

Explorando a Terceira Lei de Newton para Simulação N-Corpos em uma Plataforma GPU

Vinicius A. Herbstrith, Lucas Mello Schnorr
{vaherbstrith,schnorr}@inf.ufrgs.br



Contexto

- Simulações N-corpos são simulações de um sistema de partículas sobre a influência das forças físicas externas e forças geradas pelas outras partículas do mesmo sistema
- Simulações deste tipo são largamente aplicadas em astrofísica, física de plasma e dinâmica molecular
- A simulação ocorre em intervalos de tempo onde, para cada intervalo e para cada partícula, é calculada a força resultante que uma partícula sofre devido a interação com as outras
- Complexidade $O(N^2)$, onde N é a quantidade de partículas presentes no sistema.
- Bom desempenho no ambiente massivamente paralelo da GPU.
- Algoritmo proposto por Halpem (CppCon2014)
 - Terceira lei de Newton: Para toda interação, na forma de força, que um corpo A aplica sobre um corpo B, dele A irá receber uma força de mesma direção, intensidade e sentido oposto
 - Consegue reduzir o número de iterações pela metade
 - Algoritmo recursivo (evita condições de corrida)
 - Algoritmo criado com a arquitetura CPU em mente

Implementação

- Decomposição da execução recursiva
- Processar de forma ascendente
 - Começar a partir das folhas da árvore recursiva
 - Distribuição das folhas entre os blocos de thread da GPU
 - * Gera condições de corrida
- Tratando a Condição de corrida:
 - Evitar que blocos de thread processem uma mesma partícula
 - Modelo mestre-trabalhador
 - * Mestre controla o avanço dos blocos entre as folhas
 - * Mutex para a área de código que define o papel de mestre
 - Blocos esperam os outros blocos terminarem o processamento

Experimentos

- Análise de desempenho:
 - 1000 partículas
 - 10 iterações
 - 10 execuções
- Comparação com a implementação n-corpos disponibilizada em GPU Gems
- Utilizamos gráficos com a média e o desvio padrão das 10 execuções
- Plataforma experimental:
 - Processador Intel i7-4770 CPU (3.40GHz)
 - GPU NVIDIA GTX 760 (1152 cuda cores, 980Mhz)
 - Disco rígido HDD 1TB SATA II
 - Memória de 8GiB DIMM DDR3(1.6GHz)

