



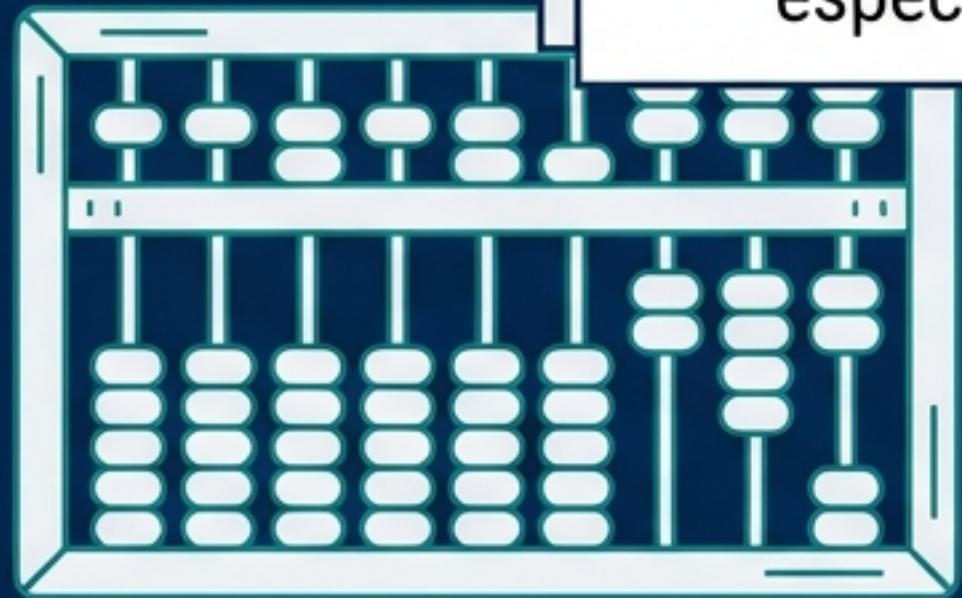
Algoritmia y Complejidad Computacional

Fundamentos para la Optimización Heurística de Estructuras

Máster Universitario de Ingeniería del Hormigón
Universitat Politècnica de València
Autor: Prof. Dr. Ing. V. Yepes

¿Qué es exactamente un algoritmo?

Es un **conjunto** prescrito de reglas o instrucciones bien definidas para la resolución de un **problema** específico en un **número finito de pasos**.



Sentido Antiguo:

Reglas para realizar operaciones algebraicas (métodos de cálculo).

Sentido Moderno:

Una 'receta' o método para resolver cualquier de problema donde se manipula información mediante operaciones elementales.

Condiciones necesarias para la validez



Precisión

Cada instrucción debe ser exacta y sin ambigüedad.

Finitud

Debe terminar siempre tras un número finito de pasos.

Efectividad

Ejecutable por persona o máquina en tiempo finito.

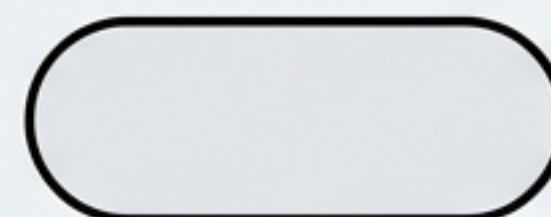
Entrada

Datos iniciales claramente definidos.

Salida

Un resultado producido por los datos de entrada.

