

# Relatório Projeto 2 - MPI

Gustavo Pereira Pazzini

Julia Rodrigues Gubolin

Docente: Aleardo Manacero Junior

Disciplina: Sistemas Distribuídos

Bacharelado em Ciência da Computação UNESP - IBILCE

14 de agosto

# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>3</b>
<b>2</b>	<b>Manual de compilação</b>	<b>3</b>
2.1	MPI . . . . .	3
2.2	Soquete . . . . .	3
<b>3</b>	<b>Resultados</b>	<b>3</b>
3.1	Tabelas de tempo . . . . .	4
3.1.1	Sequencial . . . . .	4
3.1.2	MPI . . . . .	4
3.1.3	Soquete . . . . .	4
3.2	Gráfico de comparação . . . . .	5
<b>4</b>	<b>Conclusão</b>	<b>6</b>

# 1 Introdução

Neste relatório, será descrita a análise de tempo necessário para que o cliente execute as tarefas indicadas pelo servidor utilizando MPI. A tarefa está relacionada com o espalhamento de cores em uma matriz. Este tempo será comparado com a execução das tarefas por uma implementação usando sockets e com uma implementação sequencial.

## 2 Manual de compilação

### 2.1 MPI

O projeto utilizando MPI deve ser compilado e executado em uma máquina Linux, que deve possuir as bibliotecas do MPI instaladas.

#### Instalação das bibliotecas:

```
sudo apt install openmpi-bin  
sudo apt install libopenmpi-dev
```

#### Compilar e executar:

```
mpicc rojeto2_MPI_Julia_Gustavo.c -o Projeto2_MPI_Julia_Gustavo  
mpirun --oversubscribe --use-hwthread-cpus -np 16 rojeto2_MPI_Julia_Gustavo
```

### 2.2 Soquete

O projeto utilizando soquetes pode ser compilado tanto em ambiente Linux quanto em Windows, desde que possuam a versão 11.0.12 do JAVA. Os comandos são:

#### Clientes:

```
// Compilar  
javac TCPCliente.java  
// Executar  
java TCPCliente
```

#### Servidor:

```
// Compilar  
javac TCPServidor.java  
// Executar  
java TCPServidor
```

## 3 Resultados

Nesta seção serão apresentados os resultados das execuções das soluções com MPI, soquetes e sequencialmente. Os tempos indicados nas tabelas estão em **segundos** e são a média de 10 execuções.

