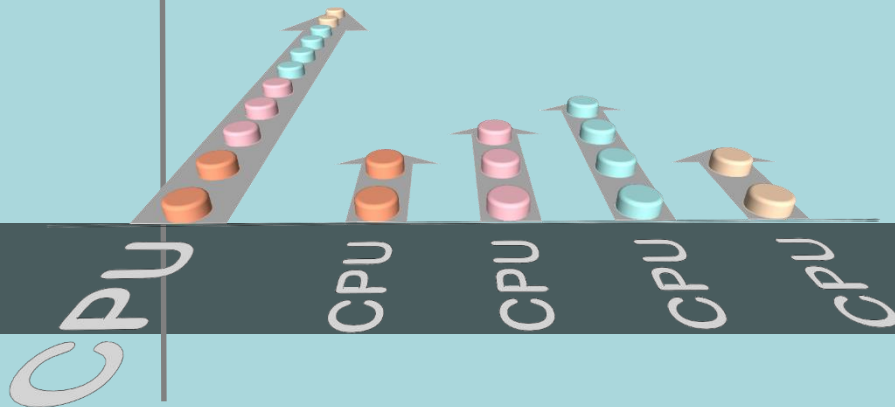


Conceptos Fundamentales de Paralelismo



Computación de alto rendimiento

Objetivos

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

Reflexión

Presentación de
prácticas

🔧 **Objetivo general:** Comprender el paralelismo y su papel en la computación moderna.

🔧 **Objetivos específicos:**

1. Definir y entender qué es la computación paralela.
2. Diferenciar entre computación secuencial y paralela.
3. Explorar los tipos principales de paralelismo.
4. Conocer las limitaciones del paralelismo (Ley de Amdahl y Gustafson).
5. Ver ejemplos de aplicaciones prácticas en la industria.



¿Qué es la computación paralela?

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

Reflexión

Presentación de
prácticas

- ❖ **Definición:** La computación paralela es un paradigma donde múltiples tareas se ejecutan simultáneamente para mejorar el rendimiento y reducir el tiempo de ejecución.
- ❖ **Motivación:** Con el crecimiento del volumen de datos y la complejidad de los problemas computacionales, dividir las tareas es clave para procesar eficientemente.
- ❖ **Ejemplo:**



Computación Secuencial vs. Paralela

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

Reflexión

Presentación de
prácticas

Ventajas de la Computación Paralela:

- Resuelve problemas complejos no manejables por una sola CPU.
- Reduce tiempos de ejecución significativamente.
- Ejecuta tareas simultáneas, aumentando el rendimiento.
- Divide grandes problemas en partes más pequeñas y fáciles de gestionar.
- Mejora el balance entre costo y rendimiento respecto a la computación secuencial.
- Es escalable y adaptable a mayores necesidades.
- Ejecuta programas de orden de complejidad mayor

Desventajas de la Computación Paralela:

- Consume más energía.
- Es más complicado programar y sincronizar procesos.
- Puede haber retrasos en la comunicación entre tareas.
- Más componentes implican mayor riesgo de fallos.
- Costos altos de implementación y mantenimiento.
- Riesgo de condiciones de carrera y corrupción de datos sin buena coordinación.
- Difícil de automatizar hasta con IA en el proceso

Motivación del Paralelismo

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

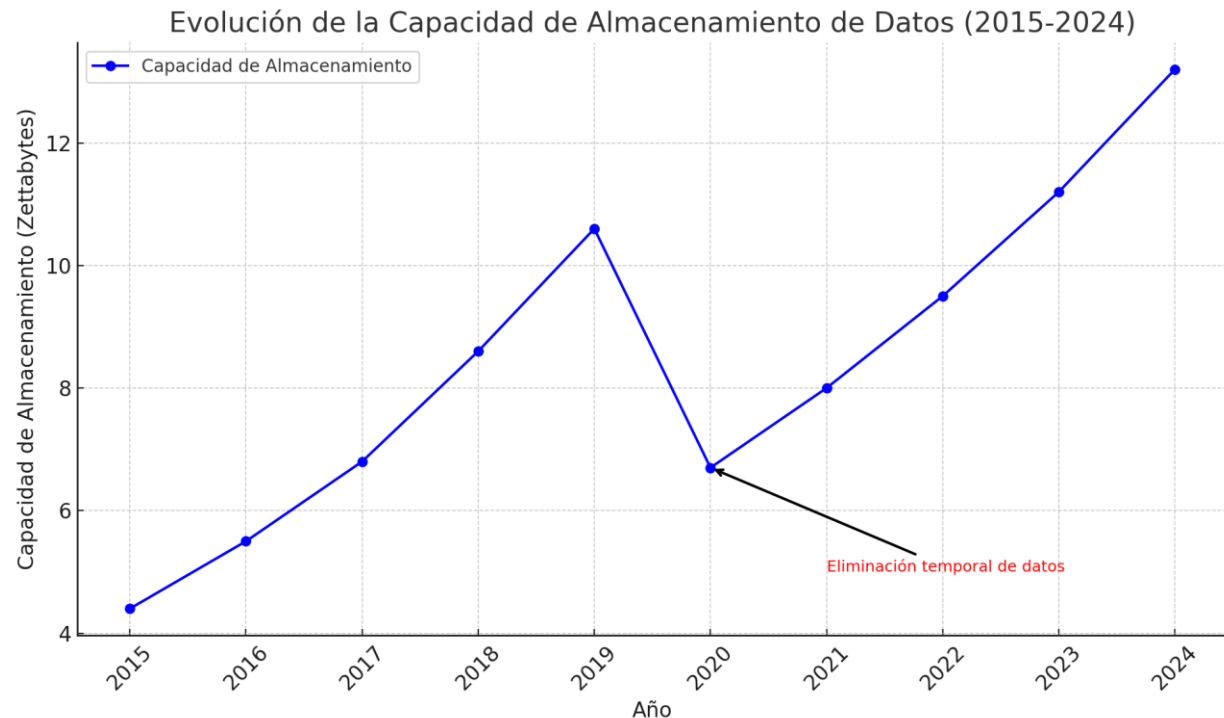
Limitaciones

Reflexión

Presentación de
prácticas

¿Por qué necesitamos paralelismo?

