

MATHEUS DE MORAES CAVAZOTTI

UM ESTUDO SOBRE REPRESENTAÇÃO E SIMULAÇÃO DE PELOS E CABELOS EM
COMPUTAÇÃO GRÁFICA

(versão pré-defesa, compilada em 26 de maio de 2022)

Trabalho apresentado como requisito parcial à conclusão do Curso de Bacharelado em Ciência da Computação, Setor de Ciências Exatas, da Universidade Federal do Paraná.

Área de concentração: *Ciência da Computação*.

Orientador: André Luiz Pires Guedes.

CURITIBA PR

2022

RESUMO

Este trabalho explica o que é representação de pelos e cabelos no contexto de computação gráfica, discorre sobre quais são seus principais desafios e comenta as soluções que surgiram ao longo dos anos. O presente trabalho também se propõe a descobrir se a representação dos fios de cabelo como sistemas massa-mola é adequada para simulações interativas em tempo real por meio de implementação e experimentação prática. Os resultados, limitações e possibilidades de melhoria são discutidos.

Palavras-chave: Computação Gráfica. Simulação Física. Animação Baseada em Física

ABSTRACT

The present work explains what constitutes representation and simulation of hair and fur in Computer Graphics, discourses about its main challenges and reviews some of the solutions that were created over the years. This work also seeks to discover if mass-spring systems are adequate for real-time interactive hair simulation by implementing and practical testing. The results, limitations and possible enhancements thereof are presented. Keywords: Computer

Graphics. Physics Simulation. Physics-based Animation.

LISTA DE FIGURAS

1.1	Exemplos de diferentes tipos de cabelos e pelos, em diferentes condições, em personagens gerados por computador.	8
3.1	Cilindro coberto com fios de cabelo	13
3.2	Exemplos de cabelos obtidos pelo método de Bertails et al. (2006)	13
3.3	Exemplo de modelo proposto por Liang e Huang (2003)	14
3.4	Uma malha complexa e dois penteados derivados dela.	15
3.5	Água fluindo através de um tapete de pelos.	16
4.1	Esquema de sistema massa-mola de um fio de cabelo.	19
5.1	Exemplo de cobertura de pelos gerada pelo simulador	23
5.2	Simulação interativa em tempo real	25
5.3	Exemplo de erro de convergência.	26

LISTA DE TABELAS

3.1	Resumo das técnicas apresentadas	17
5.1	Comparação da carga de trabalho máxima do simulador sequencial e paralelo . .	24

SUMÁRIO

1	INTRODUÇÃO	7
2	CABELOS E PELOS EM COMPUTAÇÃO GRÁFICA	9
2.1	ETAPAS NA REPRESENTAÇÃO GRÁFICA DE CABELOS	9
2.2	COMO E O QUÊ DEVE SER REPRESENTADO.	9
2.3	CONCLUSÃO	11
3	TRABALHOS RELACIONADOS	12
3.1	CABELO COMO FIOS INDIVIDUAIS	12
3.2	CABELO COMO VOLUME OU SUPERFÍCIE	14
3.3	SOLUÇÕES PARA PROBLEMAS ESPECÍFICOS	15
3.4	CONCLUSÃO	16
4	IMPLEMENTAÇÃO	18
4.1	DESCRIÇÃO DO MODELO COMPUTACIONAL	18
4.2	IMPLEMENTAÇÃO SEQUENCIAL	19
4.3	IMPLEMENTAÇÃO PARALELA	21
4.4	CONCLUSÃO	22
5	AValiação EXPERIMENTAL.	23
5.1	AMBIENTE DE TESTES	23
5.2	RESULTADOS	24
5.3	DESAFIOS E LIMITAÇÕES.	25
5.4	MELHORIAS FUTURAS	26
5.5	CONCLUSÃO	27
6	CONCLUSÃO	28
	REFERÊNCIAS	29
	APÊNDICE A – LISTA DE ARTIGOS SOBRE RENDERIZAÇÃO DE	
	CABELOS E PELOS	30
	APÊNDICE B – IMPLEMENTAÇÃO PARALELA EM <i>UNITY</i>	31

1 INTRODUÇÃO

Um dos pontos cruciais para se gerar imagens realistas de seres humanos e animais (principalmente mamíferos) é ter a habilidade de representar fielmente cabelos e pelos. Isso não é uma tarefa computacionalmente trivial, visto que uma cabeça humana tem, em média, de 100.000 a 120.000 fios de cabelos, cada um dos quais medindo entre 40 a 120 μ m de diâmetro (Rosenblum et al., 1991). Ademais, os fios estão continuamente colidindo entre si e frequentemente sofrem a ação de agentes externos, como a resistência do ar, o vento e outras substâncias.

Apesar de esse ser um problema complexo e repleto de nuances, ele é extremamente comum nas indústrias de efeitos especiais, animação, jogos eletrônicos e mídias interativas (ver Figura 1.1). Os cabelos (a partir de agora será omitido o termo "pelo", a título de brevidade), além de serem uma parte natural da anatomia de muitos seres vivos, também podem ser elementos importantes no desenvolvimento da narrativa de uma história ou de um personagem. Sendo assim, a pesquisa e desenvolvimento de novas técnicas para representar e simular cabelos expande o horizonte de possibilidades dos artistas no processo de criar narrativas visuais.

Este trabalho tem três objetivos: descrever – com mais detalhes – os desafios da representação gráfica da aparência e do comportamento de cabelos; apresentar algumas das técnicas e algoritmos que foram desenvolvidos ao longo dos anos para vencer ou contornar tais desafios, comentando seus pontos fortes e fracos; e, por fim, mostrar uma implementação de simulador de cabelos que adapta a ideia de Rosenblum et al. (1991) para o contexto de GPUs ¹ e avaliar se tal implementação permite simulações interativas em tempo real.

Sobre a estrutura deste trabalho: no Capítulo 2 será explicado com mais detalhes o que constitui o problema de representar e simular cabelos; no Capítulo 3 serão apresentados os principais algoritmos e técnicas que foram desenvolvidos ao longo dos anos, comentando seus pontos fortes e fracos; no Capítulo 4 será descrito com detalhes o algoritmo e a implementação de um simulador de cabelos; no Capítulo 5 será avaliado o desempenho da implementação e se discorrerá sobre possibilidades de melhorá-la. Finalmente, no Capítulo 6 serão feitas as observações finais sobre o presente trabalho.