Simbología estándar de flujo



Símbolo terminal (Comienzo / Fin)



Entrada / Salida (Input / Output)



Procesamiento de datos (Ejecución de instrucción)



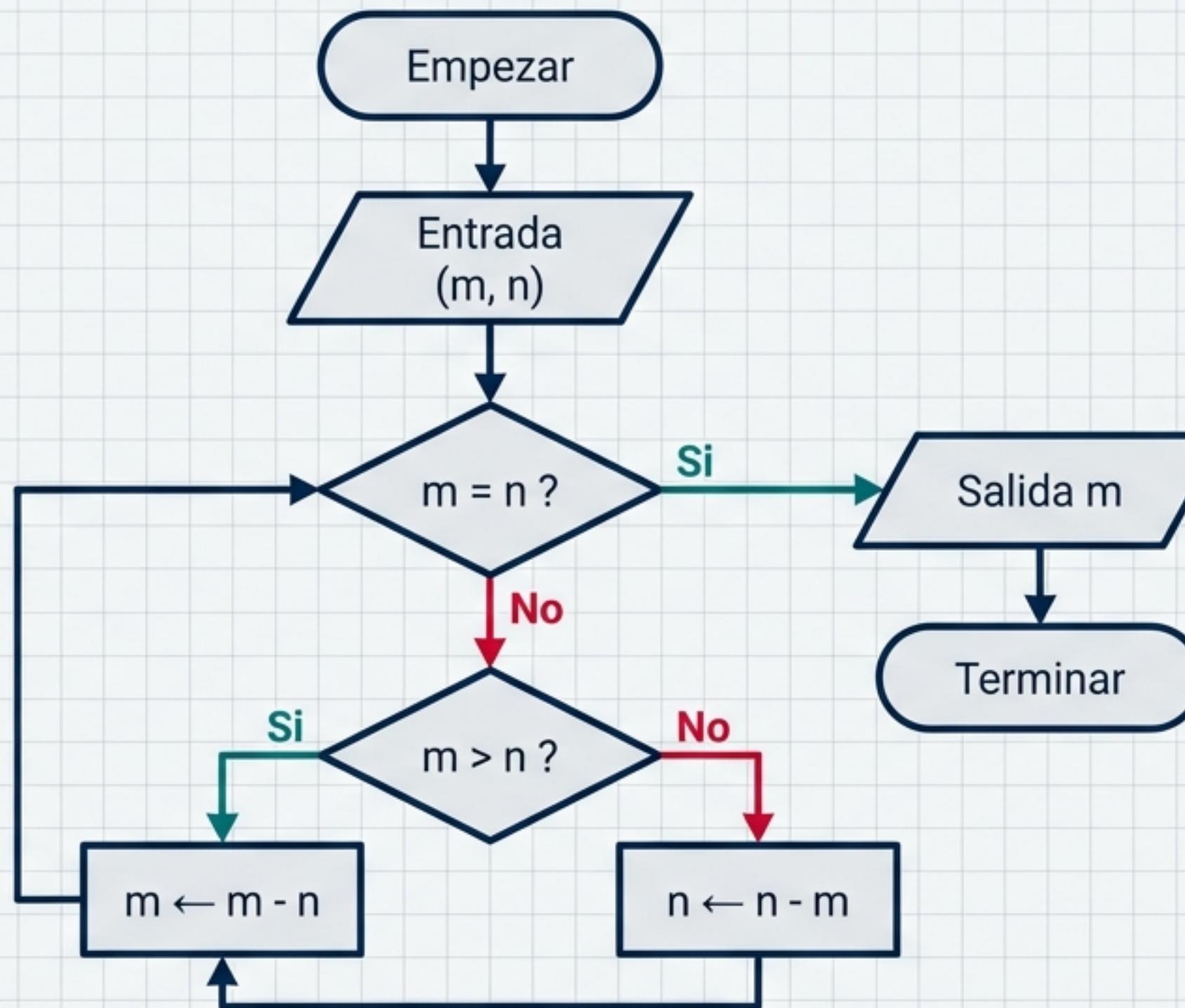
Decisión (Bifurcación lógica)



Conector (Punto de unión de caminos)

Caso de Estudio: El Algoritmo de Euclides

Cálculo del Máximo Común Divisor (MCD) entre enteros positivos m y n



Rendimiento Algorítmico y Coste

Rendimiento

Medida del tiempo medio de ejecución empleado para completar la operación con un conjunto de datos.

