

RELATÓRIO DO PROJETO FINAL

Murilo Magiolo Geraldini, RA: 201151162

Ramon Varela Gonzalez, RA: 201150212

Problema A

O problema A é o problema de classificação de um ponto P dado um conjunto de pontos S divididos em n grupos. A paralelização do código sequencial fornecido durante o desafio de programação não é trivialmente paralelizável, pois um dos laços internos introduz uma condição de corrida com threads tentando ler e escrever no vetor *distances* que é compartilhado entre todas as threads. A solução mais imediata, que é de definir aquele pedaço do código como uma região crítica utilizando *#pragma omp critical*, não apresenta resultados satisfatórios. Isso ocorre porque a região crítica reduziria significativamente a quantidade de código que pode ser executado em paralelo e também aumentaria o tempo de execução devido ao overhead associado à criação de uma região crítica no código.

A solução encontrada foi a reimplementação do código onde cada thread possui um vetor de distâncias privado, dessa forma eliminando a condição de corrida existente. Isso, porém, adiciona passos de computação que antes não existiam. Após o cálculo das distâncias de cada thread, os valores encontrados precisam ser armazenados em um vetor final de distâncias que seu tamanho é a soma dos tamanhos dos vetores de distância de cada thread. Diferente do vetor de distância do programa original, esse vetor de distância final não está necessariamente ordenado da menor distância para a maior e por isso é necessário fazer executar a ordenação desse vetor enquanto mantém o vetor de grupos sincronizado, já que *labels[i]* corresponde ao grupo de *distances[i]*.

Além disso, é necessário verificar se todos os elementos dos vetores são válidos uma vez que os vetores locais são iniciados com -1 e, diferente do código original, não há garantia de que todos os elementos do vetor serão sobrescritos com distâncias válidas. Após isso, a execução do código segue igual ao da versão sequencial.

Mesmo com todas essas adições o código provou ter um desempenho melhor. A evidência de maior desempenho vem da execução feita no supercomputador Santos Dumont que foi disponibilizado no desafio de programação com 48 threads e considerando um conjunto de testes grande. A execução em um computador pessoal com 8 threads e utilizando o conjunto de testes disponibilizado pelo desafio de programação não mostrou resultados tão satisfatórios.

Problema B

O problema B é o problema de suavização de uma imagem. Esse problema é trivialmente paralelizável e também se beneficia muito de vetorização. Por conta disso, há possivelmente ganho de performance em utilizar *#pragma omp simd*, AVX intrinsics ou até mesmo GPU. Porém, devido ao pouco tempo restante no desafio de programação, apenas a primeira opção de vetorização simples com *#pragma omp for* foi testada, com resultados satisfatórios quando executado no supercomputador Santos Dumont.