

Sistemas Operacionais – 2021/2 – UnB
Felipe Oliveira Magno Neves – 180016296
Trabalho 2 - Boids



Descrição do trabalho:

Para este trabalho, decidi por propor um problema a ser abordado, em vez de seguir a orientação proposta. O problema escolhido foi a simulação de Boids, um problema de paralelismo em uma simulação de corpos com muitas interações. Segue um link para a página da universidade Stanford com uma descrição básica do problema:

<https://cs.stanford.edu/people/eroberts/courses/soco/projects/2008-09/modeling-natural-systems/boids.html#:~:text=Boids%20is%20an%20artificial%20life,behavior%20of%20each%20individual%20bird>.

Devem ser seguidas as regras de separação, alinhamento e coesão. Além disso, escolhi incluir uma regra extra que impede que os objetos se aproximem muito dos cantos do espaço de simulação.

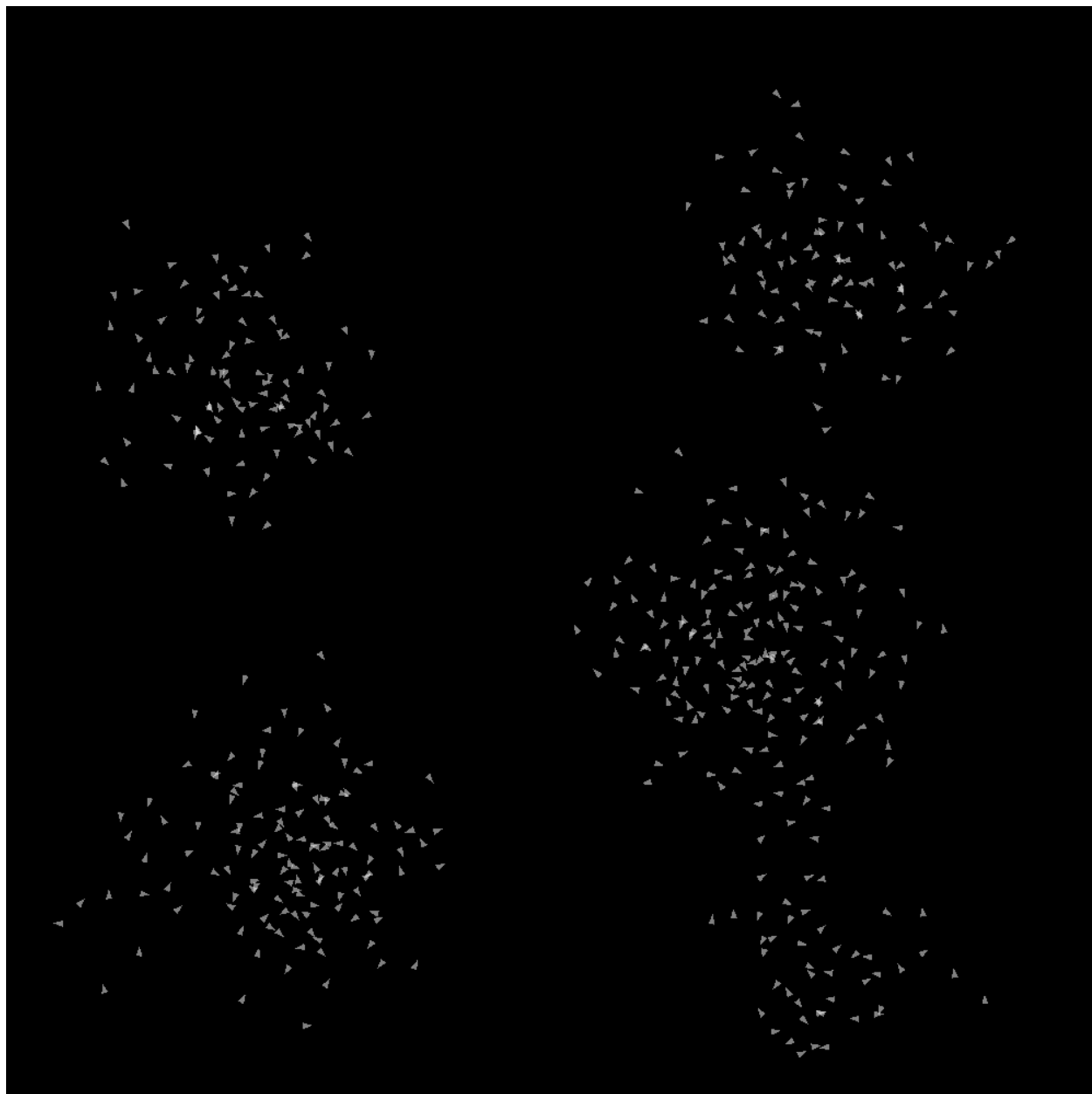
Após feita a implementação básica, foi escolhido uma forma de otimizar o problema para a divisão em threads de forma que a qualidade da simulação não fosse penalizada. Para tal, o espaço foi dividido em células quadradas de lado igual à distância de visão de um boid, de forma tal que seja necessário que o corpo em uma célula necessite iterar apenas sobre os outros em sua célula e células vizinhas – o que já é uma grande economia em relação à implementação simples, na qual cada corpo deve iterar sobre todos os outros. Esta é uma ideia parecida com a de divisão do espaço em quadrees,

Então, escolhe-se um número de threads a ser usado, e cada uma fica responsável pelo processamento de um número (quase) igual de células. Para tal, o processamento é dividido em 3 etapas: etapa de cálculo, na qual cada boid calcula as estatísticas necessárias sobre seus vizinhos. Etapa de atualização dos boids, na qual a posição e velocidade de cada boid é atualizada com base nas estatísticas calculadas. Etapa de atualização dos quadrantes, na qual cada quadrante verifica se um dos boids sob sua responsabilidade passou para um quadrante vizinho, e transfere a responsabilidade caso isso ocorra. Essas etapas são executadas uma por vez (aguarda-se que os threads finalizem seu processamento antes de passar para a próxima etapa).