Análisis Asintótico

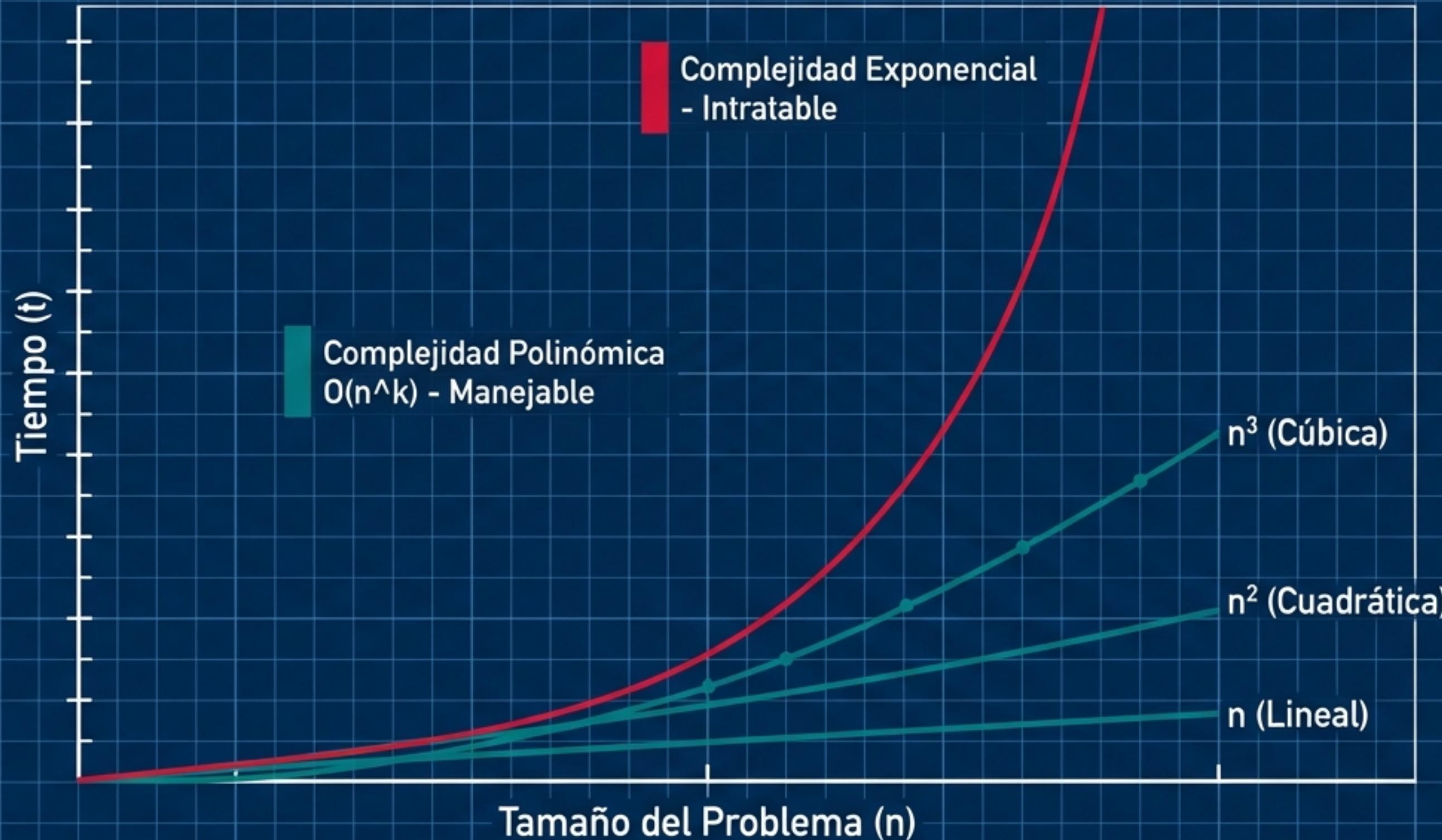
Relación entre el esfuerzo de cálculo y la dimensión del problema (n).

