



# Insper Supercomputação

# Objetivos de aprendizagem

1. Criar implementações eficientes para problemas computacionalmente difíceis;
2. Planejar e projetar sistemas de computação de alto desempenho, escolhendo as tecnologias mais adequadas para cada tipo de aplicação;
3. Utilizar recursos de computação multi-core para melhorar o desempenho de programas sequenciais;
4. Implementar algoritmos ingenuamente paralelizáveis em GPU;
5. Analisar resultados de desempenho levando em conta complexidade computacional e tecnologias usadas na implementação.

# Recursos computacionais

- GCC 8.0 (ou superior) -- C++11
- Linux (Ubuntu 18.04 ou superior)
- Monstrão (containers/VMs)
  - ambiente de testes padrão

# Problemática

- Algoritmos complexos são aplicados em diversas situações para orientar decisões de negócios e para otimizar a alocação/distribuição de recursos
- Um determinado algoritmo é atualmente considerado lento demais
- E agora?

# O que é uma solução de alto desempenho?

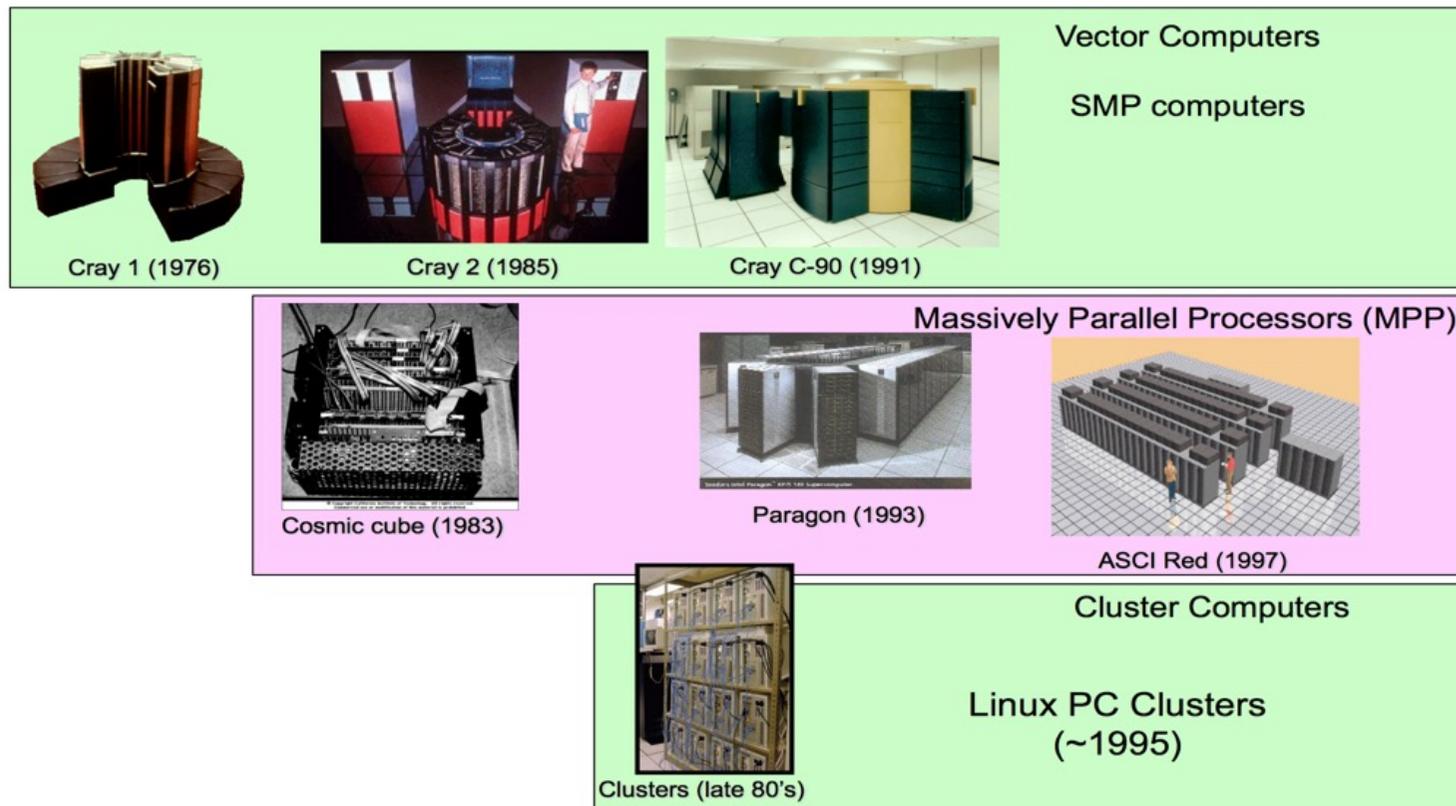
1. Algoritmos eficientes

2. Implementação eficiente

- Cache, paralelismo de instrução
- Linguagem de programação adequada

3. Paralelismo

# Paralelismo



# Resolução no acesso a recursos

- Supercomputação sob demanda

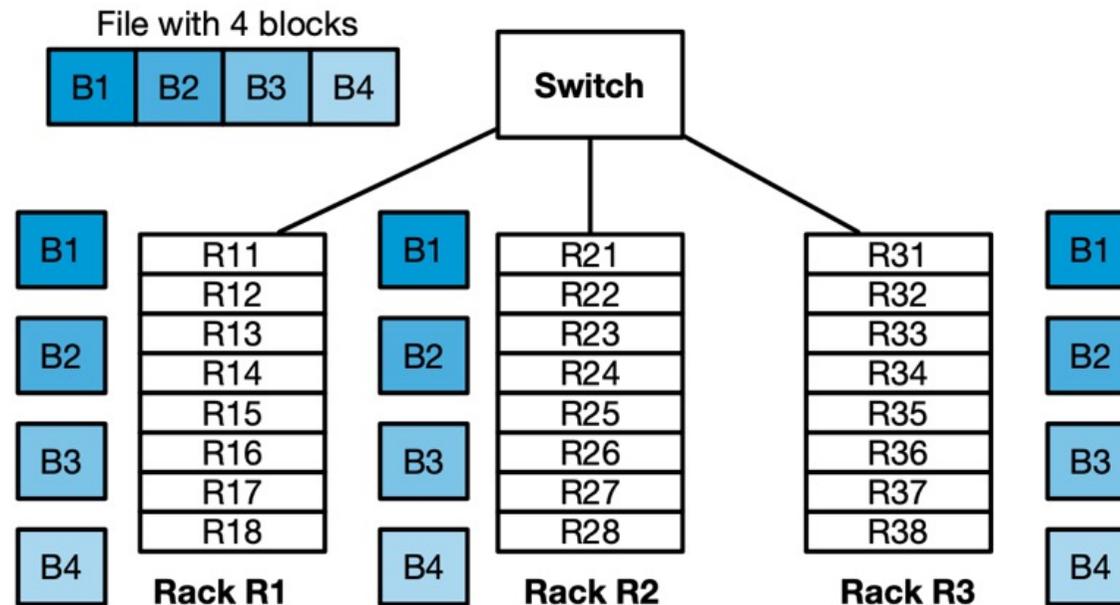


# Escalabilidade

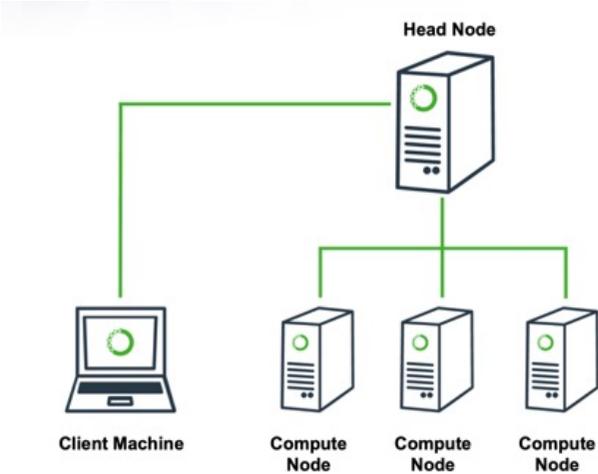
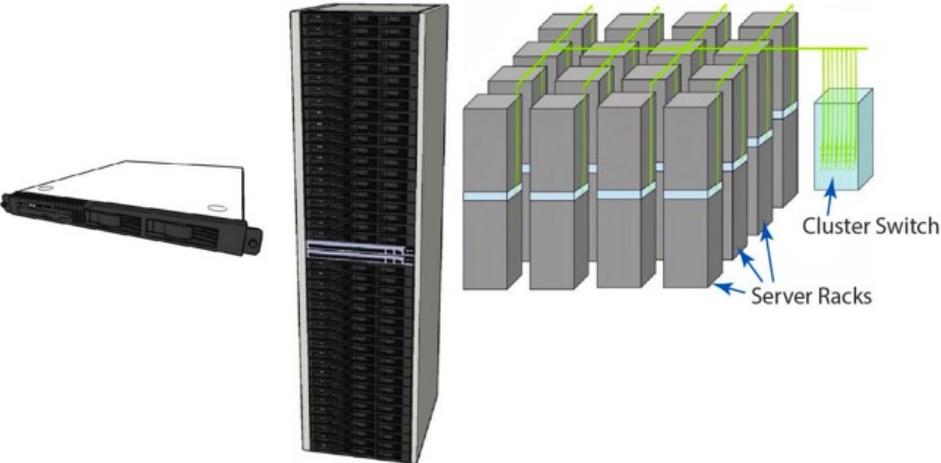
- É a habilidade de um sistema lidar com o aumento da carga de processamento sem apresentar uma degradação significativa em seu desempenho
- Há duas formas de se obter a escalabilidade:
  - **Escalabilidade vertical – *scale-up***
    - Fazer upgrade na infra existente (+ memória, por exemplo)
  - **Escalabilidade horizontal – *scale-out***
    - Adicionar novas máquinas ao parque computacional
    - Distribuir os dados e o trabalho de processamento em diversas máquinas

