

## Identificação

Osvaldo Henrique Becker - 175963@upf.br

## Implementação

Algumas modificações incluem a criação de uma nova struct para armazenar variáveis, além da criação de um tipo MPI\_DATATYPE específico para essa struct, para permitir a sua transferência via mensagens entre processos. Nessa implementação, o trabalho é dividido em partes iguais para todos os processos. O processo principal envia as partes para os outros processos, processa a sua parte, recebe a resposta dos outros processos com o resultado e armazena todos os resultados. Além disso, alguns broadcasts de variáveis são feitos para manter a consistência dos dados.

## Recursos de programação paralela utilizados

Na parte de paralelização foram utilizados recursos da MPI, que é uma API que fornece recursos de programação paralela em C, C++ e Fortran.

## Análise dos resultados dos testes

Foram feitas 3 execuções seguidas para cada caso de teste, a fim de produzir uma média do tempo de execução. Como o tempo só variou em milissegundos em todas as execuções para cada caso de teste, para facilitar, a média será arredondada em minutos e segundos. Abaixo a tabela.

	Processos: 1	Processos: 2	Processos: 4
Execução 1	4m12,771s	2m30,570s	1m36,224s
Execução 2	4m12,929s	2m30,700s	1m36,022s
Execução 3	4m12,850s	2m30,787s	1m36,034s
Média	4m12s	2m30s	1m36s

Em relação aos tempos de execução, quando executado com 2 processos o tempo foi 40% menor do que com 1 processo, e quando executado com 4 processos foi 60% menor do que com 1 processo.

## Considerações finais

É possível concluir que a paralelização deste algoritmo utilizando MPI teve resultados positivos, visto que os tempos de execução reduziram significativamente ao ser executado com até 4 processos. Por limitações técnicas, não foi possível realizar o teste com mais de 4 processos ou com vários computadores interligados em uma rede.

## **Anexos:**

### **Ambiente de execução**

O testes foram feitos em um computador com o sistema operacional Linux Mint 20 Cinnamon instalado nativamente, com o Cinnamon na versão 4.6.7 e o kernel 5.4.0-52-generic. Na parte de hardware, possui um processador Intel Core i7-7700HQ com 4 núcleos e memória RAM de 15.5 GiB.

### **Medidas de desempenho**

Com os resultados dos tempos de execução, foi calculado as medidas de desempenho de aceleração, eficiência e custo para cada um dos casos de teste. Abaixo a tabela.

	Aceleração	Eficiência	Custo
Processos: 1	1	1	252
Processos: 2	1.68	0.84	300
Processos: 4	2.62	0.65	384