- 🔑 **Crecimiento exponencial de datos:** Procesar grandes volúmenes de datos rápidamente es esencial en aplicaciones como IA y Big Data.



Nota: Un **zettabyte (ZB)** equivale a 1,000 exabytes, o un trillón de gigabytes

Fuente. <https://almacenamientoit.ituser.es>

Motivación del Paralelismo

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

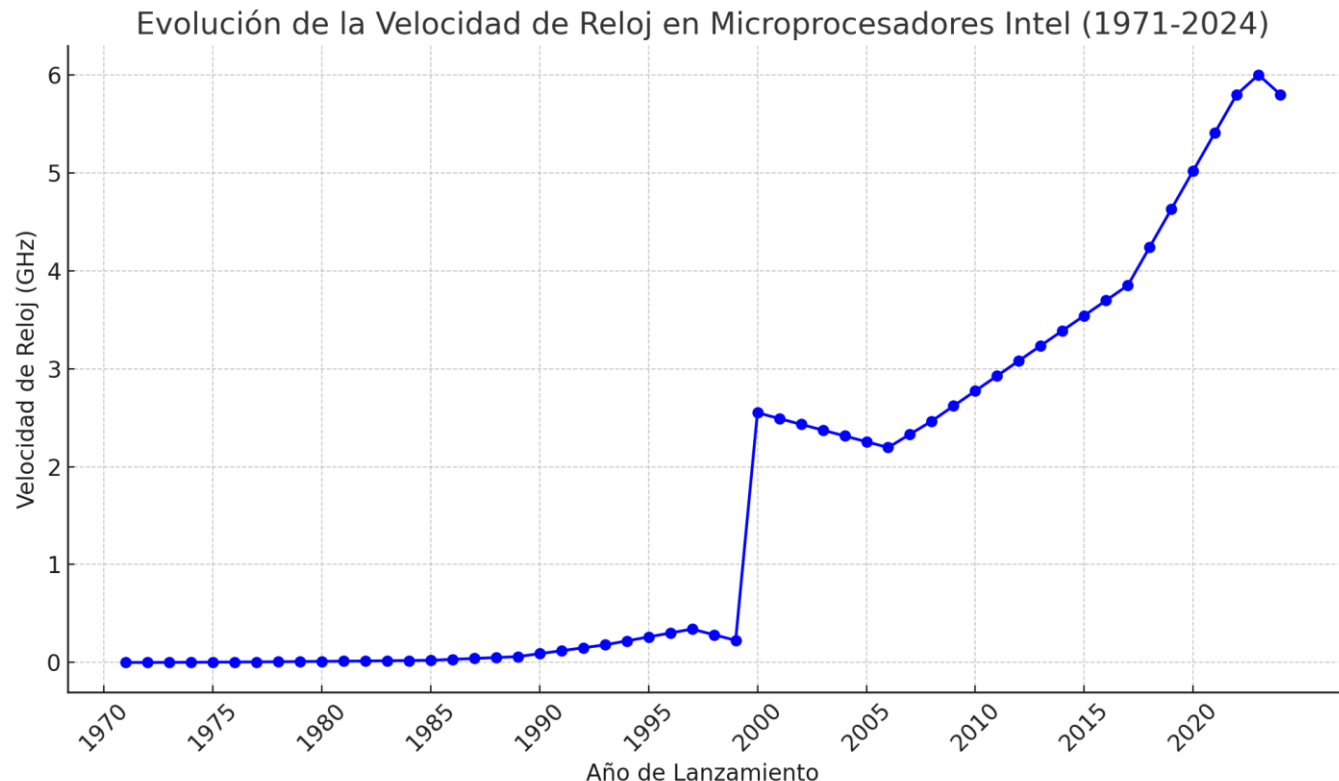
Limitaciones

Reflexión

Presentación de
prácticas

¿Por qué necesitamos paralelismo?

- 🔧 **Estancamiento en la velocidad de los procesadores:** Aumentar la frecuencia del reloj se ha desacelerado, así que el enfoque actual es añadir más núcleos.