# Escalabilidade horizontal

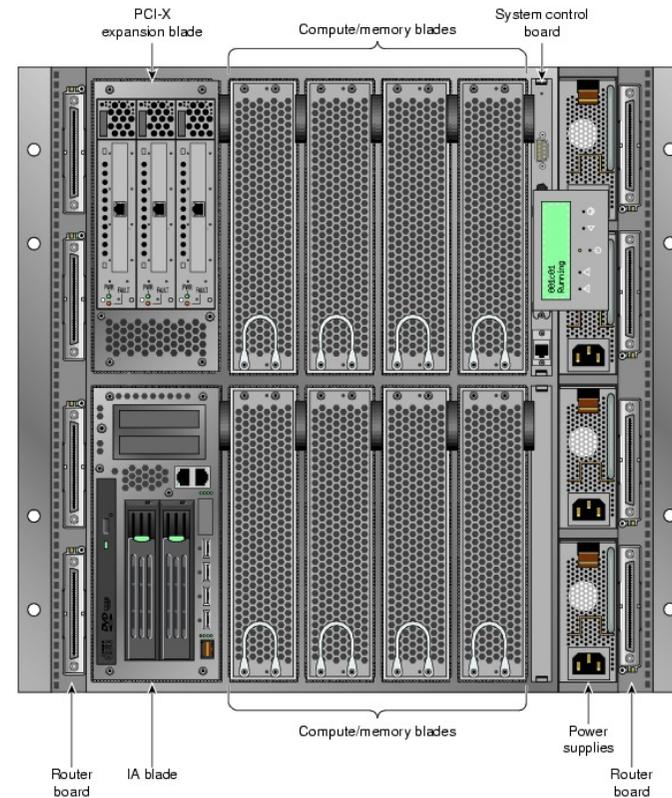
- Podemos fazer uso de um **cluster** de máquinas
- Os dados são armazenados em um sistema de arquivos distribuído (e.g., HDFS)
- Cada arquivo é dividido em blocos de tamanho fixos
- Cada bloco é **replicado** em diversos nós do cluster



# Cluster



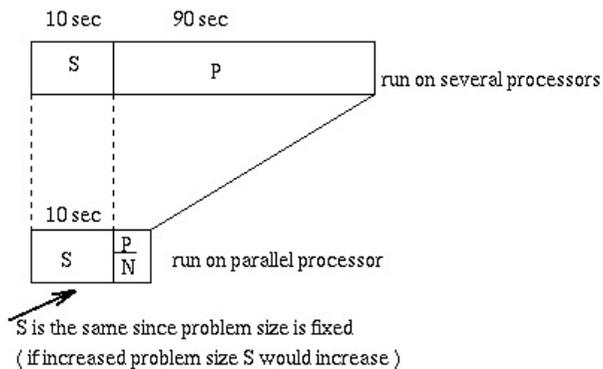
# Supercomputador



# Lei de Amdahl

- Numa aplicação existe sempre uma parte que não pode ser paralelizada
- Seja **S** a parte do trabalho sequêncial, **1-S** é a parte susceptível de ser paralelizada
- Mesmo que a parte paralela seja perfeitamente escalável, o aumento do desempenho (**speedup**) está limitado pela parte sequêncial

n = número de processadores



$$\text{Speedup} = \frac{1}{S + \frac{(1 - S)}{n}}$$

# Speedup

- Se 10% das operações de um código precisam ser feitas sequencialmente, então o speedup não pode ser maior do que 10, independente do número de processadores

