diminuído para a superfície entrar em equilíbrio com as dobras formadas. Esta simulação é equivalente a situação em que uma pessoa faz certa força para deformar um tecido e, após deformado, o segura com menor força para manter as dobras formadas.

A imagem da figura 5.12(b) mostra as deformações da superfície após 2s sob a ação das forças externas. Pelo fato da seda ser muito flexível, rapidamente o tecido se deforma formando diversas dobras. Nas imagens da coluna (b) pode-se notar que mesmos nos instantes iniciais de deformação já é possível verificar qual tecido é mais permissivo para se curvar em relação ao outro, que no caso, após a seda é o algodão, *jeans* e por último, a lona. Na figura 5.12(c) e nas outras três desta coluna são mostradas as geometrias da superfície alguns instantes de tempo à frente. Percebe-se que como as forças externas são aplicadas no sentido para o centro do tecido as dobras começam a ficar mais altas, diminuindo a área projetada no plano da superfície inicial. Após mais algum tempo, as forças são reduzidas para apenas manter as dobras criadas e o tecido entra em estado de equilíbrio, que corresponde as imagens da coluna (d). Nesta última coluna pode-se notar que algumas dobras antes criadas já não existem mais e, no caso da seda e do algodão, as dobras remanescentes são bem finas e acentuadas, enquanto as do *jeans* e da lona são grossas e suaves.

Com relação aos resultados de *drapings*, foram selecionadas as simulações do caimento de tecidos sobre a mesa redonda de algodão e sobre a poltrona. Nestas simulações o tecido cai verticalmente e, à medida que avança encontra restrições de pontos fixos que simulam o seu contato com a poltrona. A imagem da figura 5.13(a) mostra a superfície inicial que cai sobre uma mesa redonda. Na figura 5.13(b) já é possível ver que os pontos associados a mesa já foram fixados, restando cair as áreas do tecido que ficarão penduradas. Na figura 5.13(c) pode-se ver a criação de dobras na região do tecido que não está sobre a mesa e, na figura 5.13(d) o tecido já se encontra em estado de equilíbrio. A dinâmica do tecido que cai sobre a poltrona também é feita da mesma forma, diferindo apenas no material do tecido e na definição dos pontos fixos. Partindo de uma superfície inicial como na figura 5.13(e), o tecido cai de acordo com a gravidade e o primeiro grupo de pontos fixos que

são executados são os associados a parte de cima do encosto da poltrona, conforme a figura 5.13(f). Em seguida, a figura 5.13(g) mostra a acomodação do tecido sobre os braços da poltrona e 5.13(h) o estado de equilíbrio.

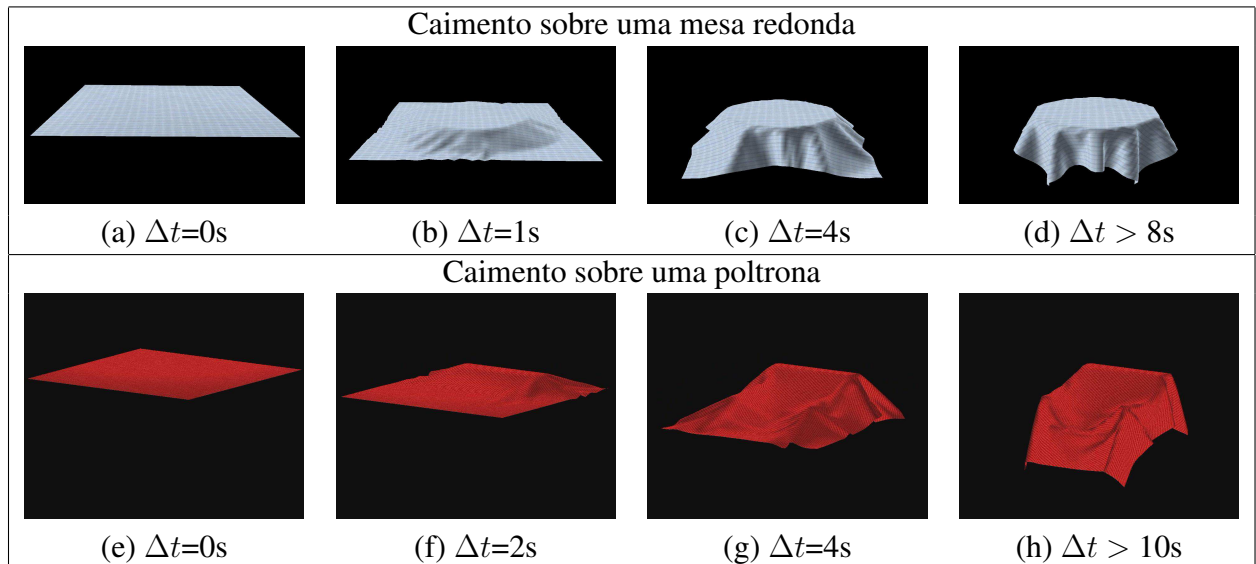


Figura 5.13: Dinâmica de *drapings*.

É interessante notar que qualidade das animações pode ser comprometida pela falta de um tratamento apropriado de colisões e auto-colisões. Embora definir os pontos de colisão como pontos fixos seja uma solução visualmente aceitável, a desconsideração de movimentos sutis, como escorregamento do tecido sobre objetos de colisão devido a fricções, pode gerar efeitos um tanto artificiais.

