

Simulação de tecidos utilizando malha irregular com CUDA

Nicholas Steppan Meschke, Anita Fernandes, Eros Comunello

PROINOVA – Universidade do Vale do Itajaí - UNIVALI

nichsm@edu.univali.br, {anita.fernandes, eros.com}@univali.br

***Abstract.** This paper presents the development of an interactive and real-time cloth simulation using an irregular triangle mesh on the graphics card. The proposed approach will implement two cloth simulation algorithms on processor and graphics card and will evaluate the performance improvement. Processor prototype implementation is finished. Next steps are (i) the implementation of those two cloth simulation algorithms on the graphics card and the (ii) performance improvement evaluation.*

1. Introdução

Em uma aplicação na qual existem personagens virtuais há a necessidade de criar uma simulação de tecidos interativa e em tempo real. Este problema tem sido abordado por pesquisadores da computação gráfica há quase duas décadas [Provot 1995; Baraff e Witkin 1998; Volino e Magnenat-Thalmann 2000; Müller et al. 2007].

A busca de realismo em simulação de tecidos na computação gráfica é uma tarefa de elevado custo computacional. Os algoritmos de simulação de tecidos possuem, em geral, grande quantidade de operações sequenciais executadas na CPU (*Central Processing Unit*). Este trabalho visa à implementação de dois destes algoritmos em GPU (*Graphics Processing Unit*) utilizando uma malha de triângulos irregular.

2. Abordagem Proposta

A simulação de tecidos é feita através de uma malha de triângulos na qual cada vértice desta malha é chamado de partícula e cada aresta é chamada de mola, cada partícula possui uma posição, velocidade, força e massa, cada mola possui duas partículas, uma distancia de descanso, uma constante de amortecimento e uma constante de rigidez, além de operações no conjunto de partículas e molas, executadas a cada quadro do programa.

A Figura 1 apresenta o encadeamento dessa operações, as quais consistem na inicialização, onde são atribuídas as partículas suas posições correspondentes a malha de triângulos e calculadas as distancias de descanso das molas, no cálculo de força, onde é calculada a quantidade de força necessária para que as duas partículas de uma mola voltem a sua posição de descanso, no calculo de velocidade, onde é calculada a velocidade com base na gravidade, no tempo e nas forças da partícula e na integração, onde as forças e a velocidade da partícula combinadas ao tempo são utilizadas para calcular a nova posição da partícula, simulando o comportamento de um tecido. A cada quadro da simulação, o programa executa estas operações sequencialmente na CPU.

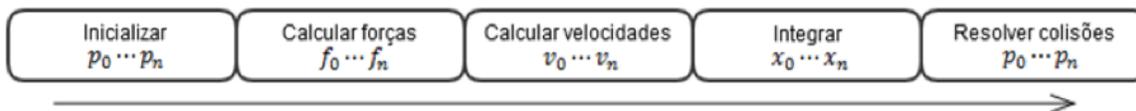


Figura 1. Visão geral da simulação de tecidos sequencial

Contudo, tais operações podem ser implementadas utilizando uma abordagem concorrente empregando GPGPU (*General Purpose Graphics Processing Unit*). A referida abordagem pode proporcionar a simulação em tempo real – considerando um quadro de resolução 800x600 sendo executado a 60 quadros por segundo.

Serão paralelizados os cálculos das forças, os cálculos das velocidades e integração das partículas, conforme mostrado na Figura 2. A plataforma de programação paralela a ser usada será CUDA (Compute Unified Device Architecture)¹, desenvolvida pela NVIDIA. Uma função em CUDA a ser executada na GPU é chamada de kernel. Estes kernels são trechos de código que serão executados em paralelo pelas threads da GPU. Cada thread executada será equivalente a uma operação sobre uma partícula.