Fuente de los datos: <https://www.intel.la>

Motivación del Paralelismo

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

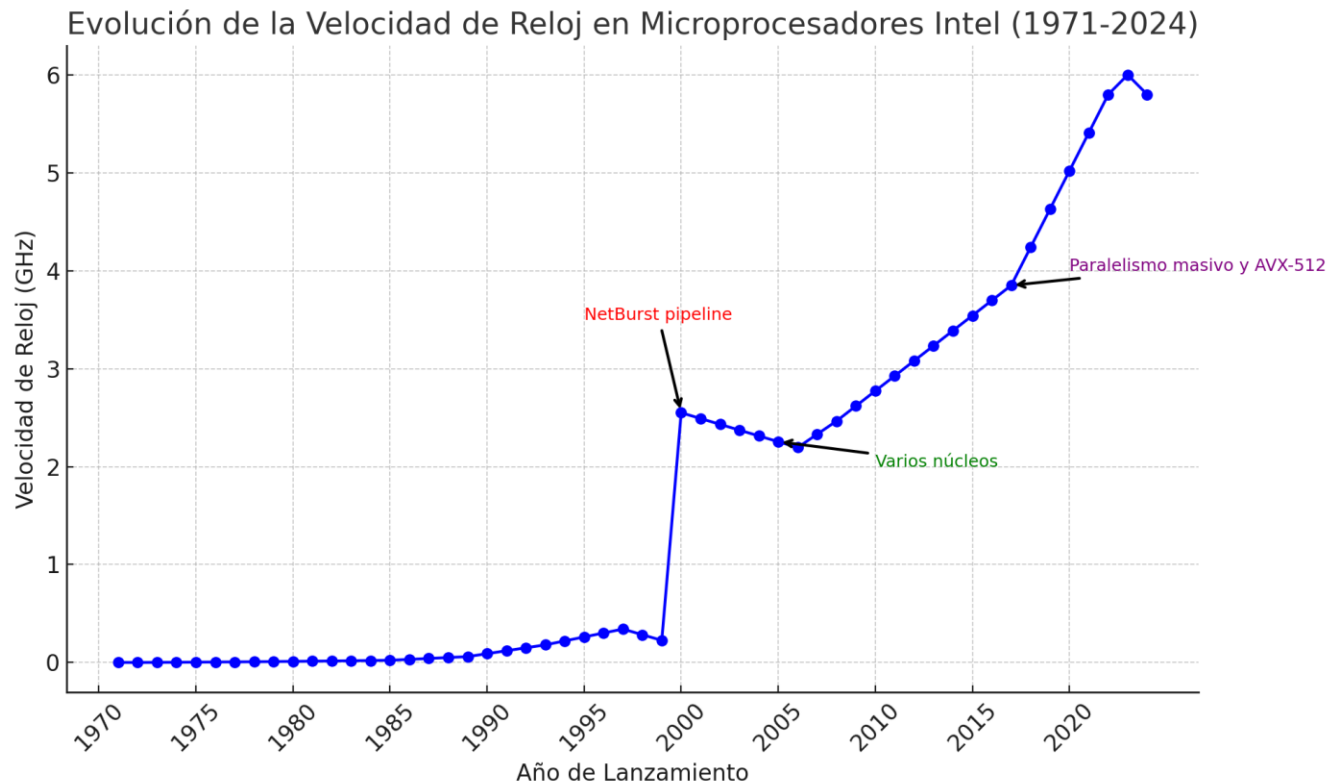
Limitaciones

Reflexión

Presentación de
prácticas

¿Por qué necesitamos paralelismo?

- ❖ **Estancamiento en la velocidad de los procesadores:** Aumentar la frecuencia del reloj se ha desacelerado, así que el enfoque actual es añadir más núcleos.



Fuente de los datos: <https://www.intel.la>

Motivación del Paralelismo

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

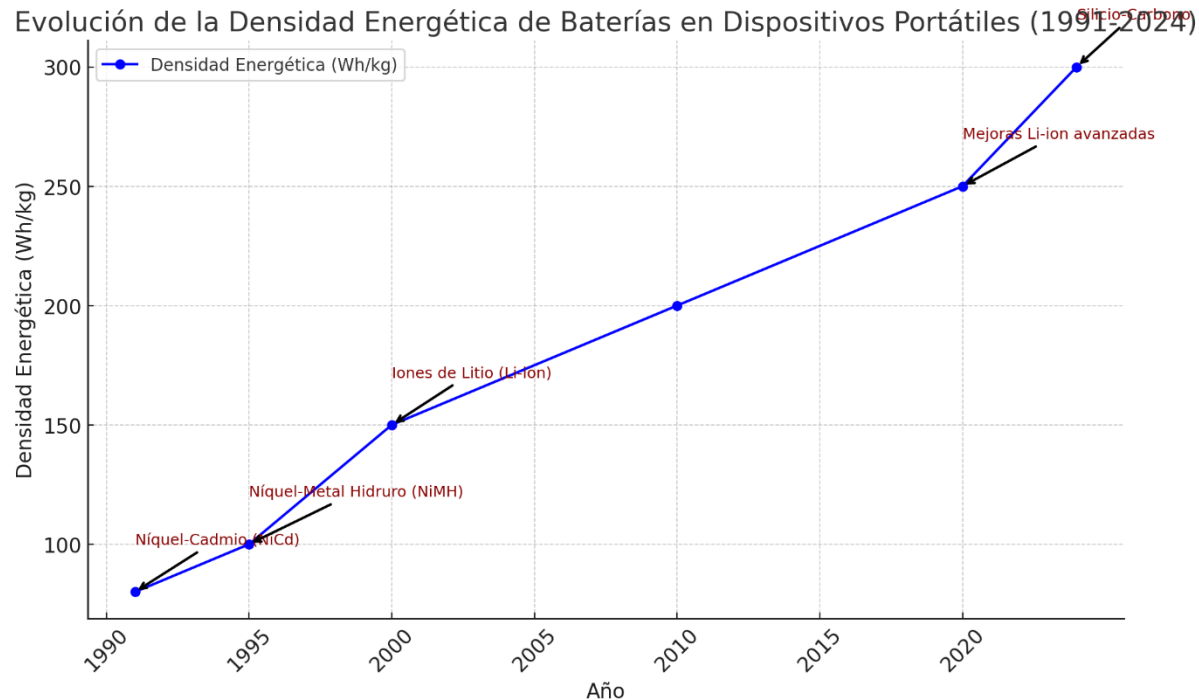
Limitaciones

Reflexión

Presentación de prácticas

¿Por qué necesitamos paralelismo?