Por fim, para mostrar a dinâmica dos resultados de *hanging*, foram selecionadas as simulações da bandeira balançando com a força do vento e dois tipos de cortina, a mais flexível e a mais rígida. A figura 5.14(a) mostra a superfície inicial da cortina laranja da figura 5.10(a). Possuindo um material bastante maleável, esta cortina sofre forças externas de cada uma de suas extremidades no sentido de sua parte central, como se a cortina estivesse sendo aberta. A imagem da figura 5.14(b) mostra as primeiras deformações causadas na superfície e 5.14(c) corresponde ao exato momento em que a superfície atingiu seu limite de dobras sem auto-colisões. Neste instante, os pontos do trilho são fixados nos três eixos e então o tecido balança lentamente até parar de se mexer, que corresponde ao estado de equilíbrio ilustrado na imagem da figura 5.14(d). Esta cortina flexível foi simulada considerando um material leve, como a seda, visando ser uma cortina de sala.

A cortina estampada corresponde a uma cortina plastificada de banheiro mostrada na figura 5.10(c). Ela é uma cortina leve e que se dobra com facilidade, porém de forma mais armada. A figura 5.14(e) mostra a superfície inicial desta cortina, que é de mesma resolução espacial e tem a mesma área da

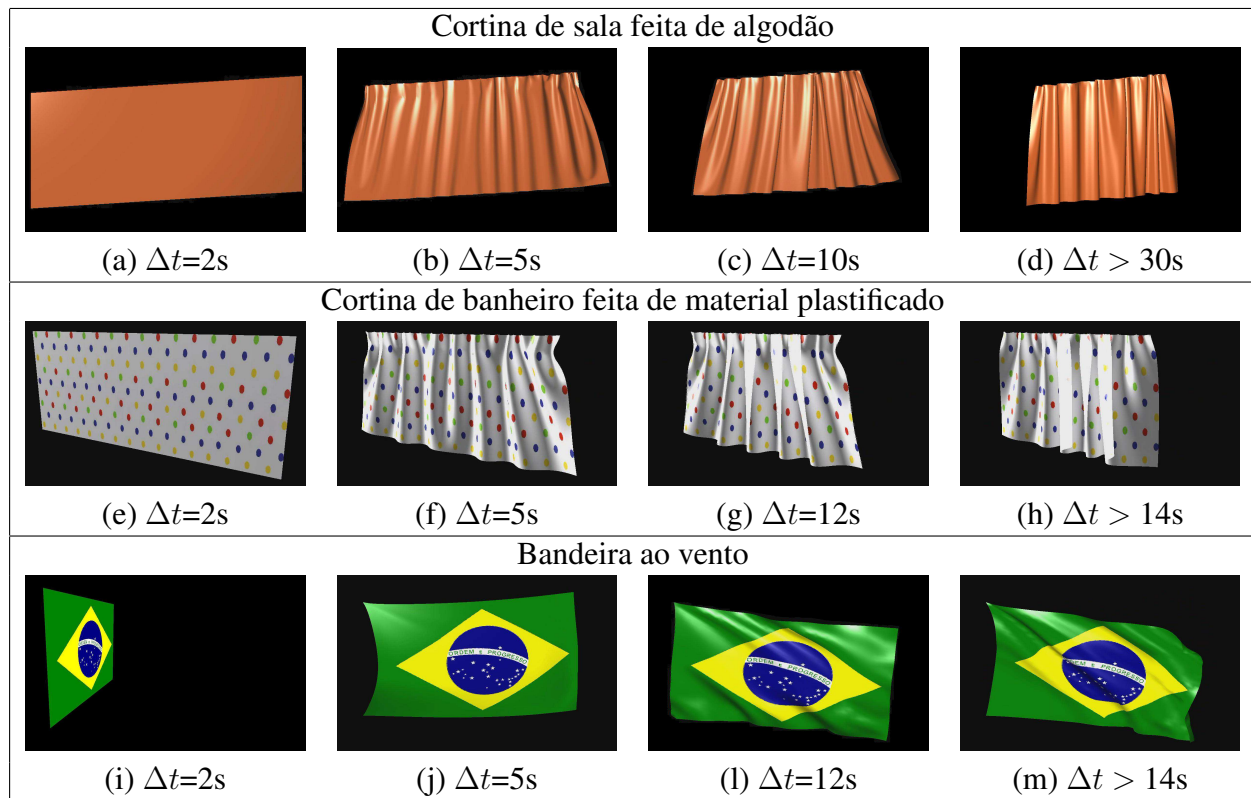


Figura 5.14: Dinâmica de *hangings*.

cortina flexível. Em seguida, pode-se ver a cortina rígida já bastante deformada em pouco tempo, como mostra a figura 5.14(f). Nesta figura é possível reparar que o número de dobras formadas ao longo da direção horizontal do tecido é menor em relação à cortina flexível. Na figura 5.14(g) esta diferença fica maior, com as dobras da cortina de banheiro tomando formas mais armadas. Por fim, a figura 5.14(h) mostra o estado de equilíbrio, com as partes de baixo e cima da cortina alinhadas, assim como ocorre na flexível.

