

# Leis de Amdahl e de Gustafson

João Vitor de Alencar<sup>1</sup>, Lucas Reinaldo Pacífico<sup>2</sup>, Marcelo Augusto Fiedler Schmoeller<sup>3</sup>

Área de Engenharias, Exatas e Tecnológicas (Engetec) — Universidade da Região de Joinville (Univille)

Caixa Postal 89219-710 - Joinville - SC - Brazil

joao.alencar@univille.br<sup>1</sup>, lucaspacifico@univille.br<sup>2</sup>,  
marceloschmoeller@univille.br<sup>3</sup>

**Resumo** — Este artigo compara as Leis de Amdahl e Gustafson, destacando suas diferenças na análise de desempenho em programação paralela. Amdahl foca em problemas de tamanho fixo, enquanto Gustafson considera a escalabilidade do problema com o número de processadores. A análise comparativa indica qual lei é mais adequada conforme o contexto, auxiliando na otimização de sistemas concorrentes.

**Abstract** — This paper compares Amdahl's and Gustafson's Laws, highlighting their differences in analyzing parallel programming performance. Amdahl focuses on fixed-size problems, while Gustafson considers problem scalability with processor count. The comparative analysis shows which law suits different contexts, aiding concurrent system optimization.

## 1. Introdução

A computação paralela consiste na técnica de dividir um problema computacional em múltiplas tarefas menores que podem ser executadas simultaneamente em diferentes processadores ou núcleos. Com o avanço das técnicas empregadas e a crescente demanda por sistemas mais rápidos e eficientes, a programação concorrente tornou-se uma área essencial da ciência da computação.

O desempenho de programas paralelos depende diretamente da capacidade de distribuir tarefas entre múltiplos processadores. Para avaliar esse desempenho, duas leis principais são comumente utilizadas: a Lei de Amdahl e a Lei de Gustafson.

Este artigo apresenta uma visão detalhada dessas leis, suas formulações, bem como uma análise comparativa, destacando como esses conceitos podem orientar a programação concorrente para otimizar o uso dos recursos computacionais.

## 2. Lei de Amdahl

Proposta por Gene Amdahl em 1967, a é utilizada para estimar o ganho máximo de velocidade na execução de um programa que pode ser paralelizado parcialmente. A fórmula central é dada por:

$$S = (1 - P) + NP1$$

Sendo que:

$S$  é o *speedup* máximo esperado,

$P$  é a fração do programa que pode ser paralelizada,

$N$  é o número de processadores.

Aqui, podemos definir *speedup* como:

$$\frac{\text{Desempenho da tarefa com aprimoramentos aplicados}}{\text{Desempenho da tarefa sem aprimoramentos aplicados}}$$

Para Reddy (2011), ela verbalmente pode ser expressa da seguinte maneira: "a melhoria geral do desempenho obtida pela otimização de uma única parte de um sistema é limitada pela fração de tempo em que a parte melhorada é realmente usada".

Em outras palavras, a lei mostra que o ganho de desempenho é limitado pela parte sequencial do programa, que não pode ser paralelizada. Mesmo com um número infinito de processadores, o *speedup* máximo é restrito. Isso evidencia que para programas com grande parcela sequencial, a paralelização traz ganhos marginais.

Um exemplo prático:

	Comp. A	Comp. B	Comp. C
Programa 1	1	10	20
Programa 2	1000	100	20
Total	1001	110	40

Observando apenas a velocidade relativa de um único programa, pode-se pensar:

- "O computador B é 10 vezes mais rápido que o A no programa 1"
- "O computador C é 50 vezes mais rápido que o A no programa 2"

Todavia, não significa que o **desempenho geral** corresponderá a esses ganhos, devido à Lei de Amdahl. Como vimos, o *speedup* geral é limitado pela fração de tempo gasto nas partes inalteradas.

No Computador A, o programa 1 representa basicamente 0,1% do tempo total. Portanto, mesmo se o programa 2 torne-se infinitamente rápido, ainda ficará preso àquele 1 segundo do programa 1 — o "gargalo serial". Portanto, isso demonstra que o tempo total do Computador C (40) não chega nem perto de ser "50x mais rápido" do que o total do Computador A (1001).

### 3. Lei de Gustafson

A Lei de Gustafson, proposta por John L. Gustafson em 1988, surge como uma resposta às limitações da Lei de Amdahl, especialmente no contexto de escalabilidade de problemas. Diferentemente da Lei de Amdahl, que considera um problema fixo, Gustafson assume que o tamanho do problema cresce proporcionalmente ao número de processadores, mantendo constante o tempo de execução.

