

**Universidade Federal de São Carlos**  
**Programa de Pós-Graduação em Ciência da Computação**

**CCO-129-7 Introdução à Computação de Alto  
Desempenho (2021/2)**

---

**Exercício Programa - EP 4**

**OpenHPC UFSCar**

---

**Alcides Mignoso e Silva      760479**

**Submission Date : 20/03/2022**

# 1 Atividade

O objetivo desta atividade é paralelizar a soma de dois vetores  $C = A + B$  em MPI. O código base utilizado foi o existente em [https://github.com/HPCSys-Lab/HPC-101/tree/main/examples/vec-add/vecadd\\_seq.c](https://github.com/HPCSys-Lab/HPC-101/tree/main/examples/vec-add/vecadd_seq.c) e este relatório visa discutir valores de tempo de execução, speedup e eficiência das execuções.

As execuções foram realizadas utilizando vetores com 1.000.000 elementos, sendo que cada execução do programa realizou a operação 10.000 vezes. Optou-se por realizar a execução mais vezes em detrimento do número de elementos dos vetores por conta de problemas com a memória das máquinas, que estava estourando para vetores com a quantidade de elementos maior do que a utilizada. As execuções foram feitas no cluster HPC da UFSCar, utilizando 1, 2, 3, 5, 10, 15, 20, 30, 50, 90 nós (máquinas), cada máquina rodando o problema em uma única thread.

## 2 Resultados

### 2.1 Tabelas

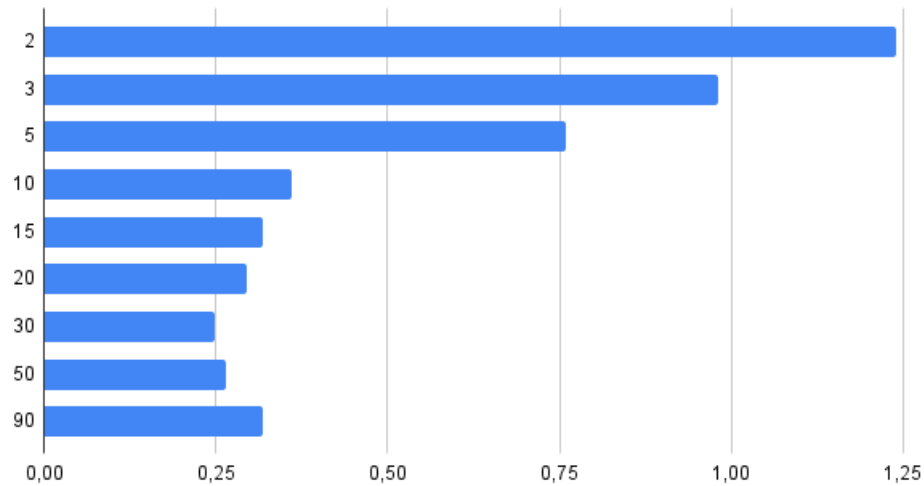
A tabela abaixo trás os tempos de execução do programa (em segundos) na versão sequencial e na paralelizada.

N. Nós	Sequencial	MPI
1	95,1102	95,1102
2	95,1102	117,8356
3	95,1102	93,2660
5	95,1102	72,2204
10	95,1102	34,2729
15	95,1102	30,3737
20	95,1102	28,1064
30	95,1102	23,6775
50	95,1102	25,0998
90	95,1102	30,3135

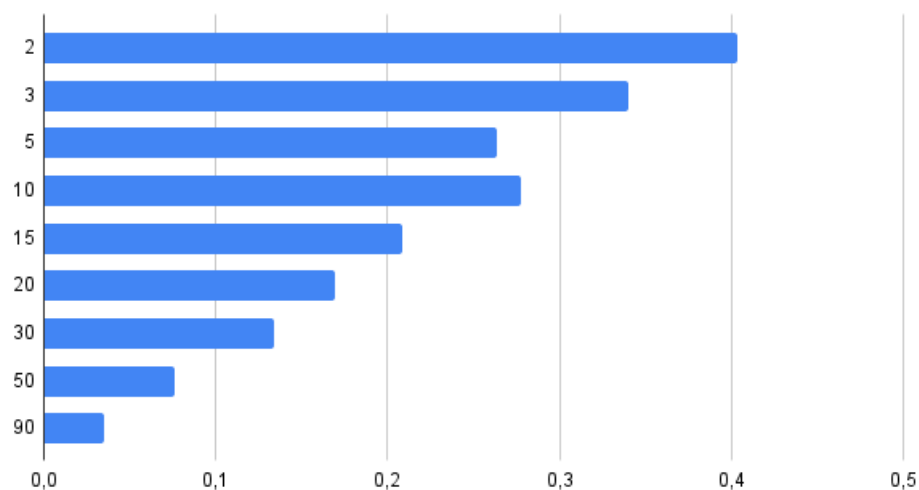
## 2.2 Gráficos

Os gráficos abaixo ilustram valores de speedup e eficiência do programa (horizontal) em relação ao número de nós utilizados na execução (vertical).

Speedup



Eficiência



## 2.3 Discussão

Como pode-se perceber através do gráfico de speedup e da tabela de tempos de execução, o tempo de execução do problema diminui conforme a quantidade de nós vai sendo acrescida. É esperado que esse comportamento ocorra até que a quantidade de nós seja tão grande que o overhead de comunicação (de rede, da transferência dos dados) passe a ser mais custoso do que as próprias operações de adições, e então o tempo de execução volte a crescer. Ainda assim, percebe-se valores satisfatórios de speedup e eficiência, considerando a transferência dos dados.

Um comportamento interessante é o do tempo de execução crescer com 2 nós (em relação a 1 único nó), e isso pode ser justificado também por conta do overhead de comunicação, dado que o vetor é dividido em poucas partes (2) e uma grande quantidade de dados transita entre as máquinas em uma única operação –

diferente do que se teria com dados sendo enviados para 10 máquinas ao mesmo tempo, por exemplo.

De acordo com os gráficos, a quantidade de nós ideal ficaria entre 5 e 10 nós, dado que para cima de 5 o speedup cai bruscamente e para cima de 10 a eficiência também sofre uma queda considerável. É válido ressaltar que por conta de limitações de recursos, não foram utilizados mais de 90 nós.