A dinâmica da bandeira da figura 5.11 é feita usando a força fluido. Partindo da superfície inicial ilustrada na figura 5.14(i), a qual está posicionada no plano yz , a bandeira sofre ação da gravidade no sentido para baixo e da força fluido no sentido positivo do eixo x , trazendo a superfície para o plano xy de forma já deformada, como mostra a figura 5.14(j). A partir deste momento, a superfície começa a apresentar um comportamento oscilatório provocando dobras que se formam no canto inferior esquerdo e se propagam até o canto superior direito. As figuras 5.14(l) e 5.14(m) ilustram duas geometrias da superfície com este comportamento. Nesta simulação não existe um estado de equilíbrio em que não existam mais movimentos do tecido, a não ser que a força fluido seja interrompida.

5.3.2 Dobras

Como os tecidos são muito flexíveis para se curvar e pouco para esticar, eles possuem uma grande facilidade de formar dobras à medida que se deformam. As dobras formadas normalmente são de grande diversidade em termos de quantidade e suavidade, pois dependem muito do material de que é composto o tecido. Por esta diversidade, as dobras formadas nos tecidos são problemas que demandam mais atenção nas simulações de tecidos, pois conseguir associar cada tipo de deformação com um material é uma tarefa complexa. Nesta subseção são analisadas as dobras nos tecidos das imagens apresentadas, associando-as com os parâmetros do material da superfície.

Para melhor apresentar esta análise, os valores dos materiais do tecido utilizados nas simulações deste capítulo estão sintetizados na tabela 5.1. Ela mostra o número de pontos (representado #pontos) que formam a superfície do tecido, sua massa, os coeficientes elásticos e o tempo de processamento. Este tempo de processamento corresponde ao tempo gasto por cada iteração da dinâmica de tecidos, com o intuito de mostrar o valor do tempo gasto para resoluções espaciais da malha discreta que não foram usadas no capítulo 3.

Imagem	#pontos	μ (g/m ²)	$\zeta_{\alpha\beta}$	$\xi_{\alpha\beta}$	$\phi_{\alpha\beta}$	Tempo (s)
Figure 5.4(a)	14400	78	100	0.01	0.1	0.2998
Figure 5.4(b)	14400	88	100	0.02	0.1	0.2941
Figure 5.4(c)	14400	111	100	0.03	0.1	0.2807
Figure 5.4(d)	14400	133	100	0.05	0.1	0.2816
Figure 5.5(a)	3600	80	100	0.015	0.1	0.0827
Figure 5.5(b)	3600	100	100	0.03	0.1	0.0755
Figure 5.7(b)	900	100	100	0.01	0.01	0.0174
Figure 5.7(d)	900	120	100	0.04	0.01	0.0163
Figure 5.8(a)	900	85	100	0.005	0.01	0.0187
Figure 5.8(b)	900	125	200	0.05	0.01	0.0151
Figure 5.9	2500	110	100	0.025	0.1	0.0514
Figure 5.10(a)	14400	70	100	0.005	0.1	0.2973
Figure 5.10(b)	14400	120	100	0.2	0.1	0.2874
Figure 5.10(c)	14400	90	100	0.05	0.1	0.2803
Figure 5.10(d)	14400	100	100	0.03	0.1	0.2876
Figure 5.11(b)	1600	90	100	0.01	0.1	0.0322

Tabela 5.1: Dados das simulações.

Começando pelas imagens dos panos amassados, mostradas na figura 5.4, pode-se notar que os tecidos de seda e de algodão apresentam dobras com geometrias mais irregulares dos que as formadas nos tecidos de *jeans* e lona. A definição destas dobras irregulares proporciona um aspecto diferenciado para a qualidade da imagem, resultando em uma imagem mais realista na simulação de tecidos com essa flexibilidade. Vale destacar que todas as dobras da figura 5.4 foram obtidas por forças de compressão ao plano da superfície inicial. Embora esta situação possa ser obtida com a adição de

forças artificiais perpendiculares ao plano da superfície, o modelo de tecido utilizado neste trabalho consegue gerar este comportamento físico de forma automática.

Outro fator importante relacionado à qualidade das dobras é a resolução espacial da malha, que corresponde ao número de pontos que formam a superfície do tecido. As imagens mostradas no capítulo 3, por exemplo, utilizavam malhas de ordem 60×60 no máximo, resultando em 3600 pontos. Já os *bucklings* da figura 5.4 foram todos feitos com malhas 120×120 , resultando em 14400 vértices. Nota-se que a diferença na qualidade das imagens é muito grande. Para salientar o impacto do número de pontos que formam a superfície na qualidade da imagem obtida, a figura 5.15 mostra simulações de *bucklings* dos mesmos quatro panos da figura 5.4, porém utilizando malhas de resolução 60×60 . Nesta comparação é importante manter o valor dos espaçamentos Δx^1 e Δx^2 para que a discretização espacial não fique nem muito grosseira, estimando os elementos do modelo com pouca precisão, e nem muito pequena a ponto de se tornar instável numericamente.