### 3.1 Tabelas de tempo

#### 3.1.1 Sequencial

Tabela 1: Tempos aplicação sequencial

Matrizes	tempo
256x256	9,576
512x512	37,832
768x768	91,454

#### 3.1.2 MPI

Tabela 2: Matriz 256x256

Clientes	tempo
2	37.37964745
4	13.1793623
8	12.5814348625
16	15.9342118688

Tabela 3: Matriz 512x512

Clientes	tempo
2	239.672144
4	92.67380175
8	82.629455
16	100.986309562

Tabela 4: Matriz 768x768

Clientes	tempo
2	291,564
4	324.63720775
8	294.6713145
16	328.261068187

#### 3.1.3 Soquete

Tabela 5: Matriz 256x256

Clientes	tempo
2	95,230
4	190,015
8	278,480
16	301,421

Tabela 6: Matriz 512x512

Clientes	tempo
2	212,799
4	414,636
8	619,578
16	973,261

Tabela 7: Matriz 768x768

Clientes	tempo
2	231,184
4	376,449
8	672,853
16	2183,555

### 3.2 Gráfico de comparação

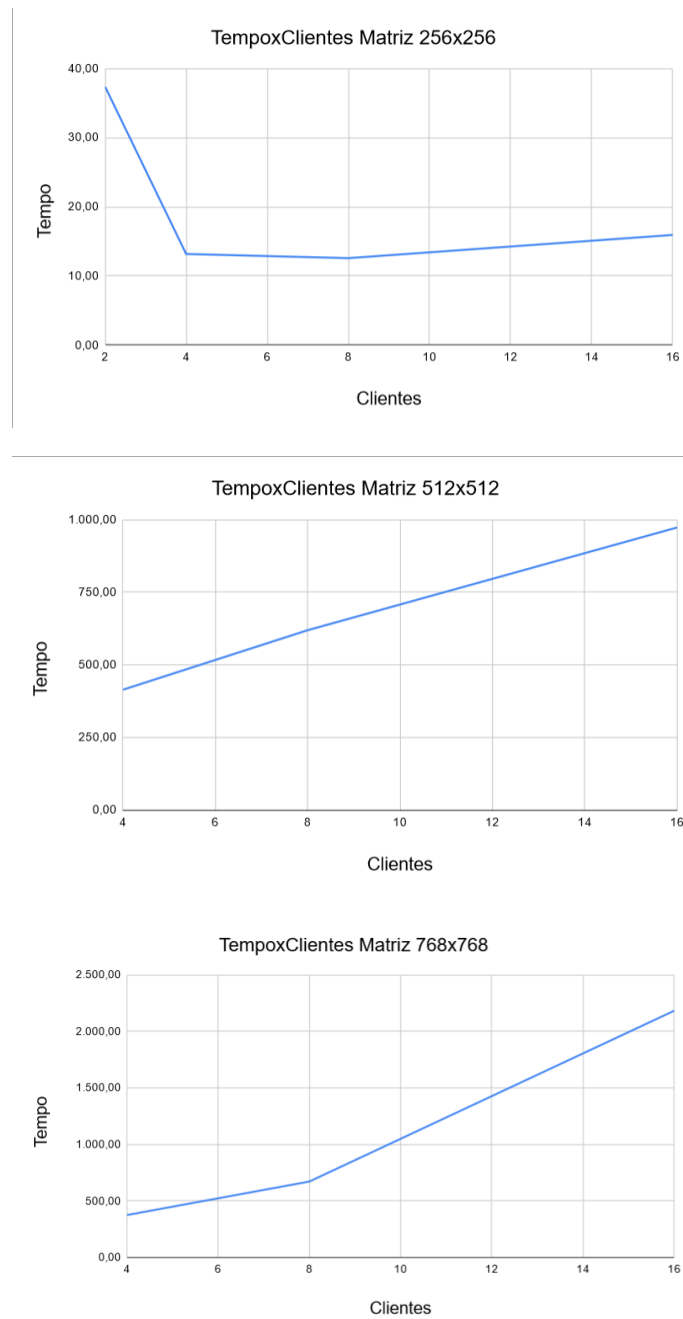


Figura 1: Gráficos MPI.

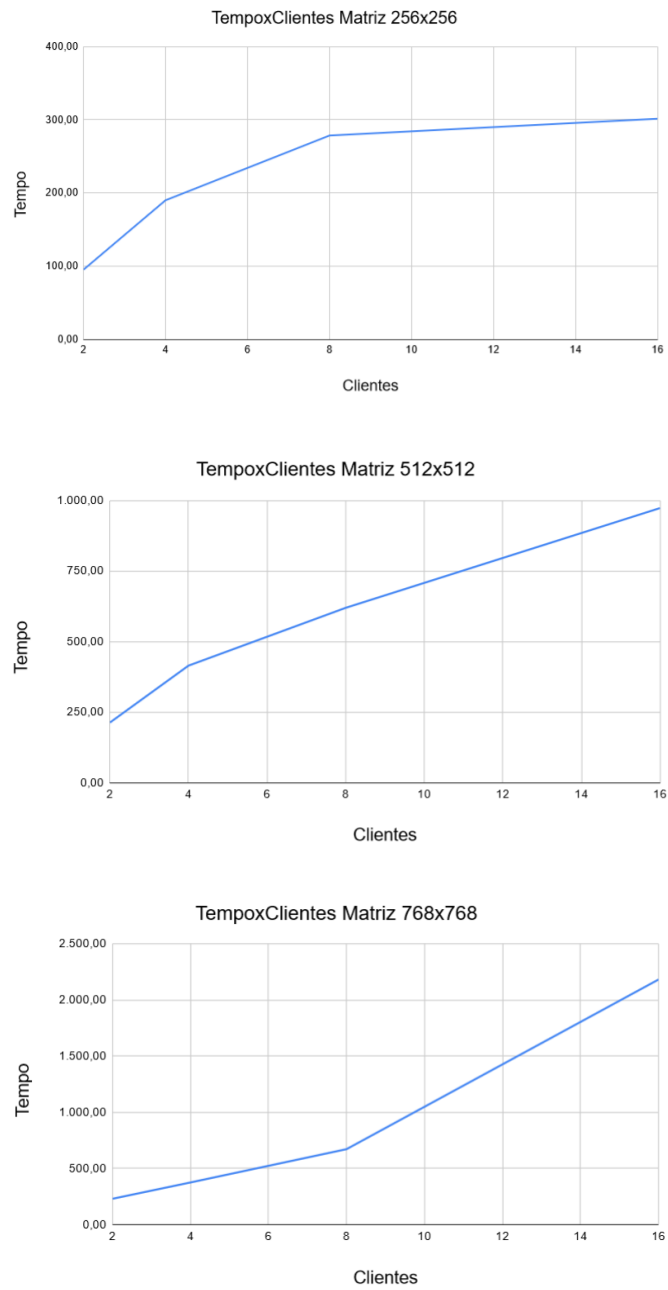


Figura 2: Gráficos Soquetes.

## 4 Conclusão

Através dos gráficos e das tabelas, podemos concluir que a aplicação utilizando MPI foi mais eficiente para os 3 tamanhos de matrizes, independente da quantidade de clientes.