¹Graphics Processing Unit

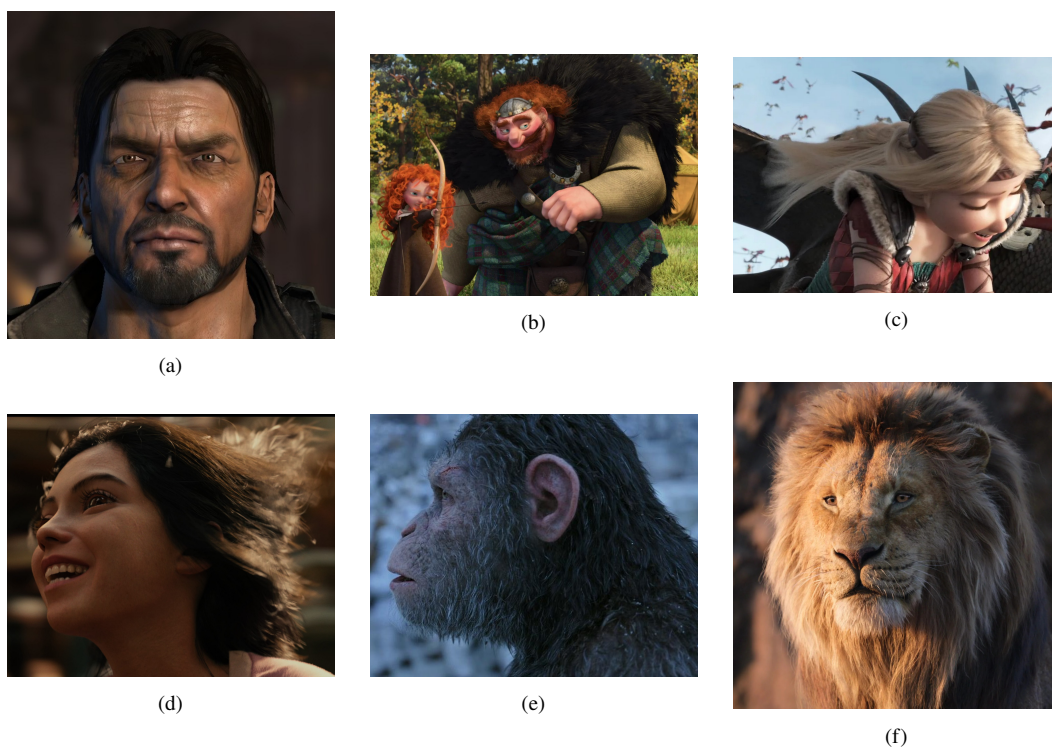


Figura 1.1: Exemplos de diferentes tipos de cabelos e pelos, em diferentes condições, em personagens gerados por computador. **1.1(a)**: *StarCraft 2* (2010), jogo de computador da Blizzard Entertainment. **1.1(b)**: *Valente* (2012), filme de animação da Pixar. **1.1(c)**: *Como Treinar o Seu Dragão 3* (2019), filme de animação da DreamWorks Animation. **1.1(d)**: *Alita: Anjo de Combate* (2019), filme *live action* da Twentieth Century Fox. **1.1(e)**: *Planeta dos Macacos: A Guerra*, (2017), filme *live action* da Twentieth Century Fox. **1.1(f)**: *O Rei Leão*, filme de animação da Walt Disney Pictures.

2 CABELOS E PELOS EM COMPUTAÇÃO GRÁFICA

Como mencionado no capítulo anterior, representar cabelos virtuais graficamente é um processo complexo. Existem muitos aspectos a se considerar para que a representação gráfica atenda às necessidades. Alguns desses aspectos serão expostos neste capítulo.

Além disso, para se representar cabelos são necessárias algumas etapas nas quais são realizadas atividades específicas. Tais etapas serão explicadas à seguir.

2.1 ETAPAS NA REPRESENTAÇÃO GRÁFICA DE CABELOS

Para se obter imagens geradas por computador (em inglês *Computer Generated Imagery* - CGI) com cabelos realistas, muitas vezes é necessário pelo menos três etapas: **modelagem**, **simulação** e **renderização**¹. Modelagem é o processo no qual se estabelece a aparência e a disposição geral do cabelo, o que inclui sua área de cobertura, comprimento e penteado. O tipo de cabelo (liso, ondulado, cacheado, etc.) e efeitos decorrentes de circunstâncias externas (água, vento, etc.) geralmente não definidos nessa etapa, mas nas etapas posteriores.

A etapa de simulação consiste em reproduzir o comportamento e a aparência do cabelo decorrentes de situações naturais, como a estrutura interna dos fios de cabelo que determina se ele será liso ou crespo, o movimento dos fios causados pela inércia, colisão ou vento, ou a coalescência causada pela interação dos fios com líquidos.

Renderização é o processo de gerar uma imagem a partir de objetos virtuais. No contexto de cabelos, o processo de renderização consiste em descrever a interação da luz com o cabelo, tentando levar em consideração a refletância, absorção e outros fenômenos óticos que ocorrem com as centenas de milhares de fios por indivíduo, a fim de transcrevê-la em *pixels*.

Essa última etapa é uma ciência em si só, com muitos desafios e características próprias. Por conta disso, este trabalho foca apenas nas etapas de modelagem e simulação, no entanto o Apêndice A traz uma relação de trabalhos que abordam esse tema.

2.2 COMO E O QUÊ DEVE SER REPRESENTADO

Como mencionado anteriormente, representar cabelos é uma tarefa complexa. No entanto, para a computação gráfica, o objetivo de tais representações é apenas *parecer* realista, sem haver a necessidade de que sejam fisicamente precisas.

¹Em fluxos de trabalho reais são mais etapas: primeiro é preciso preparar o personagem no qual será implantado o cabelo, só depois vem a modelagem do cabelo e a simulação. Antes de renderizar, é comum existir uma etapa de ajustes artísticos sobre o resultado da simulação. Esses ajustes podem incluir acrescentar detalhes no penteado ou realizar alterações para que o resultado esteja de acordo com a visão artística. Após a renderização, há a etapa de composição e pós-produção, que realiza ajustes e correções na imagem. Só então pode-se dizer que a imagem está pronta.

Isso permite flexibilidade nas abstrações no sistema (entenda-se sistema como os fios de cabelo, o penteado e tudo o mais que interagir com eles) a fim de atender as diversas necessidades e restrições. Essas abstrações incluem, mas não se limitam a: escolher um modelo físico simplificado para governar a dinâmica do cabelo, tratar o cabelo como um volume único, ou usar um número muito menor de fios para a modelagem.

Após a realização de estudos, concluiu-se que alguns dos principais aspectos que devem ser considerados ao buscar e desenvolver métodos para a representação gráfica de cabelos são:

1. **Quais são os tipos de cabelos que serão representados?** Cabelos lisos, ondulados, cacheados e crespos apresentam comportamentos dinâmicos distintos. Além disso, deve-se considerar outros fatores que alteram a aparência e a dinâmica dos cabelos, tais como frisado, sujeira, produtos fixadores e água.
2. **Qual é o nível desejado de controle sobre o penteado?** Nos casos em que a forma do penteado é importante para o produto final, é necessário que a técnica a ser utilizada incorpore isso em suas simulações. Além disso, se os penteados forem complexos, a facilidade de modelagem é outro aspecto a se considerar.
3. **O objetivo final é gerar uma imagem estática ou uma animação?** Se for o primeiro caso, técnicas que focam na modelagem do cabelo e não realizam simulações físicas complexas podem atender às necessidades. Já se o intuito for criar uma animação, simulações quase sempre serão necessárias.
4. **A simulação deve ocorrer em tempo real?** Se o tempo gasto para simular um *frame* for uma restrição, talvez técnicas que fazem aproximações – às vezes não tão precisas ou detalhadas – da dinâmica dos cabelos sejam melhores que abordagens que trazem resultados muito fieis à realidade.
5. **Quais são os recursos computacionais disponíveis?** Dependendo das restrições nesse aspecto, abordagens que buscam simular os fios individualmente podem ser inviáveis.
6. **Quais serão as interações que o cabelo irá sofrer?** O cabelo estará sujeito a movimentos bruscos? Ou muito vento? O cenário no qual ele será inserido implicam em colisões com outros objetos complexos? O cabelo estará imerso em água ou em algum outro fluido? O objeto que serve de âncora ("couro cabeludo") está sujeito a deformações?