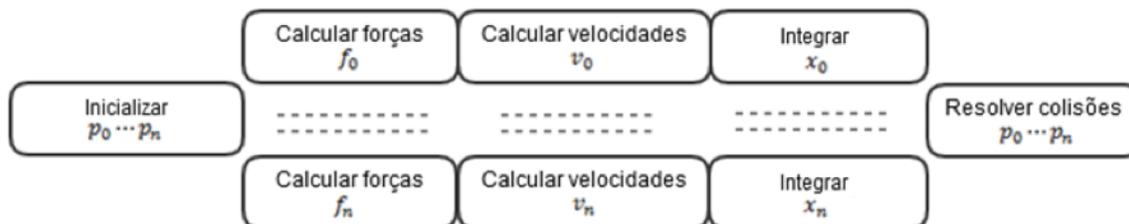


Figura 2. Visão geral da simulação de tecidos executada em paralelo

3. Considerações finais

O projeto encontra-se em fase inicial de pesquisa e desenvolvimento. Foram implementados dois protótipos de simulação de tecidos, conforme descritos por Müller et al. (2007) e Baraff e Witkin (1998), utilizando um processador Duo core e uma placa de vídeo NVIDIA 8800 GT foram obtidos 29 quadros por segundo no algoritmo de Müller e 20 quadros por segundo no algoritmo de Baraff e Witkin, utilizando uma malha irregular de 22604 triângulos, conforme pode ser visto na Figura 3. O próximo passo é a paralelização destes algoritmos em CUDA. Após a implementação em CUDA, será realizado um experimento para se avaliar a melhora de desempenho entre o algoritmo não paralelizado e o algoritmo paralelizado.

¹ Disponível em <http://www.nvidia.com.br/object/cuda_home_new_br.html>

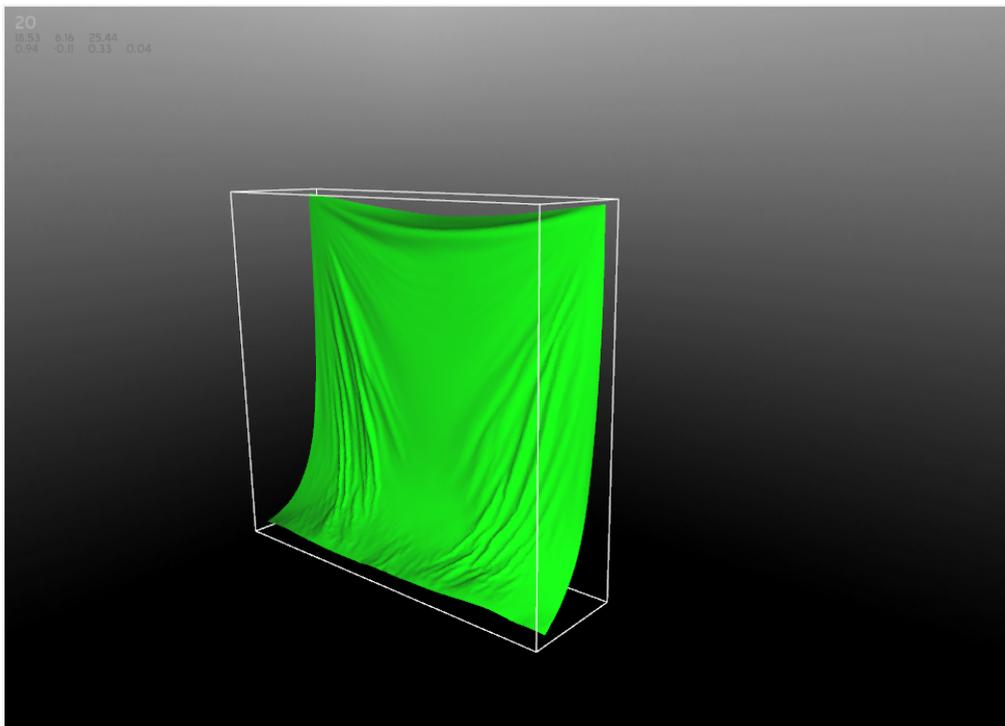


Figura 3. Bandeira com algoritmo de Baraff e Witkin

Referencias

- Baraff, D. e Witkin, A. (1998). Large steps in cloth simulation. In Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '98, pages 43–54, New York, NY, USA. ACM.
- Muller, M., Heidelberger, B., Hennix, M., e Ratcliff, J. (2007). Position based dynamics. *Journal of Visual Communication and Image Representation*, 18(2):109 – 118.
- Provot, X. (1995). Deformation constraints in a mass-spring model to describe rigid cloth behavior. In *IN GRAPHICS INTERFACE*, pages 147–154.
- Volino, P. e Thalmann, N. (2000). Implementing fast cloth simulation with collision response. In *Computer Graphics International, 2000. Proceedings*, pages 257–266.