- ⚙️ **Eficiencia energética:** Aprovechar el paralelismo puede ser más eficiente que depender de un único procesador.
- ⚙️ Ejecución de código en móviles: la tecnología de baterías no evoluciona tan rápidamente como la de los procesadores



Fuentes: <https://es.wikipedia.org>
<https://cincodias.elpais.com>

Aplicaciones del Paralelismo

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

Reflexión

Presentación de
prácticas

- 🛡️ **Inteligencia Artificial (IA):** Acelera el entrenamiento de modelos complejos.
- 🛡️ **Big Data:** Permite procesar grandes conjuntos de datos de forma distribuida.
- 🛡️ **Simulaciones científicas:** Predicción del tiempo, modelado molecular, y simulaciones físicas.
- 🛡️ **Gráficos de videojuegos:** Generación de gráficos realistas y simulación física.

Aplicaciones del Paralelismo

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

Reflexión

Presentación de
prácticas

Aplicación del Paralelismo en Inteligencia Artificial (IA)

- **Aceleración con GPUs:** Las GPUs, diseñadas para ejecutar miles de hilos en paralelo, son clave en el entrenamiento rápido de redes neuronales profundas.
- **Procesamiento paralelo de datos:** Durante el entrenamiento, grandes conjuntos de datos (por ejemplo, imágenes) se dividen en fragmentos procesados simultáneamente.

Aplicaciones del Paralelismo

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

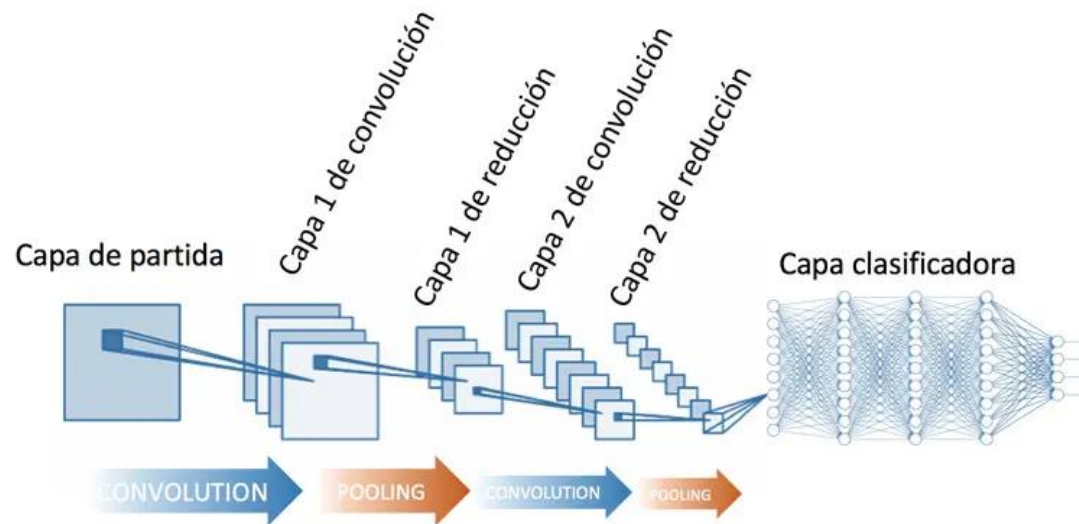
Limitaciones

Reflexión

Presentación de
prácticas

Aplicación del Paralelismo en Inteligencia Artificial (IA)

- **Ejemplo práctico:** En las redes neuronales convolucionales (CNNs), cada imagen se separa en múltiples bloques, permitiendo aplicar filtros convolucionales a cada región en paralelo.



<https://www.diegocalvo.es/red-neuronal-convolucional/>

Beneficios:

- Reducción drástica del tiempo de entrenamiento (de días a horas).
- Mejora de la escalabilidad, permitiendo entrenar modelos más grandes.
- Optimización energética al usar mejor los recursos del hardware.

Aplicaciones del Paralelismo

Big Data y su Procesamiento Distribuido:



Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

Reflexión

Presentación de
prácticas

Aplicaciones del Paralelismo

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

Reflexión

Presentación de
prácticas

Big Data y su Procesamiento Distribuido:

- 🔧 **Procesamiento distribuido:** Herramientas como **Apache Spark** dividen la carga de datos entre múltiples nodos, acelerando el análisis.
- 🔧 **Paralelismo natural:** Los datos masivos se dividen en partes que pueden procesarse en paralelo.
- 🔧 **Escalabilidad horizontal:** Permite agregar más nodos al clúster según el crecimiento de los datos.
- 🔧 **Manejo de datos heterogéneos:** Capacidad para procesar datos estructurados y no estructurados (imágenes, texto, videos).
- 🔧 **Tolerancia a fallos:** Los sistemas replican datos, garantizando continuidad ante fallos.