Não existe uma única solução como o padrão da indústria que atenda a todos os cenários sugeridos acima (Ward et al., 2007): dependendo do propósito e das restrições, as abordagens mais adequadas e suas implementações podem variar muito.

2.3 CONCLUSÃO

Para gerar uma CGI com cabelos realistas, são necessárias três etapas: modelagem, que define a forma geral do cabelo; simulação, que trata de reproduzir interações e características físicas do cabelo; e renderização, que captura a interação da luz com o cabelo e a registra nos *pixels* de uma imagem.

Em cada uma dessas etapas é necessário considerar uma série de cenários, situações e características que envolvem o cabelo em si e o contexto no qual ele está inserido. Também é necessário ter em conta restrições relativas ao contexto em que será usada a imagem produzida.

Tudo isso influencia quão adequada uma técnica pode ser para uma dada situação. No próximo capítulo será mostrado como alguns trabalhos se propuseram a atender alguns dos cenários acima.

3 TRABALHOS RELACIONADOS

Existem muitas técnicas e abordagens para modelar, simular e renderizar cabelos e pelos. Neste capítulo serão mostrados alguns trabalhos cujo foco é a modelagem e simulação. A maior parte desses trabalhos não têm relação direta com a implementação que será descrita no próximo capítulo. Contudo, o objetivo de se expor e comentar várias técnicas é mostrar a diversidade de abordagens existentes, cada qual com seus pontos fortes e suas limitações.

É possível agrupar as diversas técnicas existentes em várias categorias. Essas categorias dizem respeito ao modo como o cabelo é representado computacionalmente, ou seja, sobre quais elementos os algoritmos vão operar. São algumas delas (ver Ward et al., 2007):

1. Técnicas em que os fios são tratados individualmente;
2. Técnicas nas quais o cabelo é representado como um volume sólido ou um conjunto de superfícies;
3. Técnicas que representam o cabelo como conjuntos adaptativos de mechas;
4. Técnicas nas quais o cabelo é tratado como um fluido ou um meio contínuo

Este capítulo apresentará trabalhos que se enquadram nas duas primeiras categorias: **cabelo como fios individuais** e **cabelo como volume ou superfície**, pois são as comuns na indústria. Neste capítulo, o termo "modelar" se refere à modelagem das estruturas de dados que são empregadas nas técnicas descritas, e não na modelagem do penteado em si.

A primeira abordagem (fios individuais) possui a modelagem mais próxima à natureza, que possibilita alcançar resultados, em geral, mais realistas e detalhados a um custo computacional maior. A segunda categoria (volumes e superfícies) é um modelo mais distante da natureza real do cabelo, mas que possui vantagens únicas.

Ao longo dos anos também foram desenvolvidas técnicas que modificam modelos computacionais já existentes a fim de solucionar desafios específicos. Este capítulo também trará dois artigos que exemplificam tais modificações.

3.1 CABELO COMO FIOS INDIVIDUAIS

Um dos primeiros trabalhos que descrevem uma técnica que aborda o cabelo como fios individuais é o de Rosenblum et al. (1991). Nele, a simulação da dinâmica de cada fio é modelada como um sistema massa-mola: o fio de cabelo é representado como uma sequência de partículas interligadas por molas. A influência das molas sobre as partículas é descrita pela lei de Hooke.

Essa técnica é simples de ser implementada e traz resultados visualmente realistas para condições amenas, isto é, condições nas quais não hajam movimentos ou colisões extremas.

Essa limitação se dá pelo fato de que a estrutura de molas que representa os fios não garante o comprimento imutável dos fios, logo, os fios podem se contrair ou esticar se forem submetidos à ação de forças suficientemente grandes.

Além disso, observou-se que o método de integração dessa técnica consiste em um sistema rígido – e, conseqüentemente, que pode ser numericamente instável –, o que limita a sua eficiência. Mais detalhes sobre esse fenômeno serão dados no Capítulo 5. A Figura 3.1 mostra um dos resultados obtidos pelos autores.



Figura 3.1: Cilindro coberto com fios de cabelo (Rosenblum et al., 1991)

Uma outra técnica que trata os fios individualmente foi proposta por Bertails et al. (2006). Nela, os fios são modelados como hélices de hastes elásticas e sua dinâmica é descrita por equações de Kirchhoff para hastes elásticas, que logram capturar o comportamento inextensível e não linear dos fios de cabelo (esse comportamento é exemplificado quando, ao torcionar-se um fio de cabelo o suficiente, ele começa a enrolar-se sobre si mesmo).

O trabalho de Bertails et al. (2006) propôs um novo método de integração das equações de Kirchhoff em relação ao tempo que é estável e é adequado para simular, de modo realista, o comportamento de cabelos lisos, ondulados e encaracolados. A Figura 3.2 mostra alguns dos tipos de cabelo que o método das "super-hélices", como os autores o chamam, consegue reproduzir.



Figura 3.2: Exemplos de cabelos obtidos pelo método de Bertails et al. (2006)

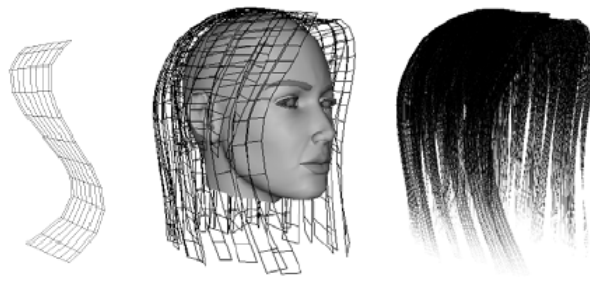


Figura 3.3: Exemplo de modelo proposto por Liang e Huang (2003). Uma fita de cabelo (esquerda), um modelo de penteado (centro), penteado texturizado com transparência (direita).

3.2 CABELO COMO VOLUME OU SUPERFÍCIE

Em Guang e Huang (2002) é descrito um método para representar cabelos curtos conforme a seguinte abordagem: dada uma malha como base para servir de couro cabeludo, é realizada a extrusão dos vértices dessa malha e, em seguida, são geradas faces internas ao volume do poliedro recém criado. Essas faces são mapeadas à uma imagem com transparência de fios de cabelo. Para adicionar dinamicidade ao penteado, a posição dos vértices extrudados ao longo do tempo pode ser governada por um sistema massa-mola que está sujeito a forças externas.

Esse método é adequado para cabelos curtos e lisos com penteados simples. A sobreposição da textura de fios de cabelo traz um certo grau de realismo. Do mesmo modo, a simulação da interação do penteado com forças externas traz resultados realistas em tempo real. No entanto, tal método parece não ser adequado para outros tipos de cabelo e penteados, e não parece prover alto nível de detalhes.

Seguindo a mesma abordagem, em Liang e Huang (2003) é proposto um método um pouco diferente. O penteado é modelado usando alguns "fios de controle", que servem como pontos de controle *NURBS*¹. É realizada, então, a tesselação das curvas (ou seja, a conversão da curva para uma malha poligonal), governada por esses "fios de controle", e são geradas "fitas" que, por sua vez, serão "dobradas" em forma de U e texturizadas de modo semelhante àquele descrito no método acima (ver Figura 3.3). A simulação do cabelo também se baseia em sistemas massa-mola: as fitas são tratadas como sequências de massas conectadas por molas. Esse método também propõe uma solução para simular auto-colisão e colisões externas.