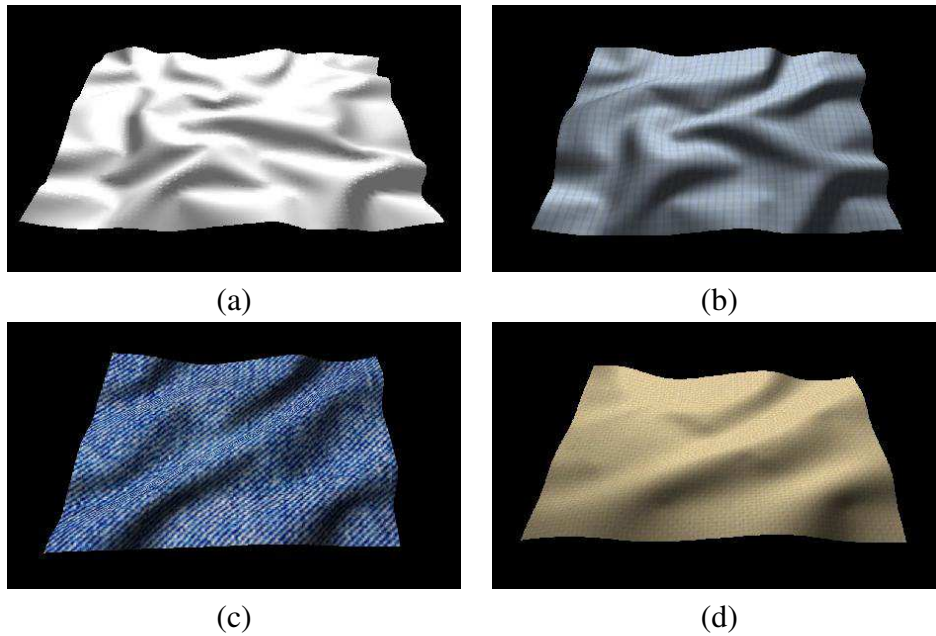


Figura 5.15: *Bucklings* em grades 60×60 .

Embora a qualidade das dobras caia bastante ao usar os mesmos parâmetros para os materiais e modificar a resolução espacial da malha discreta, os resultados continuam condizentes com a física, ou seja, o tecido de seda é o que apresenta mais dobras, depois o algodão, o *jeans* e a lona. Este comportamento é reflexo dos valores usados na definição dos materiais. Pode-se destacar entre estes valores a influência do parâmetro $\xi_{\alpha\beta}$ na formação das dobras, determinando a suavidade com que elas se formam. Assim, valores altos de $\xi_{\alpha\beta}$ na definição do material tendem a formar dobras bem suaves no sentido de serem dobras armadas, como as dobras do *jeans* e principalmente as da lona, enquanto valores baixos de $\xi_{\alpha\beta}$ produzem dobras acentuadas, como as produzidas pela seda e algodão.

Estes comportamentos distintos são manipulados de forma bem sensível através do valor de $\xi_{\alpha\beta}$, pois a seda, que é um extremo, tem o valor de 0.01, enquanto a lona, o outro extremo, 0.1.

O coeficiente $\theta_{\alpha\beta}$, assim como $\zeta_{\alpha\beta}$, é mantido com os mesmos valores nas quatro simulações da figura 5.4. Contudo, $\theta_{\alpha\beta}$ é de fundamental importância para quantidade de dobras, principalmente neste caso em que só há forças tangenciais de compressão. O valor deste coeficiente elástico determina a frequência com que as dobras são formadas com deslocamentos da métrica. Apesar deste parâmetro ser fixado em 0.1 nos quatro *bucklings* de panos, cada material possui um grau de facilidade diferente para a formação de dobras. O uso de outros valores de $\theta_{\alpha\beta}$ resultaria em resultados com dobras ainda mais diferenciadas, porém, devido a sua sensibilidade numérica maior em relação a $\xi_{\alpha\beta}$, optou-se por utilizar o mesmo fator de criação de dobras para mostrar que somente com $\xi_{\alpha\beta}$ é possível ter controle sobre a suavidade das dobras formadas. O importante é que o valor do coeficiente $\theta_{\alpha\beta}$ não seja nulo, pois neste caso nenhum dos quatro panos sofreriam deformações nestas simulações.

De forma similar às imagens dos panos amassados, as gravatas em formato de formato borboleta da figura 5.5 foram geradas apenas com valores distintos da massa do material e do coeficiente elástico $\theta_{\alpha\beta}$. Sem a presença da gravidade e utilizando forças pontuais nos mesmos pontos da malha discreta, nota-se que a imagem da figura 5.5(a) se curva em relação ao seu centro com dobras mais arredondadas nas bordas do que a figura 5.5(b), que possui uma resistência maior à curvatura.

As dobras criadas no caimento de tecidos sobre mesas, como as apresentadas na figura 5.7, são de resolução pequena em relação aos panos amassados, por exemplo, mas mesmo assim é possível encontrar uma configuração parecida com as situações das fotos reais. Os valores dos parâmetros de material do linho quanto os de cetim são valores intermediários entre a seda e o *jeans*. O que acontece muitas vezes com os tecidos é que os fios que os formam são compostos de mais de um tipo de fibra. Atualmente, é fácil encontrar uma camiseta feita 50% de algodão e 50% de poliéster, que é feita deste modo para diminuir custos e também permitir que ela amasse menos do que uma feita 100% de algodão. A figura 5.8(a) ilustra outra possível variação de material alterando apenas o coeficiente $\xi_{\alpha\beta}$, enquanto na figura 5.8(b) o material do tecido é definido de forma bem mais rígida e suas dobras são bem armadas. O coeficiente $\theta_{\alpha\beta}$ é mantido com o valor 0.01 nestas simulações de caimento pois não tem tanta influência visual como nas simulações de *buckling*.

Com relação à poltrona da figura 5.9, foi considerado como material um tecido de veludo, caindo sob a força de gravidade. O veludo é configurado com valores próximos aos do algodão, mas com uma massa um pouco maior. Nesta simulação pode-se ver que as dobras formadas, principalmente na região próxima do assento da poltrona, são dobras mais armadas.

