

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO

Bacharelado em Ciência da Computação

Programação Paralela e Distribuída

Profº: Gabriel Pereira da Silva

# Relatório - Trabalho 1

Eduardo Barbosa - 116150432

Gabriel Villares - 114089936

## Questão 3

### Objetivo

A terceira questão consiste em modificar um programa já paralelo que calcula a equação de uma reta utilizando o método de mínimos quadrados. As modificações consistem em trocar os métodos de comunicação usados no programa para os métodos de comunicação coletiva pedidos, foram usados o MPI\_Bcast e MPI\_Reduce, também foi pedido a troca das rotinas de envio dos valores de x e y para o MPI\_Bsend. Além disso foi feito um programa que gera o arquivo de entrada para o algoritmo dos mínimos quadrados conforme as especificações dadas na questão.

### Metodologia

A primeira modificação feita foi a troca das funções que enviam o valor de n lido do arquivo, no algoritmo original este envio era feito através de várias chamadas às rotinas MPI\_Send e MPI\_Recv, a modificação consistiu em trocar essas rotinas por MPI\_Bcast, o que resultou em uma redução no número de chamada de funções, originalmente o processo 0 chamava MPI\_Send nprocs-1 vezes e os outros nprocs-1 processos chamavam MPI\_Recv uma vez, agora todos os processo, incluindo o 0, chamam MPI\_Bcast uma vez.

A segunda modificação implementada foi trocar as rotinas que enviam x e y para cada um dos processos por MPI\_Bsend, para tanto foram adicionadas 4 variáveis ao código, um void pointer para o buffer, e 3 variáveis para armazenar os tamanhos intermediários e finais do buffer, além disso foi usada 2 chamadas a rotina MPI\_Pack\_Size para calcular o tamanho correto do mesmo. Essa modificação no entanto não alterou o número de chamadas às rotinas de comunicação, que continua sendo  $4 \cdot (\text{numprocs} - 1)$  chamadas a Bsend e 4 chamadas a MPI\_recv.

A última modificação foi a substituição das rotinas de recepção e adição das somas parciais, as rotinas usadas originalmente, MPI\_Send e MPI\_Recv, foram substituídas por 4 chamadas a MPI\_Reduce, em oposição às 4 chamadas a MPI\_Send feitas pelo processo 0 e às  $4 \cdot (\text{numprocs} - 1)$  chamadas a MPI\_Recv feitas pelos demais processos

O programa que gera o arquivo de entrada usa a equação  $y = 1 \cdot x + 5$  para fazer as mutações.

### Conclusão

Após a realização dos testes e geração dos gráficos, foi possível observar que o algoritmo não se mostrou eficiente com muitos processos, havendo casos onde sequer houve ganho de desempenho comparado a execução com um único processo. Além disso foi possível observar um grande retorno negativo no speedup, a partir de 8 processos.