Observa-se que esse método é adequado para cabelos lisos de qualquer tamanho. As fitas em forma de U dão maior realismo quando comparado com fitas retas. A simulação é feita em tempo real e parece ser adequada para situações nas quais o nível de detalhamento e o grau de realismo não são tão altos.

Uma técnica mais moderna que também parte do princípio de representar o penteado com superfícies e volumes foi apresentada em Yuksel et al. (2009). Nela, o artista modela o penteado como se estivesse construindo uma malha poligonal, tendo a possibilidade de criar penteados arbitrariamente complexos. Após essa fase de modelagem, os fios de cabelo são

¹*Non-uniform rational B-spline*. É um modelo matemático usado comumente para representar superfícies e curvas. A forma da superfície, ou curva, é determinada por uma série de pontos de controle.



Figura 3.4: Uma malha complexa e dois penteados derivados dela. (Yuksel et al., 2009)

gerados a partir do volume dos poliedros formados por essa malha. Na etapa de finalização, é possível configurar o grau de ondulação dos fios de cabelo e também adicionar detalhes, como frisado no cabelo. Essa última etapa permite que diferentes estilos visuais sejam criados a partir de uma mesma malha, conforme ilustrado na Figura 3.4.

Esse método traz um alto nível de controle e permite que penteados complexos e detalhados sejam criados com certa facilidade. A simulação da dinâmica do cabelo é semelhante a Guang e Huang (2002): os vértices da malha são tratados como massas de um sistema massa-mola. Isso possibilita simulações em tempo real, mas não parece ser adequado para representar interações complexas entre o cabelo e o ambiente (como, por exemplo, movimentos em alta velocidade, ventos fortes, etc.).

3.3 SOLUÇÕES PARA PROBLEMAS ESPECÍFICOS

Além das diferentes formas de modelar computacionalmente o cabelo, existem trabalhos que tratam de problemas específicos e que são aplicadas sobre técnicas como as descritas acima.

Em Iben et al. (2019) são propostas algumas técnicas para se manter a forma do penteado durante simulações, que podem ser aplicadas em modelos massa-mola semelhantes àquele apresentado por Rosenblum et al. (1991). A primeira proposta é modificar o parâmetro de deslocamento da lei de Hooke (o x em $F = -kx$) no que autores chamaram de "*gravity preloading*". Essa alteração faz com que o estado de repouso do sistema massa-mola seja o penteado definido pelo artista. A outra proposta nesse trabalho consiste em calcular um centroide em relação aos conjuntos de pontos (partículas) que devem permanecer próximos. Esse centroide é atualizado ao longo da simulação e serve de restrição (*constraint*) que, em conjunto com outros mecanismos, "puxa" os fios de cabelo para a posição desejada.

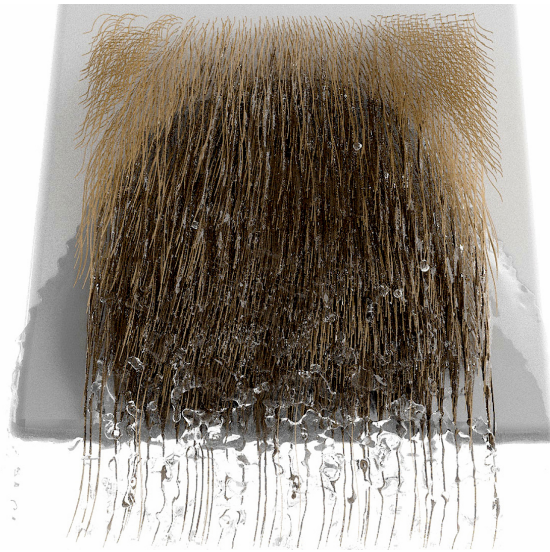


Figura 3.5: Água fluindo através de um tapete de pelos. (Fei et al., 2017)

No trabalho de Fei et al. (2017) é descrito um sistema de simulação fio a fio que suporta os vários fenômenos que correm nas interações entre cabelos e líquidos, tais como captura do líquido pelos fios de cabelo, fluxo de líquido ao longo do cabelo, maior coalescência dos fios e gotejamento. Além disso, esse sistema considera a influência do líquido na distribuição de massa ao longo dos fios e na conservação de energia e de quantidade de movimento. Nesse método, os fios de cabelo são modelados como hastes elásticas discretas, semelhante ao método de Bertails et al. (2006). As interações cabelo-líquido são computadas usando o método APIC² para simular líquido livre e um novo método proposto pelos autores para computar o comportamento do líquido nos próprios fios de cabelo. A Figura 3.5 mostra o nível de realismo obtido por esse sistema.

3.4 CONCLUSÃO

Pôde-se observar como diferentes técnicas para solucionar um mesmo problema (representação e simulação de cabelos) apresentam características distintas, cada qual com vantagens e desvantagens próprias. A Tabela 3.1 apresenta um resumo das abordagens para se modelar computacionalmente cabelos. No próximo capítulo será descrito com detalhes o método de simulação proposto por Rosenblum et al. (1991) e a implementação do mesmo.

²*Affine Particle-In-Cell*. Um método para simulação de fluidos.

Técnica	Tipos de cabelo	Capacidade de simulação	Simulação em tempo real
Massa-mola (Rosenblum et al., 1991)	Liso	Suporta movimentos complexos, mas não mantém comprimento constante	Não
"Super-hélice"(Bertails et al., 2006)	Liso, ondulado e encaracolado	Suporta movimentos complexos, mantém propriedades físicas de cabelos reais	Não
Faces texturizadas (Guang e Huang, 2002)	Liso e curto	Movimentos simples	Sim
Curvas NURBS (Liang e Huang, 2003)	Liso	Movimentos simples	Sim
Malha poligonal (Yuksel et al., 2009)	Liso, ondulado e encaracolado, com penteados complexos	Movimentos simples	Sim

Tabela 3.1: Resumo das técnicas apresentadas

4 IMPLEMENTAÇÃO

Para experimentar os conceitos apresentados anteriormente, foi desenvolvido um simulador de cabelos baseado no método proposto por Rosenblum et al. (1991), pois ele é simples de ser implementado.

Passaram-se cerca de 30 anos desde a publicação desse trabalho e, durante esse período, a capacidade computacional dos *hardwares* disponíveis para o consumidor aumentou muito. O objetivo do experimento era avaliar se era possível, através do uso do *hardware* disponível atualmente, chegar a uma simulação interativa em tempo real usando essa técnica.

4.1 DESCRIÇÃO DO MODELO COMPUTACIONAL

Cada fio de cabelo é representado como uma sequência de segmentos de reta. e cada vértice desses segmentos é tratado como uma partícula de massa m . Duas partículas adjacentes exercem força, uma sobre a outra, de acordo com a lei de Hooke acrescida de um termo D_m de amortecimento:

$$F_m = k_m x - D_m k_m x \quad (4.1)$$

Na Equação 4.1, x é a diferença entre a distância entre as partículas e a distância de repouso e k_m é a constante de elasticidade da mola que liga as duas partículas. Para simular a rigidez do fio, foram adicionadas "dobradiças elásticas" para cada 3 vértice consecutivos P_i , P_j e P_k (o vértice do meio, P_j serve de pivô). Essas dobradiças são descritas por uma versão angular da lei de Hooke:

$$F_d = k_d \theta - D_d k_d \theta \quad (4.2)$$

Onde θ é a diferença entre o ângulo $\angle P_i P_j P_k$ e o ângulo de repouso, k_d é a constante de elasticidade da dobradiça e D_d é uma constante de amortecimento. O esquema do sistema massa-mola que representa um fio de cabelo pode ser observado na Figura 4.1.