Aplicaciones del Paralelismo

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

Reflexión

Presentación de prácticas

Simulaciones Científicas:

🔧 Meteorología:

🔧 Gran cantidad de datos:

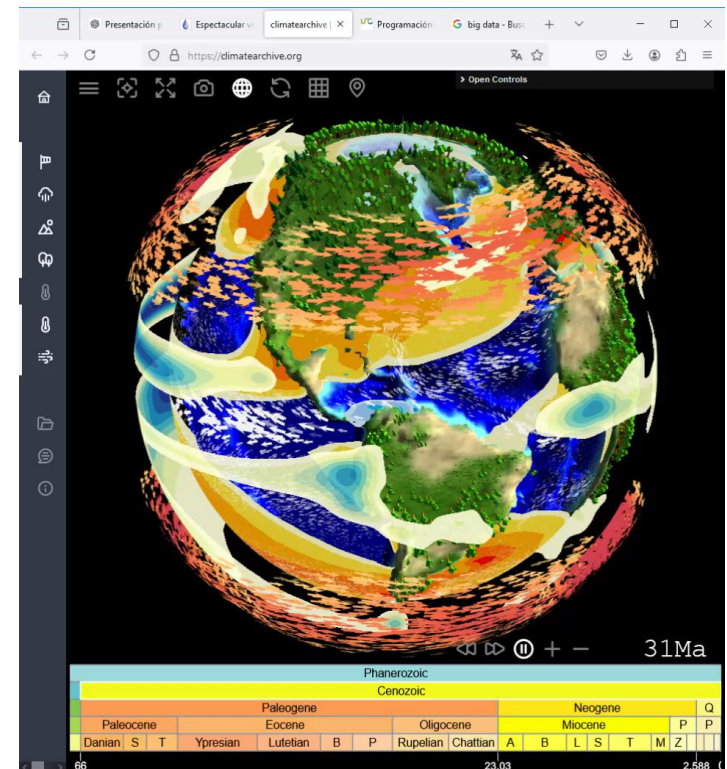
- Temperatura, humedad, presión, viento, radiación solar, etc.
- Escala: Se generan y procesan datos a nivel local, regional y global.

🔧 Cálculos intensivos:

- Los modelos climáticos dependen de ecuaciones complejas de dinámica de fluidos, transferencia de calor y humedad.
- Estos cálculos deben repetirse para cada punto geográfico y momento del tiempo.

🔧 Escalabilidad:

- A medida que aumentan la resolución y los datos, se pueden añadir más nodos de procesamiento.



<https://climatearchive.org/>

Aplicaciones del Paralelismo

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

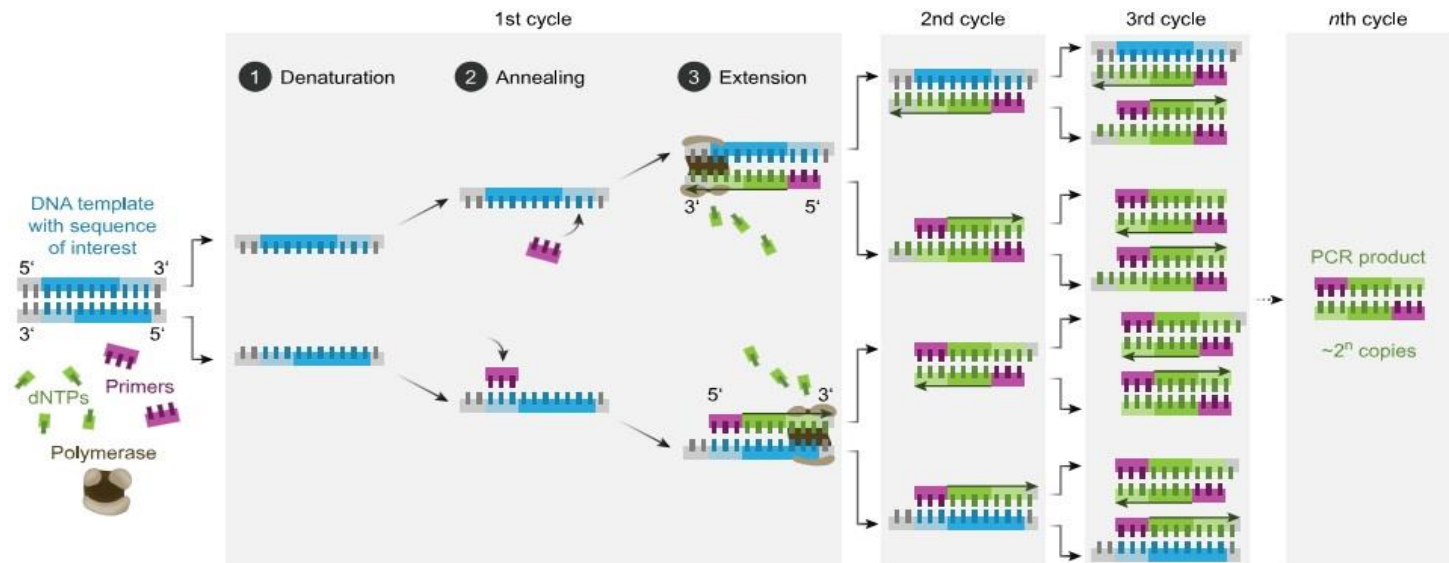
Reflexión

Presentación de prácticas

Simulaciones Científicas:

Computación masiva en la secuenciación del ADN y el proceso de PCR

- ❖ **Procesamiento masivo de datos:** Permite dividir y analizar **millones de fragmentos** de ADN simultáneamente.
- ❖ **Iteraciones rápidas:** La paralelización reduce el tiempo de análisis de semanas a horas, permitiendo resultados rápidos y eficientes.
- ❖ **Análisis paralelo de muestras:** Procesa diferentes muestras de ADN **en paralelo**, optimizando tiempos.



