

# Custo Computacional vs. Precisão do Resultado

O trade-off entre o custo computacional e a precisão do resultado

Werbert Arles de Souza Barradas

Universidade Federal do Rio Grande do Norte (UFRN)  
Disciplina de Programação Paralela - DCA3703

20 de agosto de 2025

# Introdução

## O Trade-Off Fundamental

O objetivo é demonstrar empiricamente um dos princípios mais importantes da computação de alto desempenho: a relação direta entre **custo computacional (tempo)** e a **precisão do resultado**.

## O Experimento Modelo

Este estudo utiliza o cálculo de  $\pi$  através de uma série matemática para visualizar este trade-off. Ao variar o número de iterações, mede-se como o tempo de processamento aumenta em troca de uma diminuição no erro da aproximação.

## Relevância

Esta análise serve como um modelo simplificado para entender desafios em domínios como simulações, renderização gráfica e IA.

## A Série de Leibniz

Foi utilizada a série de Leibniz, escolhida pela sua convergência lenta, que torna a relação entre iterações e precisão mais evidente.

$$\frac{\pi}{4} = \sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1} = 1 - \frac{1}{3} + \frac{1}{5} - \dots$$

## Foco do Teste

O código foi compilado sem otimizações (-O0) para isolar o ganho de desempenho obtido **apenas pela reestruturação do código** e pela melhor exploração do hardware (ILP).

## Versão Sequencial

Cria uma dependência de dados no acumulador `sum`, limitando o ILP.

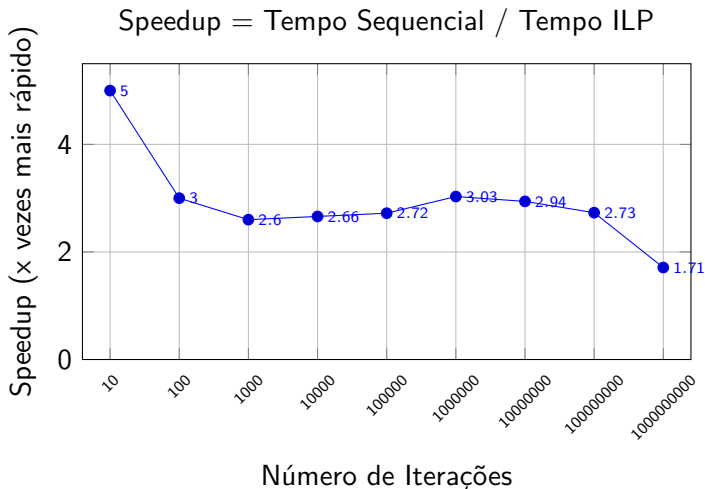
```
double calculate_pi(long long num_iter) {  
    double sum = 0.0;  
    int sign = 1;  
    for (long long i = 0; i < num_iter; i++) {  
        double term = (double)sign / (2.0*i+1.0);  
        sum += term;  
        sign *= -1;  
    }  
    return 4.0 * sum;  
}
```

## Versão Otimizada (ILP)

Usa 4 acumuladores para quebrar a dependência, permitindo que a CPU execute 4 operações em paralelo.

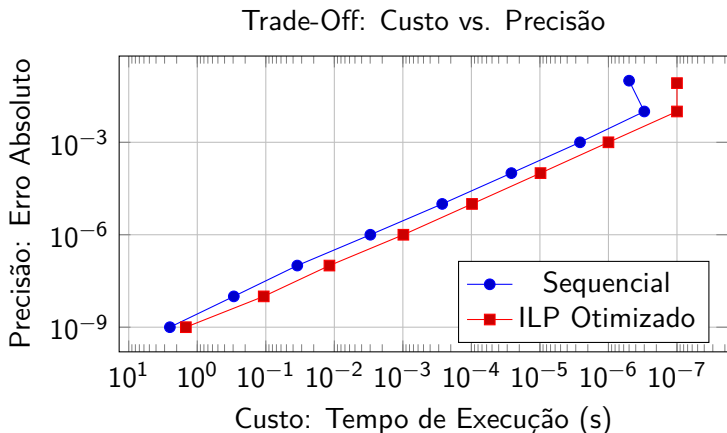
```
double calculate_pi_ilp(long long num_iter) {  
    double s1=0.0, s2=0.0, s3=0.0, s4=0.0;  
    for (long long i=0; i<num_iter; i+=4) {  
        s1 += 1.0 / (2.0*(i+0)+1.0);  
        s2 -= 1.0 / (2.0*(i+1)+1.0);  
        s3 += 1.0 / (2.0*(i+2)+1.0);  
        s4 -= 1.0 / (2.0*(i+3)+1.0);  
    }  
    return 4.0 * (s1+s2+s3+s4);  
}
```