Foi elaborado, também, um modo de benchmark simples no qual são simuladas 1000 iterações de 500 boids para cada etapa de implementação do algoritmo (implementação básica, divisão espacial e divisão em threads). Este modo permite, também, que seja especificado quantas threads devem ser usadas.

Resultados:

Imagem da simulação gráfica em tempo real:

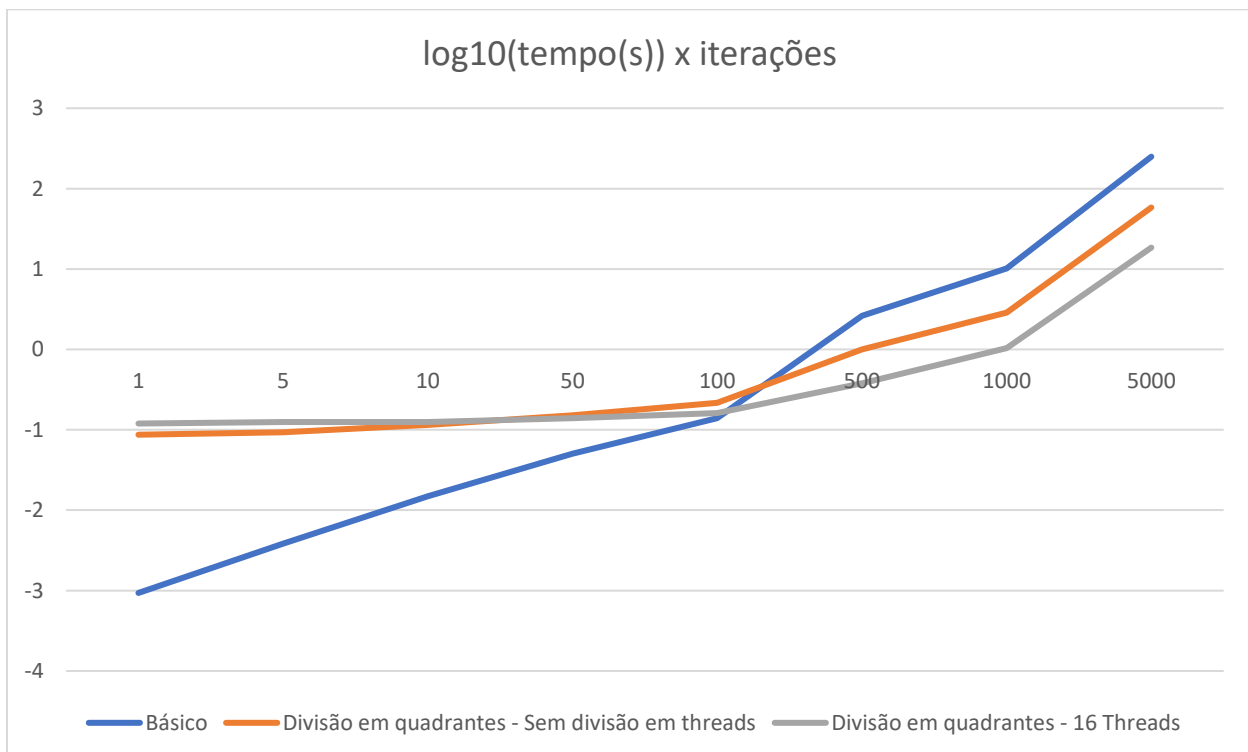


Resultados de benchmarking:

Os resultados abaixo foram gerados em um notebook com o sistema operacional Linux Ubuntu 18.04.3 LTS, com processador Intel Core i7 de 8ª geração (Dell inspiron 15 modelo P61F).

Tipo de Simulação	Tempo para 1000 iterações com 500 Boids(s)
Básica	2,6203
Divisão em quadrantes – Sem uso de thread	0,9993
Divisão em quadrantes – 2 Threads	0,7353
Divisão em quadrantes – 4 Threads	0,5315
Divisão em quadrantes – 8 Threads	0,4030
Divisão em quadrantes – 16 Threads	0,3794
Divisão em quadrantes – 32 Threads	0,4539

Tipo de Simulação	Tempo para 1000 iterações com 5000 Boids(s)
Básica	249,6766
Divisão em quadrantes – Sem uso de thread	58,25165
Divisão em quadrantes – 16 Threads	18,47199



Os resultados obtidos são considerados ótimos pois mostram considerável melhoria ao ser aplicada a divisão em threads proposta para simulações com mais de 100 boids. Vale a

observação de que diversas melhorias ainda são possíveis na abordagem, sendo a principal delas o fato de cada uma das três etapas de uma iteração precisar esperar que os threads – e consequentemente setores – finalizem sua execução.