As cortinas da figura 5.10 foram geradas com malhas bastante refinadas. Assim como os panos amassados, a superfície delas possui 14400 pontos, porém elas utilizam uma grade discreta de $240 \times$

80 ao invés de 120×120 para representar o formato de uma cortina de maneira mais apropriada. Novamente, mudando somente a resistência da curvatura do tecido e as massas dos materiais pode-se obter comportamentos bem distintos, desde a cortina da figura 5.10(a) que é bem flexível até a cortina de banheiro da figura 5.10(c), sendo mais rígida e apresentando dobras suaves.

A bandeira feita na figura 5.11 também é uma malha pequena, de material leve e maleável, que com o uso da força de fluido consegue atingir dobras bem irregulares ao longo de todo o tecido, as quais se formam e se desfazem periodicamente.

5.4 Considerações Finais

Este capítulo apresentou os principais resultados obtidos com o simulador de tecidos desenvolvido com o modelo proposto por de Melo [4] utilizando o método de Verlet para integrá-lo em relação ao tempo. A seção 5.1 apresentou as metodologias usadas no desenvolvimento das simulações, mostrando de que forma são calibrados os parâmetros do material e também as forças externas que são definidas pelo usuário como restrições. Ao final desta seção, um procedimento foi fornecido para o auxílio no ajuste de algum tipo de material específico. A seção 5.2 mostrou os resultados visuais deste trabalho, dividindo-os em categorias relativas ao tipo de deformação que os tecidos sofrem, como *bucklings*, *drapings* e *hangings*, que podem ser traduzidos como enrugamentos, caimentos e suspensões de tecidos. Por último, a seção 5.3 analisou os resultados obtidos, dividindo a análise em termos da precisão física na dinâmica de tecidos, usando seqüências de imagens, e qualidade das dobras formadas, associando os parâmetros de material com os tipos de dobras formadas.

Com o término deste capítulo pode-se concluir que a abordagem de simular tecidos proposta, a qual reduz o modelo baseado na mecânica de contínuos em sistemas de partículas e solucioná-lo com o método de Verlet, é adequada para produzir comportamentos que apresentem realismo com uma complexidade computacional linear. Este paradigma de modelagem e animação de tecidos permitiu validar praticamente as principais características do modelo de tecidos proposto por Wu e Melo [3], como produzir dobras sob forças de compressão naturalmente. A precisão física do modelo também foi validada com a dinâmica de tecidos, mostrando um comportamento adequado do tecido quando ele atinge um estado de equilíbrio. Além disso, o controle intuitivo dos parâmetros do modelo permitiram formar uma grande variedade de dobras, que foram geradas em malhas de grande resolução espacial para avaliar o realismo atingido. A análise das dobras formadas mostrou que resultados com qualidade podem ser obtidos usando-se malhas densas com o uso do método de Verlet.

Capítulo 6

Conclusões e Trabalhos Futuros

Este capítulo finaliza este trabalho apresentando as principais conclusões sobre ele e indicando os possíveis trabalhos futuros. A principal conclusão foi que é possível aplicar em um modelo de tecidos baseado na superfície de Cosserat, na qual aparecem grandezas diferenciáveis, um método de integração explícita. Usando esta abordagem, além de se obter animações realistas e condizentes com a física, pode-se agregar também desempenho computacional. Com os resultados visuais do simulador implementado pôde-se notar que a qualidade das dobras formadas contribui de maneira significativa para o aumento da realidade no movimento de tecidos. Foi possível ver também a resposta apropriada da superfície do tecido quando ela é submetida à forças de compressão. Com o gráfico do tempo de processamento relacionado com o número de pontos da superfície, viu-se na figura 3.17 que o comportamento do sistema é linear, implicando em maior eficiência computacional. Desta forma, pode-se dizer que o objetivo deste trabalho foi atingindo, pois a abordagem alternativa proposta para simular a dinâmica de tecidos, após ser estudada, implementada e analisada, mostrou-se promissora. A seguir são retomados os principais pontos apresentados em cada capítulo para tecer pequenas conclusões e finalmente apresentar as opções de desdobramento deste trabalho.

Conforme foi afirmado no capítulo 1, a simulação de tecidos pode ser feita com um modelo de tecido integrado ao longo do tempo. A modelagem do tecido corresponde a uma formulação física, que pode ser baseada na mecânica de partículas ou na mecânica de contínuos. Algebricamente, ambas as técnicas podem ser descritas com o uso de equações diferenciais ordinárias em relação ao tempo. Já os métodos numéricos de integração podem ser explícitos, implícitos ou semi-implícitos. Atualmente, os modelos baseados em partículas junto com métodos semi-implícitos são os mais utilizados para simular tecidos, porém, este trabalho usou uma abordagem alternativa a esta combinação convencional objetivando um melhor desempenho. Utilizando um modelo de tecido preciso, fisicamente embasado e teoricamente comprovado para produção de dobras sob forças de compressão [4], este trabalho foi dedicado a investigar qual a melhor técnica de integração explícita que pode ser adaptada

a ele para produzir o movimento de tecidos. A idéia é manter o realismo e a precisão física do modelo e aumentar seu desempenho computacional.