$$S = \frac{1}{0.1 + \frac{0.9}{10}} \cong 5.3$$

$p = 10$  processors

$$S = \frac{1}{0.1 + \frac{0.9}{\infty}} = 10$$

$p = \infty$  processors

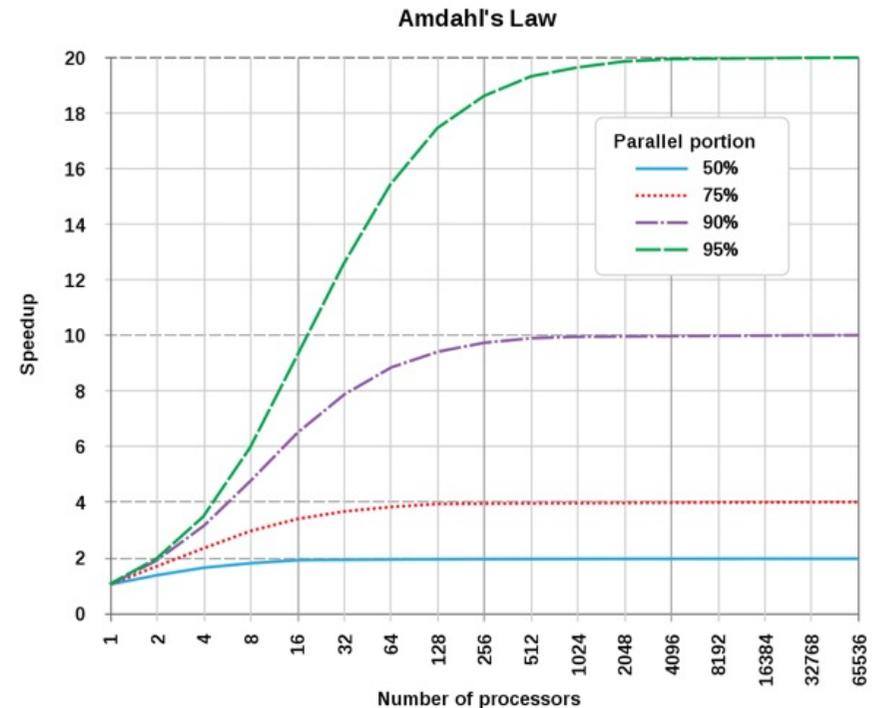


Imagem: Wikipedia

# E o Big Data?

- Gartner:

*Big Data faz referência ao grande volume, variedade e velocidade de dados que **demandam formas inovadoras e rentáveis de processamento da informação**, para melhor percepção e tomada de decisão.*