Além da força exercida pelas molas sobre as partículas, também foi considerada a atuação da força da gravidade e a resistência do ar. No entanto, essa última é representada apenas como um termo que induz à desaceleração, sem se ater a detalhes de aerodinâmica e dinâmica de fluidos:

$$F_r = -\nu D_r \quad (4.3)$$

A velocidade de cada partícula em um instante de tempo é obtida pelo princípio fundamental da dinâmica:

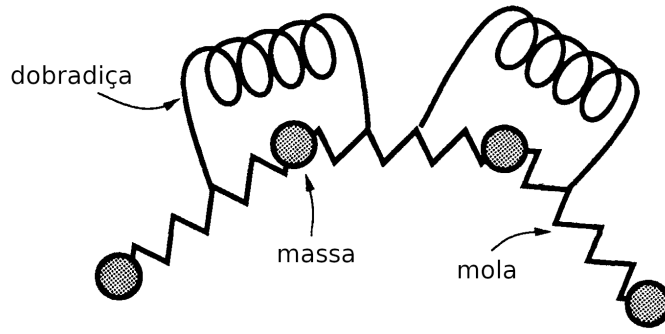


Figura 4.1: Esquema de sistema massa-mola de um fio de cabelo. Traduzido de Rosenblum et al. (1991)

$$a = \frac{1}{m} F_{total} \quad (4.4)$$

$$v_{i+1} = v_i + a\Delta t \quad (4.5)$$

Na Equação 4.5, v_i é a velocidade no instante i e Δt é a diferença de tempo entre os instantes i e $i + 1$.

O algoritmo proposto por Rosenblum et al. (1991) consiste em dividir o intervalo de tempo entre dois *frames* consecutivos da animação em passos menores e, em cada passo, calcular a nova posição das partículas. Para que a simulação traga resultados mais próximos à realidade, os parâmetros m , k_m , k_d , D_m , D_d e D_r devem ser ajustados de maneira coerente.

Por exemplo, através de experimentação, observou-se que deve-se manter uma determinada razão entre a massa das partículas m e as constantes de elasticidade k_m e k_d para evitar que a variação do comprimento dos fios de cabelo (decorrente do comportamento natural de contração e extensão das molas) seja perceptível.

4.2 IMPLEMENTAÇÃO SEQUENCIAL

A implementação desse algoritmo foi feita usando programa Unity, uma *game engine* (um programa para o desenvolvimento de jogos eletrônicos) que traz consigo todo o ferramental necessário para realizar simulações gráficas em tempo real.

O artigo usado como base para essa implementação (Rosenblum et al., 1991) descreve um método para gerar uma distribuição aleatória adequada de fios de cabelo sobre uma superfície. Contudo, para simplificar essa etapa, na presente implementação, os fios de cabelo são gerados sobre os vértices da malha de um objeto tridimensional. Essa maneira não se assemelha ao modo que os fios de cabelo estão distribuídos no couro cabeludo, mas é trivial de se implementar e serve para o propósito deste trabalho, que é avaliar a viabilidade de simulações interativas em tempo real.

O número de passos entre um *frame* e outro deve garantir que Δt seja pequeno o bastante para que a integração numérica da posição das partículas ao longo do tempo esteja dentro de uma margem de erro aceitável. Por observação empírica, concluiu-se que Δt é inversamente proporcional à magnitude das forças que estão sendo computadas.

Foi feita uma modificação da técnica proposta por Rosenblum et al. (1991) a fim de diminuir a quantidade de cálculos feitos entre um *frame* e outro. A modificação consiste em tornar o tamanho do intervalo Δt dinâmico: seu valor é definido após serem calculadas as forças atuantes sobre as partículas. Isso faz com que a duração de cada passo seja diferente. Outra diferença em relação à técnica original foi a não-implementação de mecanismos de detecção de colisão dos fios de cabelo.

O intervalo de tempo entre um *frame* e outro é dado por FPS^{-1} (onde FPS é a taxa de quadros por segundo). Assim como o Δt varia, o número de passos entre os *frames* também não é constante: enquanto a soma do tempo dos passos anteriores for menor que FPS^{-1} , o sistema entende que é possível calcular mais um passo da simulação. Para evitar o cálculo de mais passos do que é possível na janela de tempo para uma aplicação em tempo real, estabeleceu-se um número máximo de passos entre cada *frame*.

Os procedimentos descritos acima estão representados no Algoritmo 1. Nas linhas 4 a 9 se calculam as forças atuantes sobre cada partícula, nas linhas 10 e 11 é computado Δt , e nas linhas 12 a 16, a posição das partículas é atualizada.

Algoritmo 1 Laço principal na atualização das partículas:

Input: n fios de cabelo, cada qual composto por m partículas

```

1:  $T \leftarrow 0$  // tempo decorrido
2:  $C \leftarrow 0$  // contador de passos
3: repeat
4:   for all fio de cabelo  $S_k$  do
5:     for all partícula  $p_i \in F_k$  do
6:       computar forças que atuam sobre  $p_i$ 
7:        $F_i^{(k)} \leftarrow$  força resultante
8:     end for
9:   end for
10:   $F_{max} \leftarrow \max_{\substack{1 \leq k \leq n \\ 1 \leq i \leq m}} F_i^{(k)}$ 
11:   $\Delta t = \frac{FPS^{-1}}{f(F_{max})}$  //  $f$  é uma função arbitrária
12:  for all fio de cabelo  $S_k$  do
13:    for all partícula  $p_i \in F_k$  do
14:      computar nova posição de  $p_i$  em razão de  $F_i^{(k)}$  e  $\Delta t$ 
15:    end for
16:  end for
17:   $T \leftarrow T + \Delta t$ 
18:   $C \leftarrow C + 1$ 
19: until  $T > FPS^{-1}$  or  $C > maxPassos$ 

```

4.3 IMPLEMENTAÇÃO PARALELA

Como não foram consideradas colisões entre fios de cabelos, a simulação de cada fio se tornou independente das demais. Logo, percebeu-se a possibilidade de simular cada fio em paralelo e, por causa da grande quantidade de fios que são representados, a arquitetura massivamente paralela das GPUs pareceu ser adequada para esse tipo de computação.

Na implementação paralela realizada neste trabalho, cada *thread* é responsável por simular um único fio de cabelo por um número pré-estabelecido de passos, com Δt fixo. Essa diferença em relação ao algoritmo sequencial se dá por dois motivos:

1. Existe dependência de dados na etapa de encontrar a maior força resultante (linha 10 do Algoritmo 1). Logo, não é trivial realizar essa operação de forma paralela;
2. Na arquitetura atual das GPUs, muitas vezes é menos custoso realizar operações redundantes do que utilizar mecanismos de sincronização (Nasre et al., 2013).