A fórmula de Gustafson:

$$S(p) = p - \alpha \cdot (p-1)$$

Sendo:

*$S(p)$  é ganho de desempenho (speedup) usando processadores.*

*$p$  é número de processadores*

*$\alpha$  é fração do programa que é sequencial (não pode ser paralelizada)*

Em poucas palavras, a lei de Gustafson diz que ter mais processadores não serve só para acelerar programas, mas serve também para resolver problemas maiores no mesmo tempo.

#### **4. Comparação: Prós, Contras e Aplicações**

Para compreender melhor o impacto de ambas as leis na prática na programação concorrente, é essencial analisar as particularidades das duas abordagens e seus pressupostos. Dessa forma, pode ser feito um comparativo baseado em características de projetos do mundo real que seriam consideradas ao pensar nos seus usos.

##### **4.1 Prós da Lei de Amdahl**

A Lei de Amdahl fornece uma estimativa clara e direta do limite teórico de aceleração para programas com partes sequenciais significativas, isso significa que mesmo usando vários processadores a velocidade do programa só pode melhorar até certo ponto por causa das partes que precisam ser feitas em sequência. Isto é crucial para entender o impacto das seções não paralelizáveis no desempenho geral.

Esta lei é particularmente útil para otimizar programas onde o tempo total de execução deve ser reduzido para um problema específico, auxiliando no dimensionamento eficiente do hardware necessário.

##### **4.2 Contras da Lei de Amdahl**

A Lei de Amdahl assume que o tamanho do problema permanece constante, o que pode não refletir cenários reais onde o problema aumenta com a disponibilidade de mais recursos computacionais. Essa premissa pode levar a subestimar o potencial real em aplicações escaláveis, limitando a visão sobre o uso eficiente de múltiplos processadores em situações práticas.

##### **4.3 Prós da Lei de Gustafson**

A Lei de Gustafson é vantajosa ao considerar o crescimento do problema proporcional ao aumento do número de processadores, ou seja, conforme usamos mais

processadores podemos aumentar o tamanho do problema que queremos resolver. Dessa forma aproveita-se melhor a capacidade dos computadores, assim refletindo melhor a realidade de muitas aplicações científicas e industriais mais robustas.

Ela mostra que é possível obter ganhos quase lineares, incentivando o desenvolvimento de soluções que exploram a escalabilidade e o processamento paralelo em larga escala.

#### **4.4 Contras da Lei de Gustafson**

A Lei de Gustafson pode ser otimista demais para problemas com uma fração significativa de código sequencial ou quando o aumento do tamanho do problema não é viável na prática. Além disso, nem todas as aplicações permitem a expansão do problema sem afetar a qualidade ou o tempo total de execução, o que pode limitar a aplicabilidade dessa lei em certos contextos.

### **5. Conclusão**

Basicamente, a escolha entre aplicar a Lei de Amdahl ou a de Gustafson depende do contexto do problema e da arquitetura do sistema. Para problemas de tamanho fixo e quando a redução do tempo total é uma prioridade, a Lei de Amdahl é mais adequada.

Já para cenários em que é possível aumentar o tamanho do problema com o número de processadores, a Lei de Gustafson oferece uma visão mais realista e otimista do potencial. Enquanto Amdahl enfatiza a importância da parte sequencial e limita o speedup para problemas fixos, Gustafson amplia a visão para problemas escaláveis mostrando que o aumento do tamanho do problema pode justificar o uso de mais processadores de forma eficiente.

### **Referências**

MISHRA, D. Um mergulho profundo na lei de Amdahl e na lei de Gustafson. HackerNoon, 11 nov. 2023. Disponível em: <https://hackernoon.com/lang/pt/um-mergulho-profundo-na-lei-amdahls-e-na-lei-gustafsons>. Acesso em: 5 ago. 2025.

HPC Wiki. Scaling. Disponível em: <https://hpc-wiki.info/hpc/Scaling>. Acesso em: 8 ago. 2025.

REDDY, M. API Design for C++. Burlington, Massachusetts: Morgan Kaufmann Publishers, 2011. DOI: 10.1016/C2010-0-65832-9. ISBN: 978-0-12-385003-4.

SHANTHI, A. P. Summarizing Performance, Amdahl's law and Benchmarks. In: Computer Architecture: Engineering And Technology. Disponível em: <https://www.cs.umd.edu/~meesh/411/CA-online/chapter/summarizing-performance-amdahls-law-and-benchmarks/index.html>. Acesso em: 8 ago. 2025.

HENNESSY, J. L.; PATTERSON, D. A. *Computer Architecture: A Quantitative Approach*. Morgan Kaufmann, 2003. ISBN: 978-8178672663.