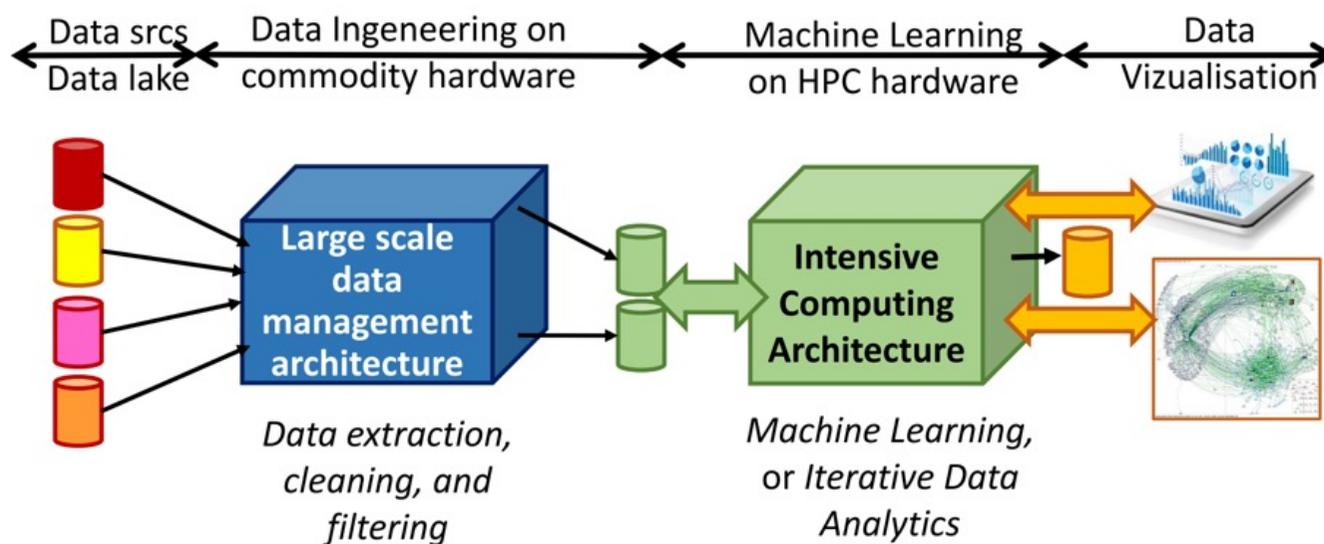
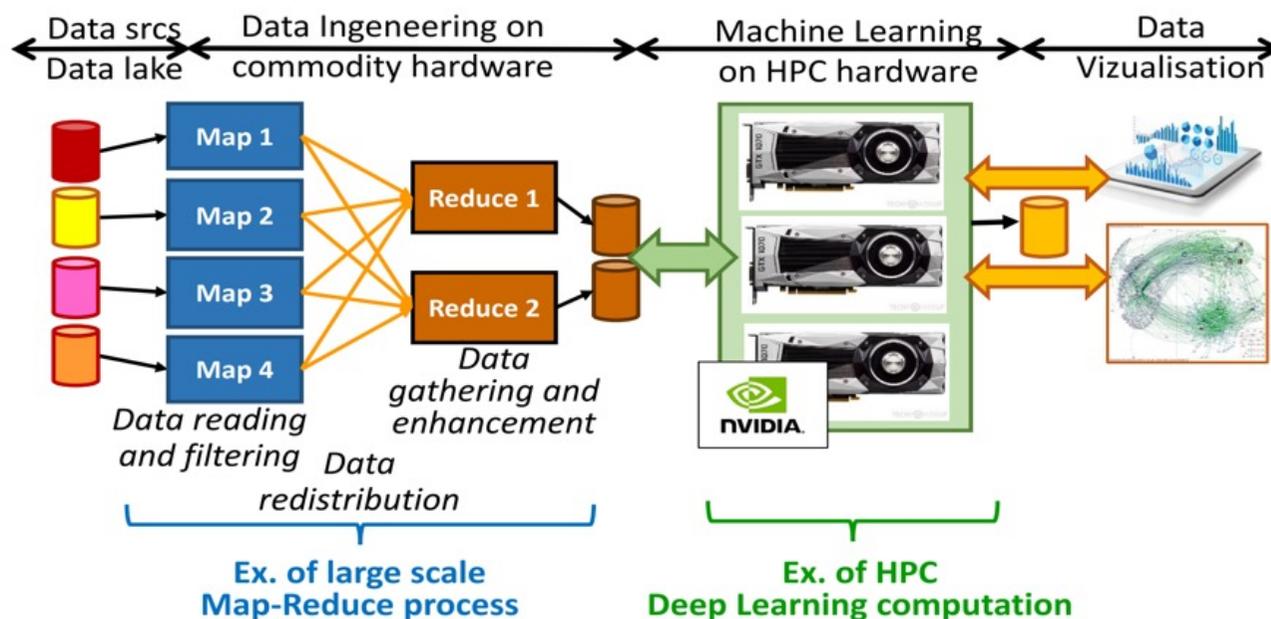


Imagem: Stéphane Vialle & Gianluca Quercini, CentraleSupélec – Université Paris-Saclay

# E o Big Data?

- Gartner:

*Big Data faz referência ao grande volume, variedade e velocidade de dados que **demandam formas inovadoras e rentáveis de processamento da informação**, para melhor percepção e tomada de decisão.*