Para minimizar a quantidade de dados transferidos entre a memória principal e a GPU, desenvolveu-se a seguinte estratégia: na etapa de inicialização do simulador, todas as partículas

que compõem os fios de cabelo são copiados para a memória da GPU. Durante a simulação, apenas as partículas que estão ligadas ao couro cabeludo são novamente copiadas para a GPU, pois suas posições são consequência da interação do usuário com o objeto que constitui o couro cabeludo. Após a computação das novas posições (depois do número de passos fixado), é inevitável realizar uma cópia da posição de todas as partículas para a memória principal a fim de mostrá-las ao usuário. Mais detalhes sobre a implementação paralela usando *Unity* podem ser encontrados no Apêndice B.

4.4 CONCLUSÃO

Como pôde ser observado neste capítulo o algoritmo apresentado por Rosenblum et al. (1991) é simples de ser compreendido e implementado. Apesar de não ter sido implementado em sua completude, foram feitas modificações na técnica apresentada no artigo base a fim de se obter maior desempenho.

A primeira modificação consistiu em implementar um método para tornar Δt dinâmico em relação às forças computadas. A segunda modificação é a adaptação do algoritmo para fazer uso dos recursos de GPUs.

Quão eficazes foram essas modificações em possibilitar a realização de simulações interativas em tempo real será avaliado no próximo capítulo.

5 AVALIAÇÃO EXPERIMENTAL

A avaliação experimental do simulador ocorreu em conjunto com sua implementação. De fato, a versão do simulador para GPU surgiu da necessidade de mais poder computacional a fim de alcançar tempos de processamento aceitáveis.

Para realizar os testes e explorar os limites do simulador para a execução em tempo real, foram usados objetos de vários níveis de complexidade para servir de base para a geração de cabelos. Esses objetos variam desde um quadrado bidimensional até um modelo de uma cabeça de macaco (*Suzanne*), como mostrado na Figura 5.1, que conta com mais de 10.000 fios de cabelo e mais de 30.000 partículas.

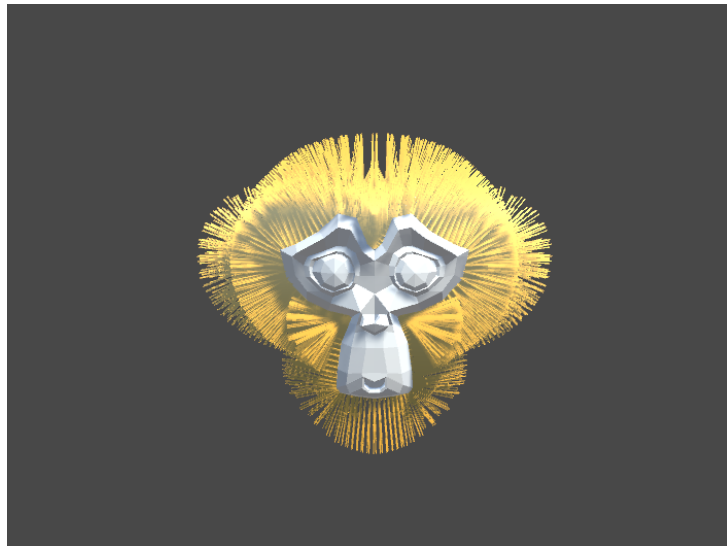


Figura 5.1: Exemplo de cobertura de pelos gerada pelo simulador. Esse modelo conta com 10168 fios e um total de 30504 partículas.

Uma simulação realista depende dos parâmetros m (Equação 4.4), k_m , D_m (Equação 4.1), k_d , D_d (Equação 4.2) e D_r (Equação 4.3) terem valores realistas ou, pelo menos, manterem proporções adequadas entre si. No entanto, não foram encontradas indicações na literatura sobre quais deveriam ser os valores desses parâmetros. Portanto, foi necessário encontrar empiricamente os valores que trouxessem os resultados mais satisfatórios. Tendo isso em vista, deve-se considerar que os resultados apresentados a seguir tenham sido influenciados por parâmetros inadequados.

5.1 AMBIENTE DE TESTES

A máquina usada para os experimentos conta com um sistema operacional GNU Linux, *kernel* versão 5.15.15, 16GB de memória RAM *single channel*, um processador Intel *Core*

	Interativo (> 8 FPS)	Tempo Real (> 12 FPS)
Sequencial	100.800	64.800
Paralelo (GPU)	1.390.500	880.650

Tabela 5.1: Comparação da carga de trabalho máxima do simulador sequencial e paralelo

i7-9750H com *clock* a 2.60GHz e 12 núcleos lógicos, e uma GPU NVidia *GeForce RTX 2060* com *clock* a 1.2GHz e 6GB de memória VRAM.

O simulador de cabelos foi executado em ambiente de desenvolvimento do Unity 2020.3.21f1, uma vez que esse ambiente permite interação com o programa.

5.2 RESULTADOS

Além de uma avaliação qualitativa da simulação, é preciso estabelecer métricas quantitativas para medir o desempenho dela, principalmente em relação à capacidade de executar interativamente em tempo real.

Para a presente aplicação, considera-se que a simulação é **interativa** se ela é capaz de responder a comandos do usuário a, pelo menos, 8 FPS. A simulação é dita ser em **tempo real** se a taxa de quadros por segundo (FPS) for igual ou maior que 12. Em outras palavras, a simulação será aceita como "interativa" se o programa for capaz de processar o comando do usuário e calcular a próxima posição das partículas em 1/8 de segundo, ou 125ms. Do mesmo modo, a simulação será aceita como "em tempo real" se for capaz de gerar o *frame* seguinte em 1/12 de segundo, ou 83ms. Esses valores foram escolhidos porque, historicamente, a taxa de quadros por segundo em animações normalmente era 12, e, no contexto de *softwares* interativos em tempo real, um FPS abaixo de 8 compromete a usabilidade do programa.

Será chamado de **carga de trabalho** o número de operações realizadas sobre as partículas entre um *frame* e outro, e cujo valor é dado pela expressão abaixo:

$$C = n_p * n_s * s \quad (5.1)$$

Onde C é a carga de trabalho, n_p é o número de partículas por fio de cabelo, n_s é o número de fios e s é a quantidade de passos que foram realizados entre um *frame* e outro. Essa fórmula foi obtida através de uma análise experimental e mostra que é possível variar os parâmetros de modo que a carga de trabalho se mantenha constante. Por desses experimentos, foi possível calcular e comparar a carga de trabalho máxima do simulador sequencial e paralelo. A Tabela 5.1 mostra os valores obtidos.

É possível ver que a versão paralela do simulador – com as configurações de número de *threads* e blocos adequadas – chega a comportar uma carga de trabalho 13 vezes maior que a versão sequencial no ambiente de testes já descrito.