Tipos de Paralelismo

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

Reflexión

Presentación de
prácticas

Tipos de Paralelismo

- **A nivel de datos:** La misma operación se aplica a diferentes datos simultáneamente.
- **A nivel de instrucciones:** Varias instrucciones se ejecutan simultáneamente dentro de una CPU.
- **A nivel de tareas:** Diferentes tareas independientes se ejecutan en paralelo.

Tipos de Paralelismo

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

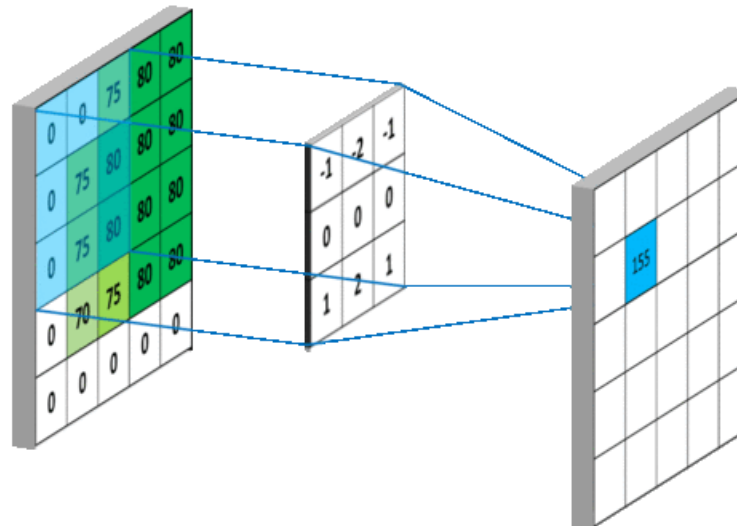
Reflexión

Presentación de prácticas

Tipos de Paralelismo

A nivel de datos:

- **Definición:** Se procesan múltiples datos simultáneamente con la misma operación.
- **Ejemplo:** Aplicar un filtro a una imagen procesando varios píxeles en paralelo (filtro mediante convolución).
- **Uso:** GPUs y arquitecturas SIMD (Single Instruction, Multiple Data).



Tipos de Paralelismo

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

Reflexión

Presentación de
prácticas

Tipos de Paralelismo

A nivel de instrucciones:

- **Pipelining:** Divide la ejecución de instrucciones en múltiples etapas (como búsqueda, decodificación y ejecución).
- **Ejecución Fuera de Orden:** Permite a las CPUs reordenar la ejecución de instrucciones para optimizar el uso de las unidades funcionales.
- **Superescalaridad:** Uso de múltiples unidades de ejecución dentro de un procesador para ejecutar varias instrucciones en un solo ciclo de reloj.

Instruction	1				2			
Fetch								
Decode								
Execute								
Write								
Clock	1	2	3	4	5	6	7	8

Non-Pipelined

Instruction	1	2			
Fetch					
Decode					
Execute					
Write					
Clock	1	2	3	4	5

Pipelined

Tipos de Paralelismo

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

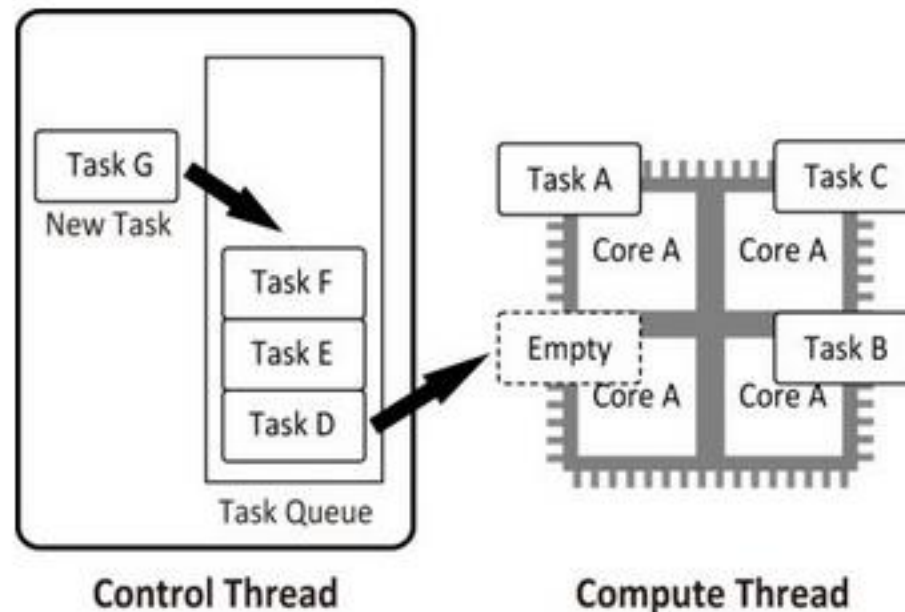
Reflexión

Presentación de
prácticas

Tipos de Paralelismo

A nivel de Tareas:

• **Definición:** Divide un problema grande en múltiples tareas independientes que pueden ejecutarse simultáneamente.