# Atividade prática

- **Comparando soluções para um problema**

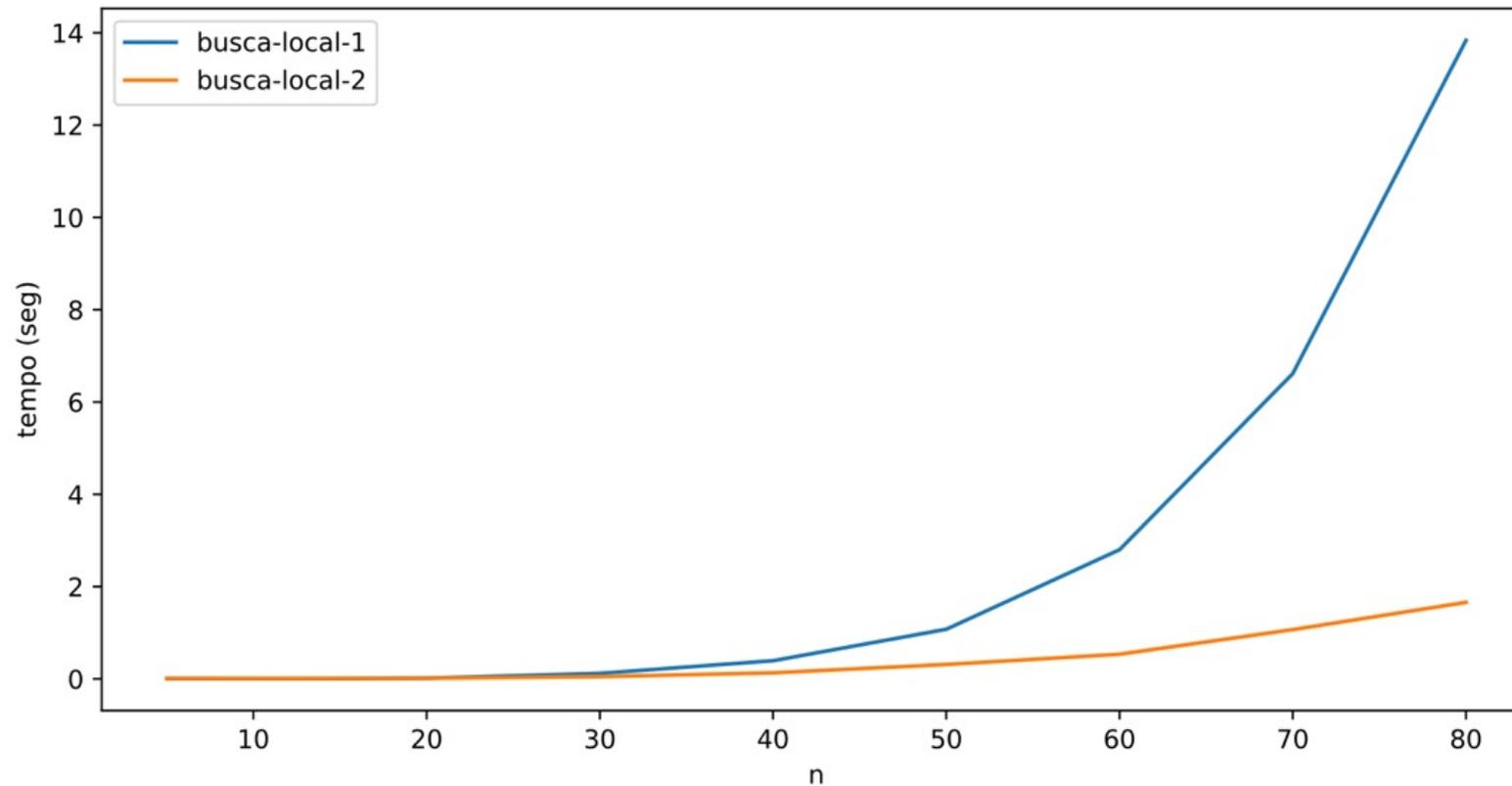
- Medir tempo de execução de programas usando Python
- Ordenar implementações de acordo com sua eficiência
- Discutir custo benefício de diferentes métodos de resolução de um problema.