Dentro de algumas limitações que serão discutidas a seguir, foi possível simular cabelos lisos em tempo real usando a versão paralela do programa. A Figura 5.2 mostra momentos de

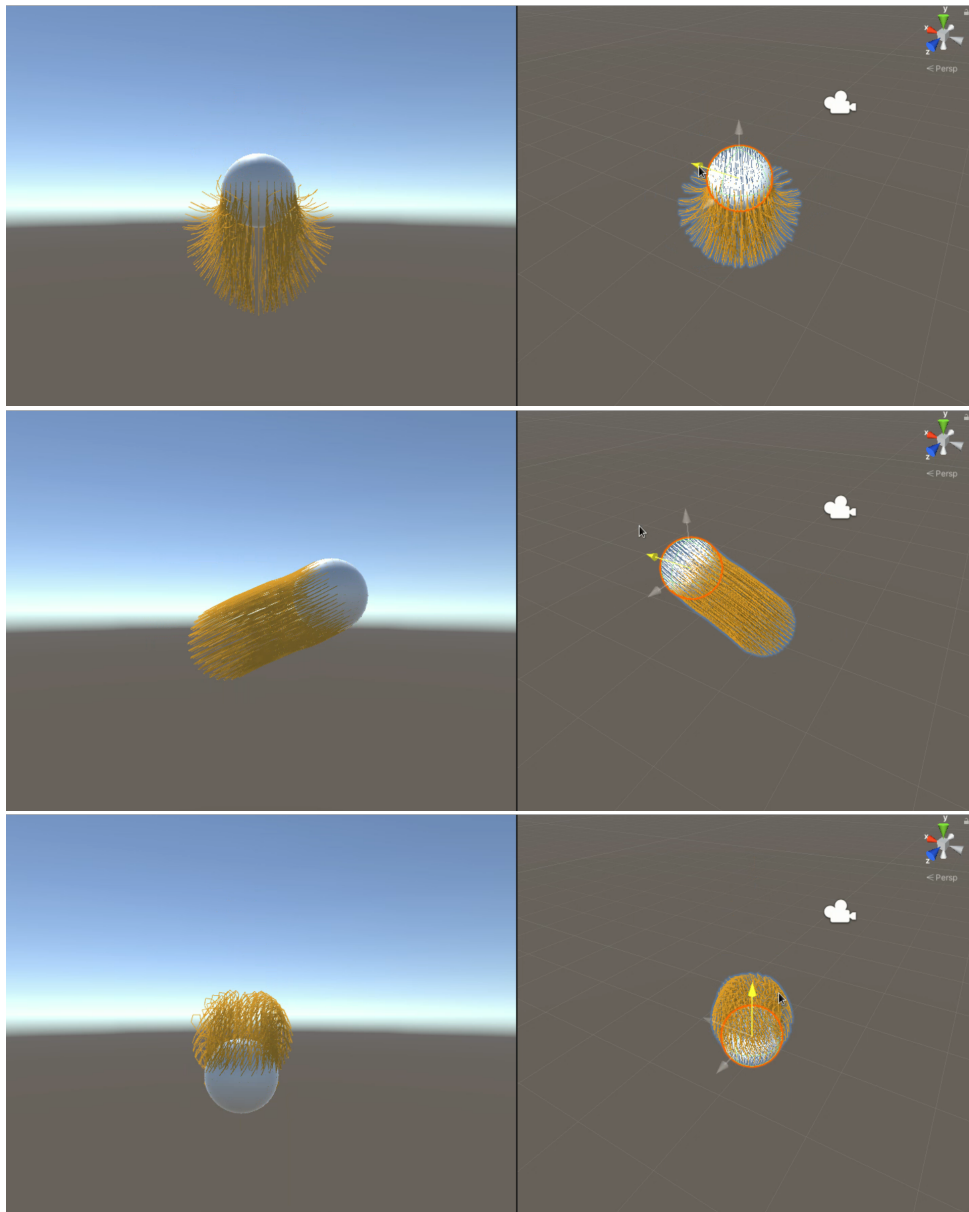


Figura 5.2: Simulação interativa em tempo real. Nestas imagens estão sendo simulados 515 fios de cabelo, e um total de 5150 partículas a uma taxa de 12 FPS.

simulação interativa em tempo real aproximadamente 12 FPS. O número de fios e de partículas do modelo simulado é, respectivamente, 515 e 5150.

5.3 DESAFIOS E LIMITAÇÕES

A primeira e maior limitação está nos fundamentos matemáticos da simulação. Em uma rápida análise, percebe-se que os cálculos realizados constituem uma equação diferencial ordinária (EDO) de segunda ordem que descreve a posição de uma partícula ao longo do tempo. A simulação é, na verdade, o cálculo dos valores de diversas equações desse tipo (uma para cada partícula) para um instante de tempo futuro.

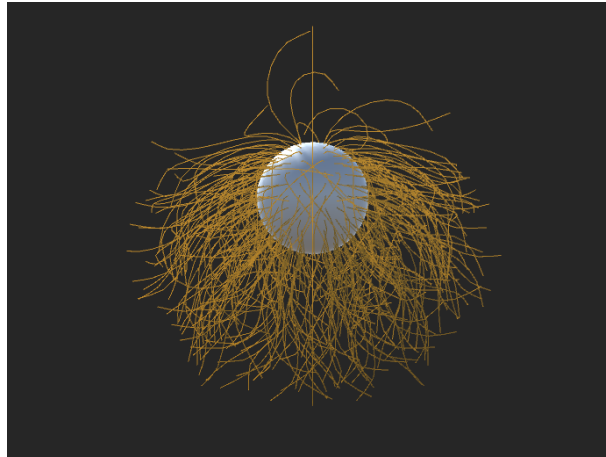


Figura 5.3: Exemplo de erro de convergência

O modo como o programa deste trabalho resolve essas equações é classificado como um método de integração **explícito**, isto é, todos os valores necessários para o cálculo da posição da partícula no momento $t + \Delta t$ já são conhecidos e a única incógnita é o próprio termo de se deseja descobrir.

Sobre métodos explícitos de integração, Marschner e Shirley (2018) comenta que eles, no geral, são facilmente computados, mas, também, a sua grande maioria se caracteriza como um sistema de equações rígido. Isso significa que os erros numéricos podem ter crescimento exponencial para valores de Δt que não tendem a zero.

Isso, de fato, foi observado durante os experimentos. O número de passos precisava ser grande o bastante para que os fios de cabelo convergissem para um estado de repouso. Além disso, percebeu-se que, quanto maior a magnitude das forças resultantes que atuam nas partículas, menor deve ser o intervalo Δt , sob o risco dos erros numéricos tornarem a simulação inutilizável.

Em termos práticos, para que os erros numéricos da simulação estejam dentro de uma margem aceitável (e consequentemente tragam um resultado realista) e a carga de trabalho esteja abaixo dos valores mostrados na Tabela 5.1, o número de fios e partículas deve ser reduzido. Após fixados esses valores, existe um limite de quão intensas podem ser as forças sobre as partículas antes que a simulação seja comprometida pelos erros. Em outras palavras, não se deve submeter os cabelos a movimentos bruscos durante a simulação interativa.

Outro desafio deste trabalho foi encontrar um valor adequado para k_d . De fato, durante a realização dos teste não se encontrou um valor que resultasse em movimentos realistas: ou a influência desse parâmetro sobre a rigidez do fio era insignificante, ou era demasiado a tal ponto que impedia que o fio entrasse em um estado de repouso, como mostrado na Figura 5.3.

5.4 MELHORIAS FUTURAS

A principal melhoria a se fazer no simulador é trocar o método de integração para outro mais estável, tornando as simulações mais eficientes e robustas. Nesse sentido, em Liu

et al. (2013) foi descrito um método que parece ser adequado com a proposta deste trabalho (simulações interativas de cabelos em tempo real). Tal método se baseia na integração implícita de Euler e trata a simulação de sistemas massa-mola como um problema de otimização no qual se busca minimizar a energia potencial do mesmo.