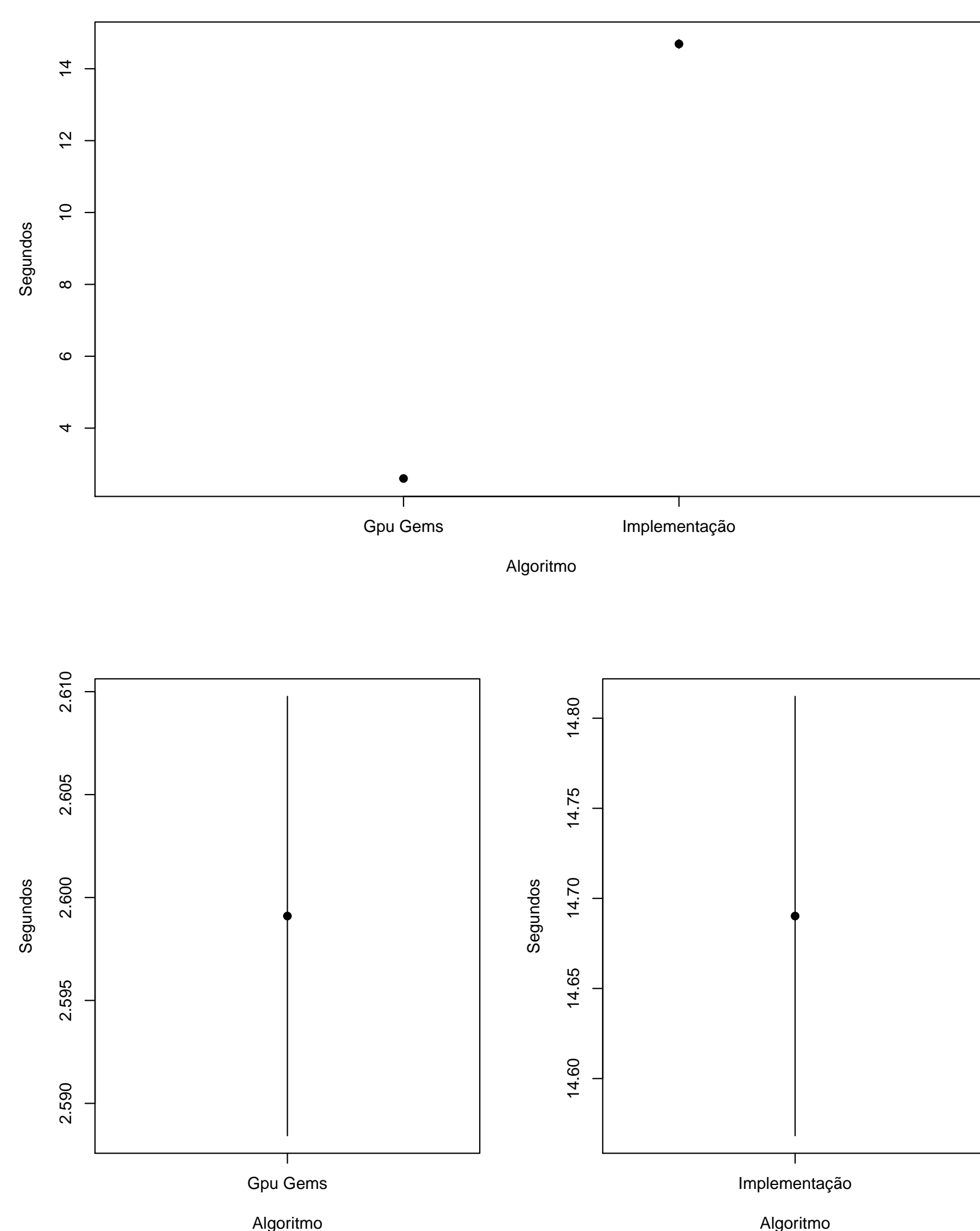


Figura 2: Tempo de execução dos experimentos

Objetivos

- Implementação do algoritmo de Halpem em um ambiente GPGPU:
 - Encontrar uma alternativa para a recursão
 - * Recursão apresenta uma performance pobre na GPU
 - Superar condições de corrida
 - Definição de um novo algoritmo que se aproveite da terceira lei de Newton