# Discussão

- **Discussão 1: o quanto um bom algoritmo faz diferença?**

# Discussão – o algoritmo importa!

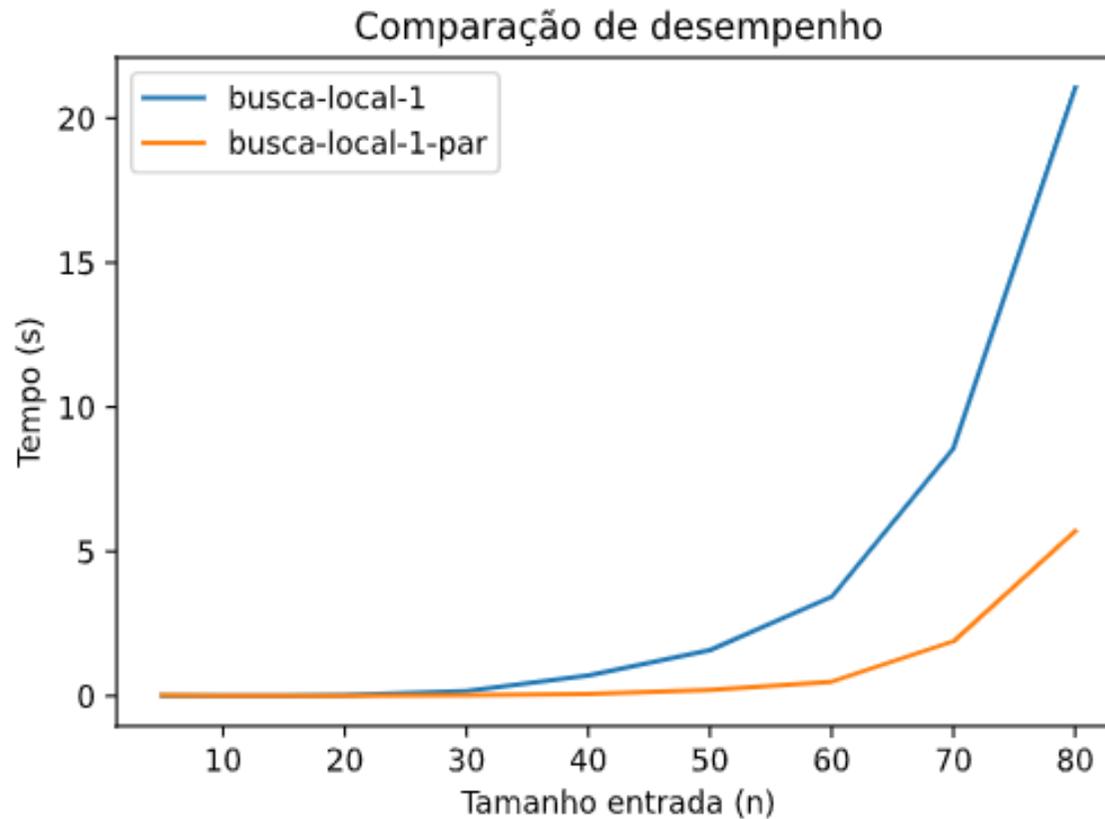




# Discussão

- **Discussão 2: o quanto paralelismo faz diferença?**

# Discussão 2 – Paralelismo importa

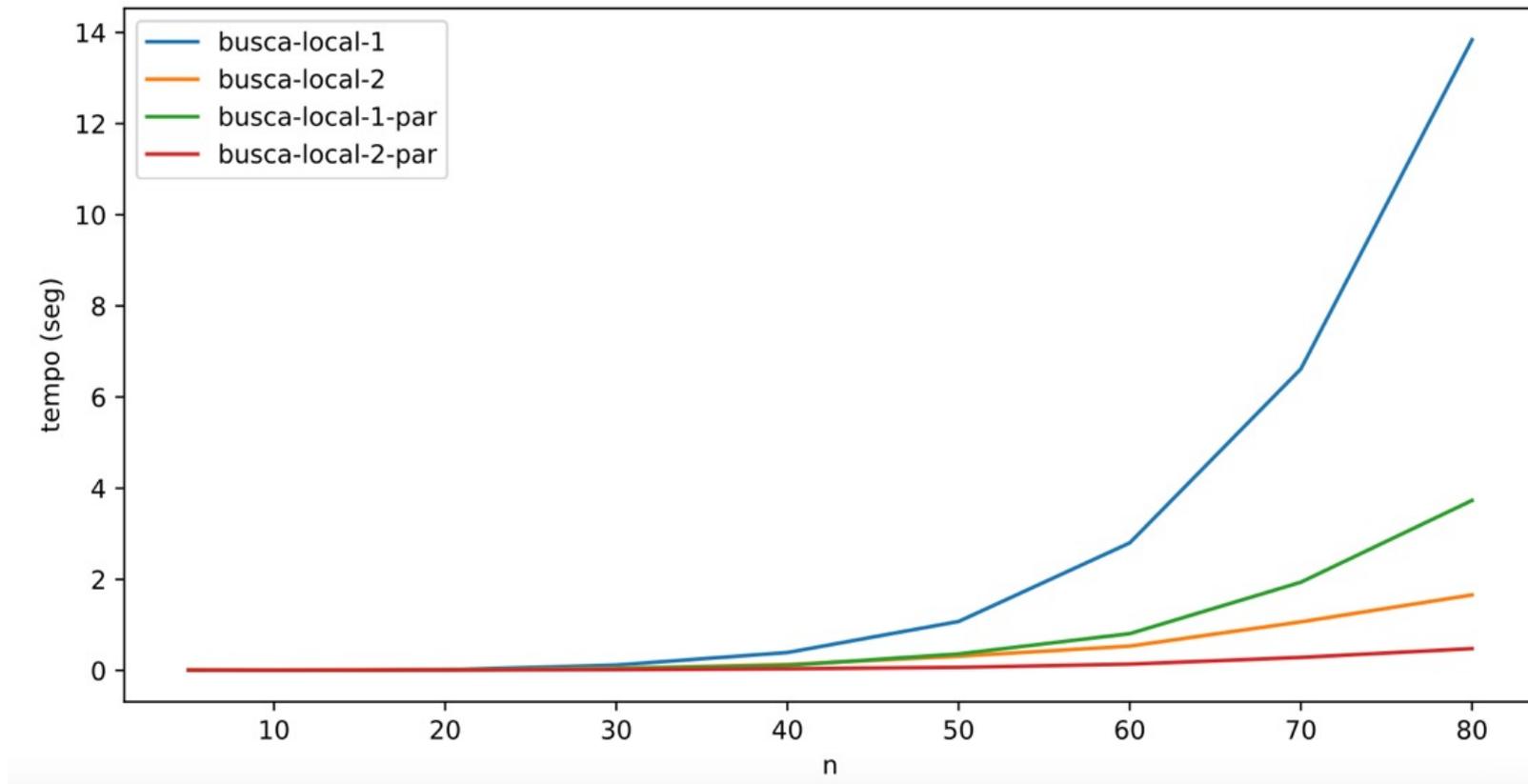