Nesse método, são geradas matrizes que representam as configurações das partículas e das molas entre elas. Tais matrizes se mantêm constantes ao longo da simulação desde que a massa das partículas, a conectividade delas e a rigidez das molas se mantenham constantes. A minimização da energia potencial do sistema é feita através de uma descida de gradiente em blocos de coordenadas, que, apesar de convergir mais devagar que o método de Newton, é computacionalmente mais barato.

O artigo em questão (Liu et al., 2013) também explica como levar em conta colisões (incluindo auto-colisão) e demonstra experimentalmente a capacidade do método simular a dinâmica de tecidos em tempo real usando apenas uma CPU. Além disso, é comentada a possibilidade de aplicar esse método na simulação de cabelos.

5.5 CONCLUSÃO

Neste capítulo foram estabelecidas métricas que foram usadas para avaliar a capacidade das versões sequencial e paralela do simulador de cabelos. Também mostrou-se que é possível realizar simulações interativas em tempo real se baseando no método de Rosenblum et al. (1991), apesar das muitas limitações que esse método apresenta.

Tais limitações foram exploradas experimentalmente e estão expostas com detalhes neste capítulo. Além disso, se propôs uma melhoria futura para remediar o problema de erro numérico decorrente da integração explícita.

6 CONCLUSÃO

No presente trabalho foi explicado o que é simulação e representação de pelos e cabelos em computação gráfica, qual é sua importância nas mídias visuais, e quais são alguns dos desafios dessa área. Também foram apresentadas as etapas do processo de representação gráfica de cabelos e pelos, e foram levantadas questões que devem ser consideradas no momento de se escolher ou desenvolver uma para realizar tal representação.

Este trabalho também mostra algumas das técnicas e soluções que foram desenvolvidas ao longo dos anos para enfrentar os desafios de reproduzir imagens e animações de pelos e cabelos realistas. Foi possível observar que existem diversas abordagens para resolver o mesmo desafio.

O objetivo principal deste trabalho foi implementar um simulador de cabelos baseado no método proposto por Rosenblum et al. (1991) e avaliar se esse método é adequado para simulações interativas em tempo real. Houve algumas modificações do algoritmo original, sendo a principal delas uma adaptação do método a fim de possibilitar a sua execução paralela em GPUs.

Por meio de testes e experimentos, chegou-se à conclusão de que, de fato, é possível realizar simulações de cabelos e pelos interativas em tempo real usando **hardware** disponível atualmente. Contudo, algumas limitações foram identificadas durante o processo de avaliação do simulador. Tais limitações foram explicadas e comentadas. Por fim, se propôs um meio para contorná-las, a fim de tornar o simulador mais robusto.

REFERÊNCIAS

- Bertails, F., Audoly, B., Cani, M.-P., Querleux, B., Leroy, F. e Lévêque, J.-L. (2006). Super-helices for predicting the dynamics of natural hair. *ACM Transactions on Graphics (TOG)*, 25(3):1180–1187.
- Fei, Y., Maia, H. T., Batty, C., Zheng, C. e Grinspun, E. (2017). A multi-scale model for simulating liquid-hair interactions. *ACM Transactions on Graphics (TOG)*, 36(4):1–17.
- Guang, Y. e Huang, Z. (2002). A method of human short hair modeling and real time animation. *10th Pacific Conference on Computer Graphics and Applications, 2002. Proceedings*.
- Iben, H., Brooks, J. e Bolwyn, C. (2019). Holding the shape in hair simulation. Em *ACM SIGGRAPH 2019 Talks*, páginas 1–2, Los Angeles California. ACM.
- Liang, W. e Huang, Z. (2003). An enhanced framework for real-time hair animation. *11th Pacific Conference on Computer Graphics and Applications, 2003. Proceedings*.
- Liu, T., Bargteil, A. W., O’Brien, J. F. e Kavan, L. (2013). Fast simulation of mass-spring systems. *ACM Transactions on Graphics*, 32(6):1–7.
- Marschner, S. e Shirley, P. (2018). *Fundamentals of computer graphics*, capítulo: 16, páginas 426–428. CRC Press.
- Nasre, R., Burtscher, M. e Pingali, K. (2013). Data-driven versus topology-driven irregular computations on gpus. Em *2013 IEEE 27th International Symposium on Parallel and Distributed Processing*, páginas 463–474. IEEE.
- Rosenblum, R. E., Carlson, W. E. e Tripp, E. (1991). Simulating the structure and dynamics of human hair: Modelling, rendering and animation. *The Journal of Visualization and Computer Animation*, 2(4):141–148.
- Ward, K., Bertails, F., Kim, T.-y., Marschner, S. R., Cani, M.-p. e Lin, M. C. (2007). A Survey on Hair Modeling: Styling, Simulation, and Rendering. *IEEE Transactions on Visualization and Computer Graphics*, 13(2):213–234.
- Yuksel, C., Schaefer, S. e Keyser, J. (2009). Hair meshes. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2009)*, 28(5):166:1–166:7.

APÊNDICE A – LISTA DE ARTIGOS SOBRE RENDERIZAÇÃO DE CABELOS E PELOS

Aqui se encontram listados alguns trabalhos que tratam do tema de renderização de cabelos e pelos:

- "Simulating the structure and dynamics of human hair: modelling, rendering and animation.", Rosenblum, R. E., Carlson, W. E., & Tripp III, E. (1991).
- "Light scattering from human hair fibers.", Marschner, S. R., Jensen, H. W., Cammarano, M., Worley, S., & Hanrahan, P. (2003).
- "Volumetric methods for simulation and rendering of hair.", Petrovic, L., Henne, M., & Anderson, J. (2005).
- "A Survey on Hair Modeling: Styling, Simulation, and Rendering.", Ward, K., Bertails, F., Kim, T. Y., Marschner, S. R., Cani, M. P., & Lin, M. C. (2007)
- "500 million and counting: hair rendering on Ratatouille.", Ryu, D. (2007)
- "Physically-accurate fur reflectance: Modeling, measurement and rendering.", Yan, L. Q., Tseng, C. W., Jensen, H. W., & Ramamoorthi, R. (2015).
- "A data-driven light scattering model for hair.", Pekelis, L., Hery, C., Villemin, R., & Ling, J. (2015).

APÊNDICE B – IMPLEMENTAÇÃO PARALELA EM *UNITY*

A *game engine Unity* permite que o comportamento, os estados e os atributos de um objeto de uma cena sejam programados usando uma API ¹ própria na linguagem de programação C#. O código escrito em C# é executado na CPU de modo sequencial (é possível escrever código paralelo, mas esse detalhe é irrelevante no presente trabalho).

Também é possível realizar chamadas para executar código na GPU, semelhante ao que é feito com as linguagens C/C++ e CUDA. No entanto, ao invés de CUDA, é usada a linguagem HLSL, e o que seria chamado de *kernel* é chamado de *compute shader*.

Para a implementação paralela do simulador, todas as partículas de cada fio de cabelo são concatenadas em um único vetor e são copiadas para a memória da GPU no início da simulação. Como mencionado no Capítulo 4, a posição das partículas no couro cabeludo é copiada novamente para a GPU a cada *frame*. Dentro do *compute shader*, cada *thread* está programada para manipular apenas o intervalo de índices do vetor de partículas que correspondem ao fio de cabelo que lhe corresponde.

Os cálculos realizados, assim como o modo como a posição das partículas é atualizada e devolvida para a memória principal já foi descrito no Capítulo 4. Portanto, não é necessário explicá-lo novamente.

¹*Application Programming Interface*