# Speedup da Versão ILP vs. Sequencial (Gráfico de Linha)



**Figure:** A versão otimizada para ILP mostra um ganho de desempenho consistente.

# Relação entre Tempo de Execução e Erro na Aproximação de $\pi$



**Figure:** Para alcançar o mesmo nível de precisão (erro), a versão ILP exige um custo (tempo) muito menor.

## Objetivo

Demonstrar o mesmo trade-off entre custo (tempo) e precisão (qualidade) do cálculo de  $\pi$ , mas num domínio prático: processamento de imagens.



**Figure:** Imagem original em alta resolução (3400x2216) usada no teste.

## O Experimento:

- Redimensionar a imagem para um tamanho menor (800x521).
- Utilizar quatro algoritmos de interpolação com diferentes complexidades:
  - 1 **Nearest Neighbor** (Rápido, baixa qualidade).
  - 2 **Bilinear** (Intermediário).
  - 3 **Bicubic** (Lento, alta qualidade).
  - 4 **Lanczos** (Muito lento, máxima qualidade).



# Comparando a Qualidade dos Resultados

## Análise Detalhada: Qualidade vs. Custo Computacional



**Figure:** Zoom numa área de detalhe da imagem redimensionada por cada algoritmo.

# Comparando a Qualidade dos Resultados

- **Nearest Neighbor:** Produz um forte serrilhado (*aliasing*).
- **Bilinear:** Suaviza o serrilhado, mas causa um borrão (*blur*), perdendo nitidez.
- **Bicubic e Lanczos:** Oferecem um resultado superior, preservando a nitidez das bordas. O Lanczos mantém os detalhes mais finos.

# Medindo o Tempo para Atingir a Qualidade

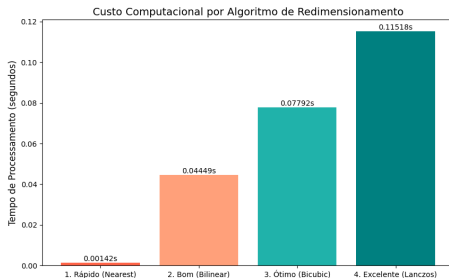


Figure: Tempo de execução de cada algoritmo de redimensionamento.

## Correlação Custo-Qualidade

Os dados mostram uma correlação direta: para obter a precisão visual superior do algoritmo Lanczos, é necessário um investimento computacional drasticamente maior.

# Conclusão do Estudo de Caso

- Assim como no cálculo de  $\pi$ , onde mais iterações (custo) levam a um erro menor (precisão), no processamento de imagens, algoritmos mais complexos (custo) resultam em maior fidelidade visual (precisão).
- A escolha do método ideal depende sempre dos requisitos da aplicação.

# Conclusão Geral do Projeto

## Tese Comprovada

Este projeto demonstrou com sucesso, tanto de forma quantitativa (cálculo de  $\pi$ ) quanto qualitativa (processamento de imagens), o trade-off fundamental entre custo computacional e precisão do resultado.

- **No cálculo de  $\pi$ :** Foi provado que a reestruturação do código para explorar o ILP (paralelismo em nível de instrução) permite alcançar a mesma precisão em um tempo significativamente menor, otimizando o uso do hardware.
- **No processamento de imagens:** Foi demonstrado que diferentes algoritmos representam pontos distintos na curva custo-benefício, onde maior fidelidade visual exige um custo de processamento exponencialmente maior.