Como pôde-se ver no capítulo 2, essa abordagem alternativa proposta preenche uma lacuna existente na literatura em relação aos trabalhos voltados à simulação de tecidos. Devido ao sucesso da abordagem que une a mecânica de partículas com métodos semi-implícitos de integração, o paradigma proposto ainda não tinha sido testado de forma profunda. Com este trabalho foi possível verificar que esta é uma interessante abordagem para quem precisa simular a dinâmica peculiar de tecidos com um custo computacional linear. Neste capítulo também foi apresentado de forma detalhada o modelo de tecido proposto por de Melo [4], que baseado na teoria geral da superfície de Cosserat [25] faz simplificações específicas para tecidos que conseguem atingir uma resposta natural da superfície quando submetida às forças de compressão. A escolha deste modelo deve-se principalmente ao fato dele considerar o acoplamento entre as deformações referentes à métrica e à curvatura da superfície que, segundo seus autores, é fundamental para a formação de dobras sob compressão.

No capítulo 3 foram apresentados cinco métodos numéricos de integração explícita que podem ser aplicados no modelo de tecido escolhido. A formulação de cada método permitiu analisar de forma teórica o número de cálculos e passos demandados, bem como a ordem de precisão numérica relacionada ao termo do erro de truncamento. Contudo, todos estes métodos precisavam ser testados com o modelo de tecido para verificar na prática qual era a melhor opção para se utilizar. Os trabalhos relacionados com a simulação de tecidos e que usam os métodos numéricos apresentados também foram discutidos neste capítulo, destacando a importância da proposta deste trabalho.

Ainda no capítulo 3, foram mostrados os resultados das simulações de teste aplicadas a todos os métodos, sendo possível concluir que os mais eficientes computacionalmente são os métodos de Euler, *Leapfrog* e Verlet, enquanto o método do ponto intermediário e o de Runge-Kutta de quarta ordem demandam um tempo de processamento bem superior. Quanto à estabilidade numérica, no sentido de qual método suporta um passo de tempo maior de forma estável, todos os métodos se mantiveram no mesmo patamar para cada simulação de teste. Investigando o motivo deste resultado, concluiu-se que o uso de vetores normais, estimados no instante $t - \Delta t$, para determinar as forças elásticas internas no instante t , conforme mostra o algoritmo 2 da seção 4.1, faz com a suavidade na variação destes vetores tenha maior influência no passo de tempo do que o método numérico. Já para o realismo obtido com cada método concluiu-se que como a qualidade das simulações eram muito próximas, a análise de qual é a melhor passa ser subjetiva e, portanto, deu-se prioridade para o desempenho na escolha do método que gerou os resultados deste trabalho. Neste sentido, foi escolhido o método de Verlet, por agregar de forma balanceada eficiência computacional com precisão numérica, além de na teoria ser também o melhor método em termos de custo em relação ao seu benefício.

No capítulo 4 foi apresentada uma implementação do modelo de tecidos com uso de técnicas

de integração explícita, construindo uma interface lógica de programação e também uma interface gráfica que permitisse ao usuário manipular poucos parâmetros para obter as simulações. Adotando o paradigma orientado a objetos, este capítulo descreveu a modelagem de dados do sistema que o separou em dois módulos principais, sendo um responsável pela dinâmica da superfície e outro pela visualização das superfícies geradas. Quanto à interface gráfica, ela foi desenvolvida tanto para a plataforma Linux quanto para a Windows, em ambas usando a biblioteca gráfica OpenGL e a linguagem de programação C++. Algumas simulações de exemplo foram explicadas para mostrar a boa usabilidade do simulador para obter resultados expressivos.

A facilidade de uso do simulador desenvolvido permitiu ainda validar na prática as propriedades do modelo de tecidos selecionado, que até então foram somente demonstradas teoricamente. No capítulo 5 foram detalhados os testes de validação e os resultados obtidos. Antes dos resultados, foi dada uma explicação sobre como calibrar os coeficientes elásticos do material do tecido e as forças externas que agem sobre sua superfície, fornecendo um procedimento de cinco passos para este propósito. Os resultados foram divididos em três categorias, que correspondem aos tipos de deformação causada nos tecidos. *Buckling* é o termo associado ao ato de amassar, enrugar, ou simplesmente criar dobras em tecidos, já *draping* é o caimento de tecidos sobre objetos e, *hanging*, a suspensão de tecidos. Com os resultados deste capítulo foi possível concluir que a qualidade e diversidade das dobras formadas está diretamente associada ao número de pontos da superfície, como também que a dinâmica mostrada pela sequência de imagens atinge estados de equilíbrio de forma apropriada. Além disso, observamos que o comportamento linear do algoritmo com a entrada do número de pontos da malha do tecido é preservado. Portanto, é possível simular malhas pequenas em tempos real, propiciando o seu emprego em jogos.

Em relação aos desdobramentos deste trabalho, pode-se indicar alguns trabalhos futuros. O primeiro deles é a inclusão de algoritmos que façam o tratamento de colisões e auto-colisões que acontecem na dinâmica dos tecidos. Este é um incremento fundamental para se atingir maior realismo como também validar a aplicação destes algoritmos com a proposta deste trabalho. Para tanto, a interface lógica criada neste trabalho visa facilitar a integração do processo referente às colisões, separando de forma apropriada a dinâmica interna de tecidos como sugere Bridson et al. [24]. Além disso, os diferentes métodos numéricos implementados podem ser de grande utilidade para investigar qual a melhor técnica de colisão que pode ser adaptada a este modelo. É interessante lembrar que alguns métodos numéricos apresentados, como o método de Verlet, possui variações bem específicas para contemplar os algoritmos de colisões que contam com passos de tempo adaptativos [24, 10].