Ley de Amdahl

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

Reflexión

Presentación de
prácticas

Limitaciones en la paralelización

- El beneficio del paralelismo está **limitado** por la **fracción del programa que no puede ejecutarse en paralelo**. Su ecuación se expresa como:

$$S(N) = \frac{1}{(1 - P) + \frac{P}{N}}$$

- Donde:**
- S(N) es la aceleración obtenida con N procesadores.
- P es la fracción del código que puede ejecutarse en paralelo.
- N es el número de unidades de procesamiento.

Ley de Amdahl

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

Reflexión

Presentación de
prácticas

Limitaciones en la paralelización

Ejemplo 1: Código con Fracción Secuencial Significativa:

- Supongamos que un programa tiene un **70%** de su código **paralelizable** ($P = 0.7$) y el **30% restante** debe ejecutarse de forma **secuencial**.
- Si usamos 4 procesadores ($N = 4$), aplicamos la ecuación:

$$S(4) = \frac{1}{(1 - 0.7) + \frac{0.7}{4}}$$

$$S(4) = \frac{1}{0.3 + 0.175} = \frac{1}{0.475} \approx 2.11$$

- Esto significa que, a pesar de cuadruplicar los procesadores, la aceleración solo es 2.11 veces mayor, muy por debajo del ideal de 4x.

Ley de Amdahl

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

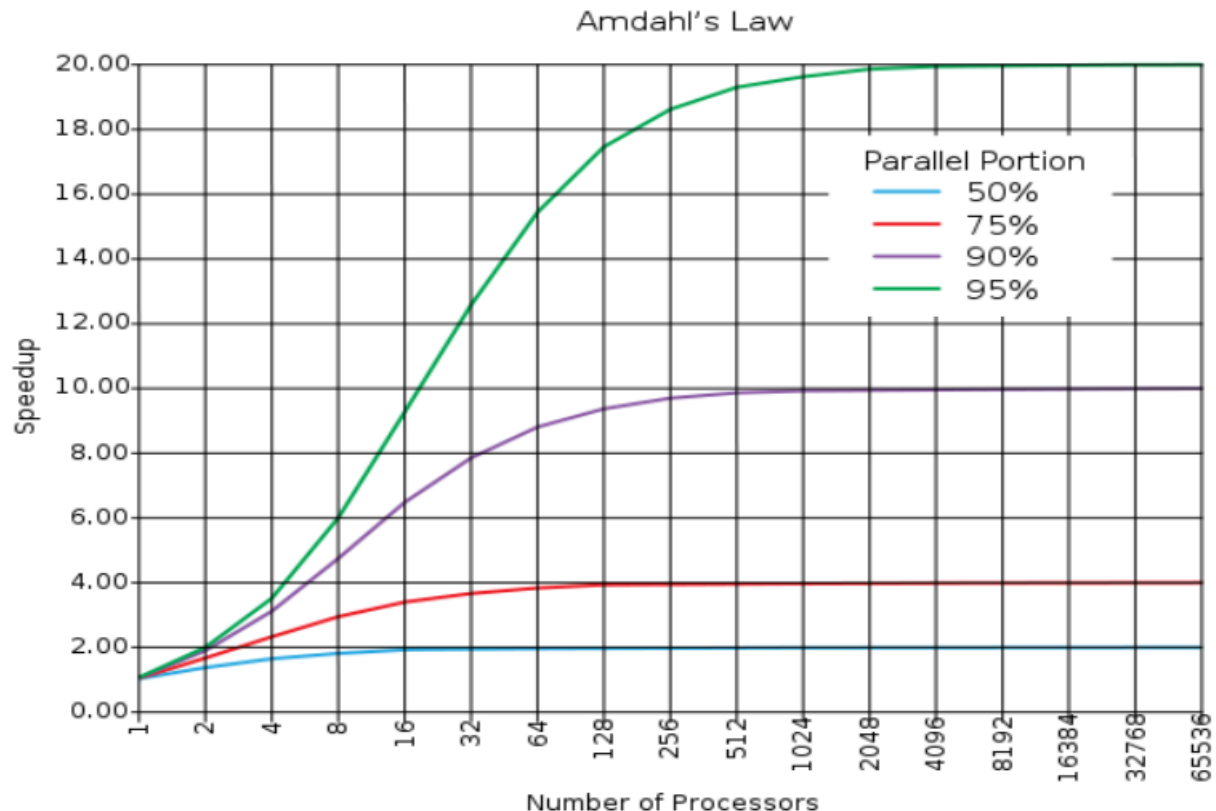
Limitaciones

Reflexión

Presentación de
prácticas

Limitaciones en la paralelización

Esta ley ayuda a definir si introducir una mejora en el sistema vale o no la pena



Ley de Gustafson

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

Reflexión

Presentación de
prácticas

Limitaciones en la paralelización

Un Enfoque Alternativo

- La Ley de Gustafson sugiere que el paralelismo es más efectivo en problemas escalables.
- Conforme aumenta el tamaño del problema, la fracción paralelizable también crece. Su ecuación se expresa como:

$$S(N) = N - \alpha (N - 1)$$

- **Donde:**
- **$S(N)$** es la aceleración obtenida con N procesadores.
- **α** es la fracción secuencial del código.
- **N** es el número de procesadores.

Ley de Gustafson

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

Reflexión

Presentación de
prácticas

Limitaciones en la paralelización

Un Enfoque Alternativo

- **Ejemplo 2:** Ley de Gustafson
- Supongamos que un problema tiene una **fracción secuencial del 5% ($\alpha = 0.05$)**. Si usamos 16 procesadores (**$N = 16$**):

$$S(16) = 16 - 0.05 (16 - 1)$$

$$S(16) = 16 - 0.05 (15)$$

$$S(16) = 16 - 0.75$$

$$S(16) = 15.25$$

- Este resultado indica que en lugar de la aceleración de 9.17 veces obtenida con la Ley de Amdahl la Ley de Gustafson sugiere una aceleración de **15.25 veces**, lo que refleja un **mejor aprovechamiento del paralelismo en problemas que pueden escalar** con la cantidad de procesadores.