Simulação

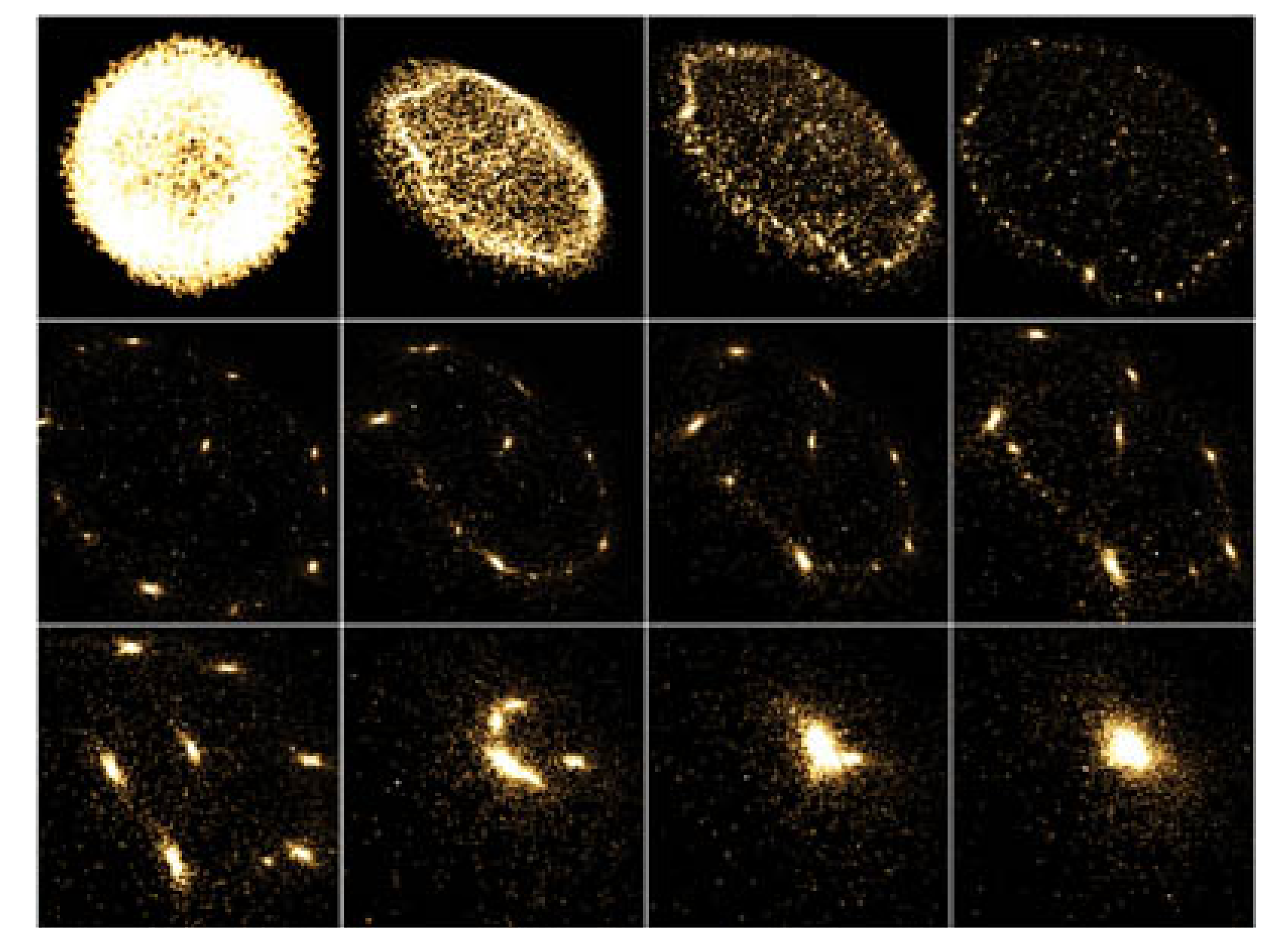


Figura 1: Imagem resultante da simulação executada pelo algoritmo disponibilizada em GPU Gems 3

Conclusões

- Resultado até 7 vezes inferior à uma implementação força bruta
- Razões
 - Necessidade de Mutex
 - Modelo mestre-trabalhador
 - Não se aproveitar da localidade de memória
- Trabalhos futuros:
 - Implementação em um ambiente em cluster

Referências

- [1] Sverre J. Aarseth. *Gravitational N-Body Simulations*. Cambridge University Press, 2003. Cambridge Books Online.
- [2] J. Barnes and P. Hut. A hierarchical $O(N \log N)$ force-calculation algorithm. , 324:446–449, December 1986.
- [3] Tomoaki Ishiyama, Keigo Nitadori, and Junichiro Makino. 4.45 Pflops Astrophysical N-Body Simulation on K computer – The Gravitational Trillion-Body Problem. 2012.
- [4] Lars Nyland and Mark Harris. Fast N-Body Simulation with CUDA, 2007. GPU Gems 3.
- [5] Pablo Halpern. Decomposing a problem for parallel execution. CppCon2014 talk, 2014.