Além da incorporação das colisões, uma das prioridades na continuidade deste trabalho é permitir o uso de qualquer tipo de malha triangular. Com isto, uma grande diversidade de objetos poderiam ser importados no simulador implementado e deformados como tecidos, retirando a limitação deste

trabalho em só deformar planos. Para desenvolver esta generalização na geometria da superfície do tecido é necessário modificar a maneira de calcular as forças internas e a discretização espacial feita com as diferenças finitas. Neste caso, as direções principais de cada ponto, que correspondem às direções dos vetores de base, poderiam ser atualizadas em cada iteração, como faz Rusinkiewicz [39], para que a deformação da superfície não influencie a precisão dos vetores de base. Apesar de demandar maior esforço computacional, esta modificação pode ampliar o domínio das geometrias processáveis.

A discretização no espaço do modelo de tecido proposto por Wu e de Melo [4] pode ser feita também com a técnica dos elementos finitos para comparar com a qualidade obtida em relação a das diferenças finitas, usada neste trabalho. Esta técnica foi usada recentemente por Bergou et al. para discretizar um o modelo de superfícies inextensíveis deformáveis, apresentando uma implementação eficiente computacionalmente [7].

Embora o modelo de tecidos usado neste trabalho possua parâmetros de material com significados geométricos intuitivos que auxiliam a definição do seu tipo, os valores dos coeficientes elásticos dependem de medidas como a massa e área do tecido. Estas relações de dependência podem ser investigadas profundamente para conseguir uma simulação mais precisa dos parâmetros mecânicos do tecido, normalizando o domínio dos coeficientes elásticos para não produzir os mesmos resultados com configurações múltiplas de outras. Isto permitiria o mapeamento automático de materiais, de forma que o usuário pudesse produzir simulações apenas selecionando um tipo de tecido entre uma lista e o simulador determinasse os coeficientes corretos para produzir o comportamento do material.

Outra possibilidade de trabalhos futuros seria integrar algoritmos de corte e costura sobre os tecidos, permitindo que eles possam formar estruturas mais complexas e que possam até ser usados por projetistas relacionados à moda. Um último trabalho futuro desafiador seria estender este simulador para realizar animações em tempo real, pois com isso pode-se pensar em aplicar as simulações até para jogos. Com o recurso feito neste trabalho de visualizar os quadros à medida que eles forem calculados, chamado de modo *on-the-fly*, viu-se que para malhas pequenas, compostas de 400 pontos aproximadamente, isso já é possível, contudo, para malhas maiores, as quais apresentam mais qualidade visual, o simulador teria que ser bem modificado para otimizar as animações. Um passo considerável neste sentido seria a implementação da alternativa proposta diretamente na GPU.

Referências Bibliográficas

- [1] Haumann, D. R. Modeling the physical behavior of flexible objects. In *ACM SIGGRAPH 87 Course Notes 17 - Topics in Physically-Based Modeling*, Anaheim, California, July 1987.
- [2] Provot, X. Deformation constraints in a mass-spring model to describe rigid cloth behavior. In *Proceedings of Graphics Interface*, pages 147–154, May 1995.
- [3] Wu, S.-T. and Melo, V. F. A deformable surface model on the basis of the theory of a cosserat surface. In *SIBGRAPI 2004 conference proceedings*, 2004.
- [4] Melo, V. F. Modelagem e controle de caimento e dobras em superfícies deformáveis. Tese de doutorado, Faculdade de Engenharia Elétrica e de Computação, UNICAMP, 2004.
- [5] Goldenthal, R. and Harmon, D. and Fattal, R. and Bercovier, M. and Grinspun, E. Efficient Simulation of Inextensible Cloth. In *SIGGRAPH (ACM Transactions on Graphics)*, 2007.
- [6] Thomaszewski, B. and Wacker, M. and Straßer, W. A consistent bending model for cloth simulation with corotational subdivision finite elements. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics symposium on Computer Animation*, pages 107–116, 2006.
- [7] Bergou, M. and Wardetzky, M. and Harmon, D. and Zorin, Z. and Grinspun, E. A quadratic bending model for inextensible surfaces. In *Eurographics Symposium on Geometry Processing*, 2006.
- [8] Eitzmuss, O. and Keckeisen, M. and Straßer, W. A fast finite element solution for cloth modeling. In *Proceedings of the Pacific Conference on Computer Graphics and Applications*, pages 244–252, 2003.
- [9] Baraff, D. and Witkin, A. and Kass, M. Untangling cloth. In *ACM Transactions on Graphics*, volume 22(3), pages 862–870, July 2003.
- [10] Bridson, R. and Marino, S. and Fedkiw, R. Simulation of clothing with folds and wrinkles. In *Eurographics / SIGGRAPH Symposium on Computer Animation*, 2003.