Ley de Gustafson

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

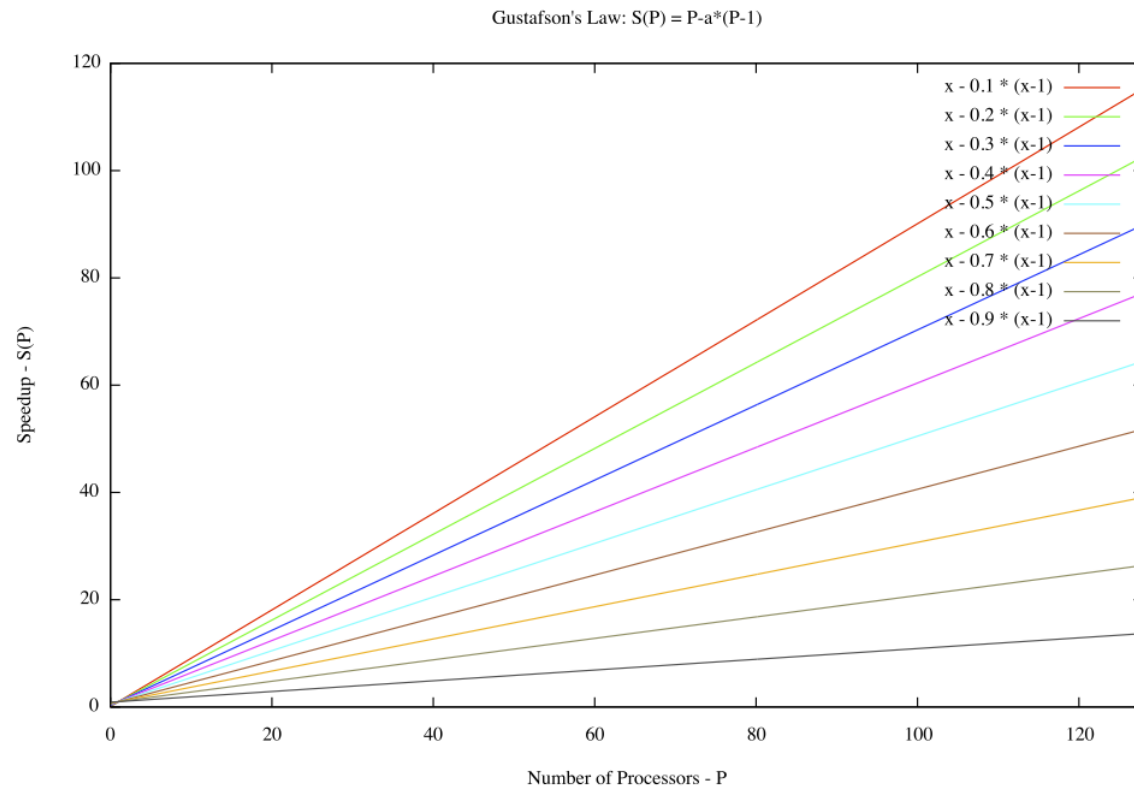
Limitaciones

Reflexión

Presentación de
prácticas

Limitaciones en la paralelización

Esta ley ayuda a definir si introducir una mejora en el sistema vale o no la pena



Amdhal Vs. Gustafson

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

Reflexión

Presentación de
prácticas

Limitaciones en la paralelización

Comparación práctica: Amdahl vs. Gustafson

- **Amdahl** nos **indica si vale la pena optimizar** el código existente, mientras que **Gustafson** nos muestra **cómo escalar la aplicación** para aprovechar más recursos.

Situación práctica	¿Qué ley usar?	Explicación
Optimización en sistemas pequeños	Ley de Amdahl	El problema es fijo y necesitas evaluar si agregar más procesadores mejorará significativamente el rendimiento.
Desarrollo de simulaciones escalables	Ley de Gustafson	El problema puede crecer (más datos, más complejidad) y necesitas aprovechar más procesadores.
Diagnóstico del rendimiento actual	Ley de Amdahl	Para identificar cuánta parte del programa no es paralelizable y cómo limita la aceleración.
Diseño de nuevas aplicaciones	Ley de Gustafson	Cuando planificas un sistema que puede aprovechar recursos adicionales en el futuro.

Limitaciones

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

Reflexión

Presentación de
prácticas

Limitaciones en la paralelización (cuellos de botella)

- **Costes de comunicación:** La transferencia de datos entre procesadores puede ralentizar el rendimiento.
- **Conflictos de acceso a memoria compartida:** Varios procesadores accediendo al mismo dato pueden generar esperas.
- **Sincronización:** Necesidad de coordinar los procesadores para evitar errores

Limitaciones

Objetivos

Introducción

Comparativa

Motivación

Aplicaciones

Tipos

Amdahl

Gustafson

Limitaciones

Reflexión

Presentación de
prácticas

Reflexión y Práctica: A continuación realizaremos dos ejercicios prácticos para reflexionar sobre:

- **La comparación entre la Ley de Amdahl y la Ley de Gustafson.**
- **La aplicación del paralelismo en la optimización de procesos reales.**