## Discussão

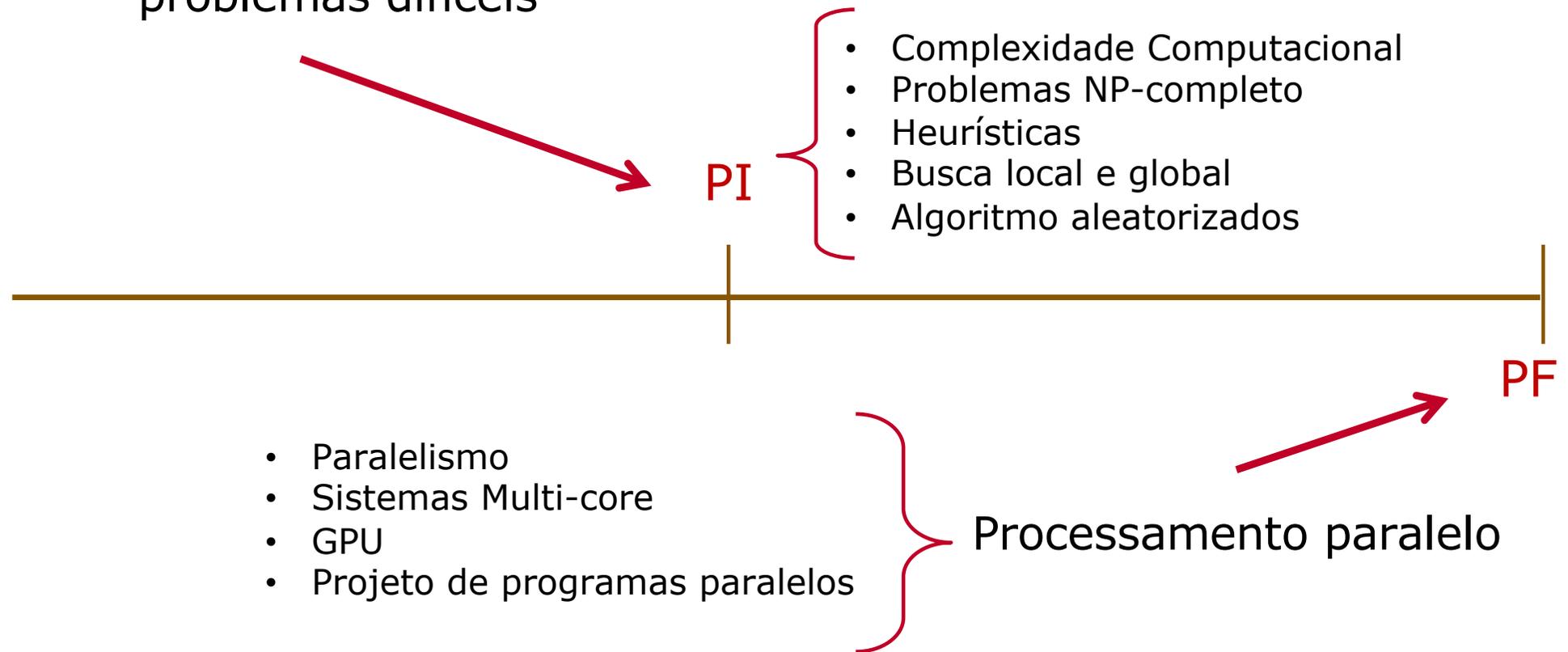
- **Discussão 3: paralelizar é a solução para algoritmos ruins?**

# Paralelizar nem sempre é a (única) solução



# Visão Geral da Disciplina

Estratégias para resolução de problemas difíceis





# Obrigado

Insper

[www.insper.edu.br](http://www.insper.edu.br)