- [11] Choi, K.-J. and Ko, H.-S. Stable but responsive cloth. In *ACM Transactions on Graphics (SIGGRAPH Proceedings)*, volume 21, pages 604–611, 2002.
- [12] Volino, P. and Thalmann, N. M. Fast geometric wrinkles on animated surfaces. In *WSCG 99*, pages 175–183, Plzen, Feb 1999.
- [13] Baraff, D. and Witkin, A. Large steps in cloth simulation. In *Proceedings of SIGGRAPH 98*, pages 43–54, 1998.
- [14] Eberdardt, B. and Weber, A. and Strasser, W. A Fast, flexible, particle-system model for cloth draping. In *Computer Graphics in Textiles and Apparel (IEEE Computer Graphics and Applications)*, pages 52–59, Sept 1996.
- [15] Eischen, J.W and Deng, S. and Clapp, T.G. Finite-element modeling and control of flexible fabric parts. In *IEEE Computer Graphics and Applications*, volume 16(5), pages 71–80, 1996.
- [16] Breen, David E. and House, Donald H. and Wozny, Michael J. A physically based particle model of woven cloth. In *The Visual Computer*, volume 8(5-6), pages 264–277, 1992.
- [17] Simo, J.C. and Fox, D.D. On a stress resultant geometrically exact shell model, part I: formulation and optimal parameterization. In *Computer Methods in Applied Mechanics and Engineering*, volume 72, pages 267–304, 1989.
- [18] Terzopoulos, D. and Platt, J. and Barr, A. and Fleischer, K. Elastically Deformable Models. In *Computer Graphics (SIGGRAPH proceedings)*, volume 21(4), pages 205–214, July 1987.
- [19] Feynman, C. Modeling the appearance of cloth. M.s. thesis, Massachusetts Institute of Technology, Massachusetts, 1986.
- [20] Weil, J. The synthesis of cloth objects. In *Computer Graphics (SIGGRAPH proceedings)*, pages 49–54, 1986.
- [21] Decaudin, P. and Julius, D. and Wither, J. and Boissieux, L. and Sheffer, A. and Cani, M.P. Virtual garments: a fully geometric approach for clothing design. In *EUROGRAPHICS 2006*, volume 25(3), 2006.
- [22] Breen, D. and House, D. *Cloth modeling and animation*. AK Peters, 2000.
- [23] Eitzmuss, O. and Gross, J. and Straßer, W. Deriving a particle system from continuum mechanics for the animation of deformable objects. In *IEEE Transactions on Visualization and Computer Graphics*, volume 9(4), pages 538–550, 2003.

- [24] Bridson, R. and Fedkiw, R. and Anderson, J. Robust treatment of collisions, contact and friction for cloth animation. In *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 594–603, 2002.
- [25] Green, A. E. and Naghdi, P. and Wainwright, W. A general theory of a cosserat surface. *Archive for Rational Mechanics and Analysis*, 20:287–308, 1965.
- [26] Breen, D. E. and House, D. H. and Wozny, M. J. Predicting the drape of woven cloth using interacting particles. In *Computer Graphics (SIGGRAPH proceedings)*, pages 365–372. Addison-Wesley, July 1994.
- [27] Smith, G. D. *Numerical solution of partial differential equations: finite difference methods*. Oxford University Press, 3rd edition, 1985.
- [28] Press, W. H. and Teukolsky, S. A. and Vetterling, W. T. and Flannery, B. P. *Numerical recipes in c*. Cambridge University Press, 2nd edition, 1992.
- [29] Volino, P. and Courchesne, M. and Magnenat-Thalmann, N. Versatile and efficient techniques for simulating cloth and other deformable objects. In *Computer Graphics (SIGGRAPH proceedings)*, pages 137–144. Addison-Wesley, 1995.
- [30] Volino, P. and Magnenat-Thalmann, N. Developing simulation techniques for an interactive clothing system. In *Virtual Systems and Multimedia (VSMM proceedings)*, pages 109–118, Geneva, Switzerland, 1997.
- [31] Oshita, M. and Makinouchi, A. Real-time cloth simulation with sparse particles and curved faces. In *Proceedings of Computer Animation*, Seoul, Korea, November 2001.
- [32] Volino, P. and Magnenat-Thalmann, N. Comparing efficiency of integration methods for cloth simulation. In *Computer Graphics International Proceedings*, pages 265–274, Plzen, Feb 2001.
- [33] Thomas Jakobsen. Advanced Character Physics. Página na internet, IO Interactive, Farvergade 2, 2001. <http://www.teknikus.dk/tj/gdc2001.htm>.
- [34] Jonathan Dummer. A simple time-corrected verlet integration method. Página na internet, GameDev.net, 2005. <http://www.gamedev.net/reference/articles/article2200.asp>.
- [35] Michael Hauth. Numerical Techniques for Cloth Simulation. In *SIGGRAPH 2003 Course 29: Clothing Simulation and Animation*, October 2003.

- [36] Suriya Natsupakpong. Numerical integration method in deformable object simulation. Página na internet, Electrical Engineering and Computer Science, Case Western Reserve University, 2004. <http://filer.case.edu/~sxn66/spring2004/eecs466/Report.htm>.
- [37] Wu, S.-T. Deformable surface modeler. Página na internet, Electrical Engineering and Computer Science Faculty, University of Campinas, 2004. <http://www.dca.fee.unicamp.br/projects/desmo/videos>.
- [38] Monteiro, L.P. Deformable Surface Modeler. Página na internet, Electrical Engineering and Computer Science Faculty, University of Campinas, 2004. <http://www.dca.fee.unicamp.br/projects/desmo/pinho>.
- [39] Rusinkiewicz, S. Estimating curvatures and their derivatives on triangle meshes. In *Proceedings of the 2nd International Symposium on 3D Data Processing, Visualization, and Transmission*, 2006.