El Peor Caso

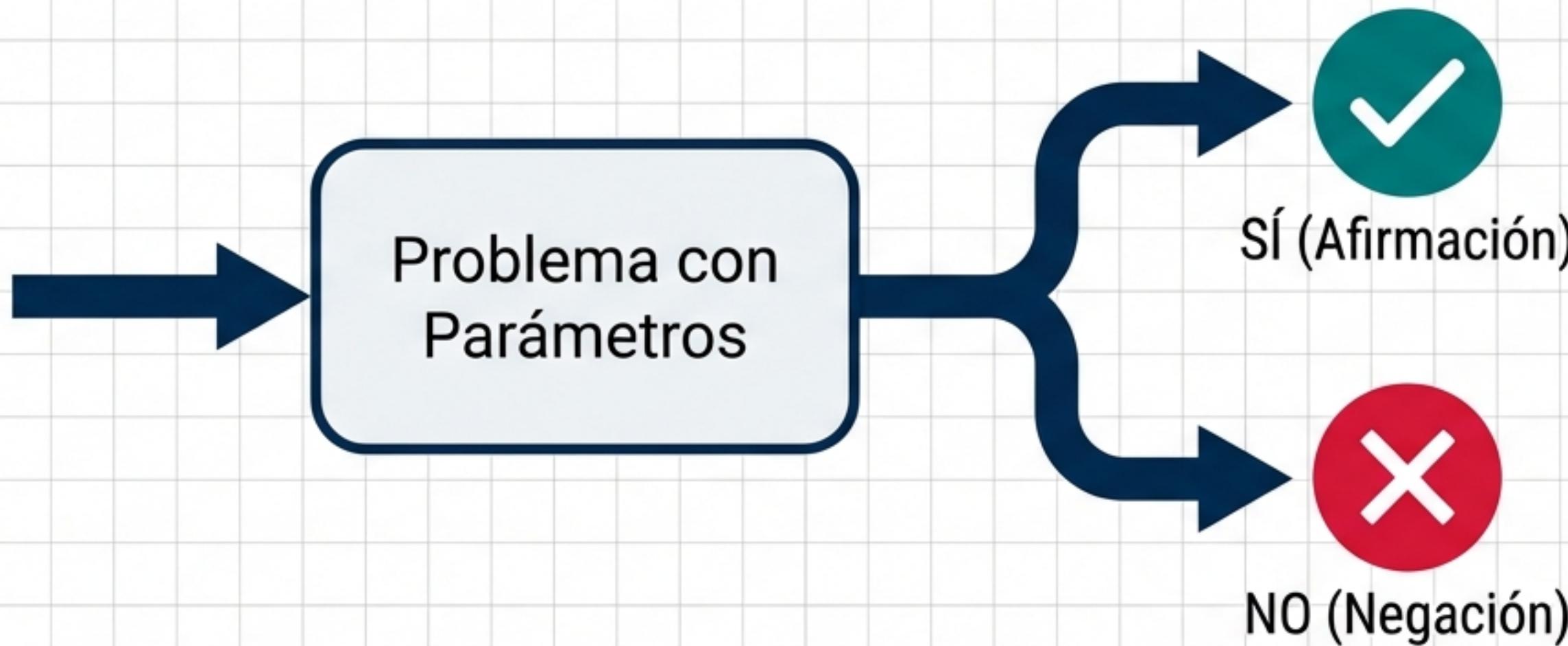
El coste máximo posible sobre todos los problemas de talla n .



La Barrera del Tiempo: Polinómica vs. Exponencial



Problemas de Decisión



Una proposición dirigida a averiguar un resultado que puede ser contestado con una afirmación o una negación.

Al especificar los valores, tenemos un caso concreto.

Clasificación de Dificultad: P vs. NP



Clase P (Polinómica)

Problemas que pueden ser RESUELTOS en tiempo polinomial.

Ejemplos: Búsqueda binaria, Multiplicación matricial.



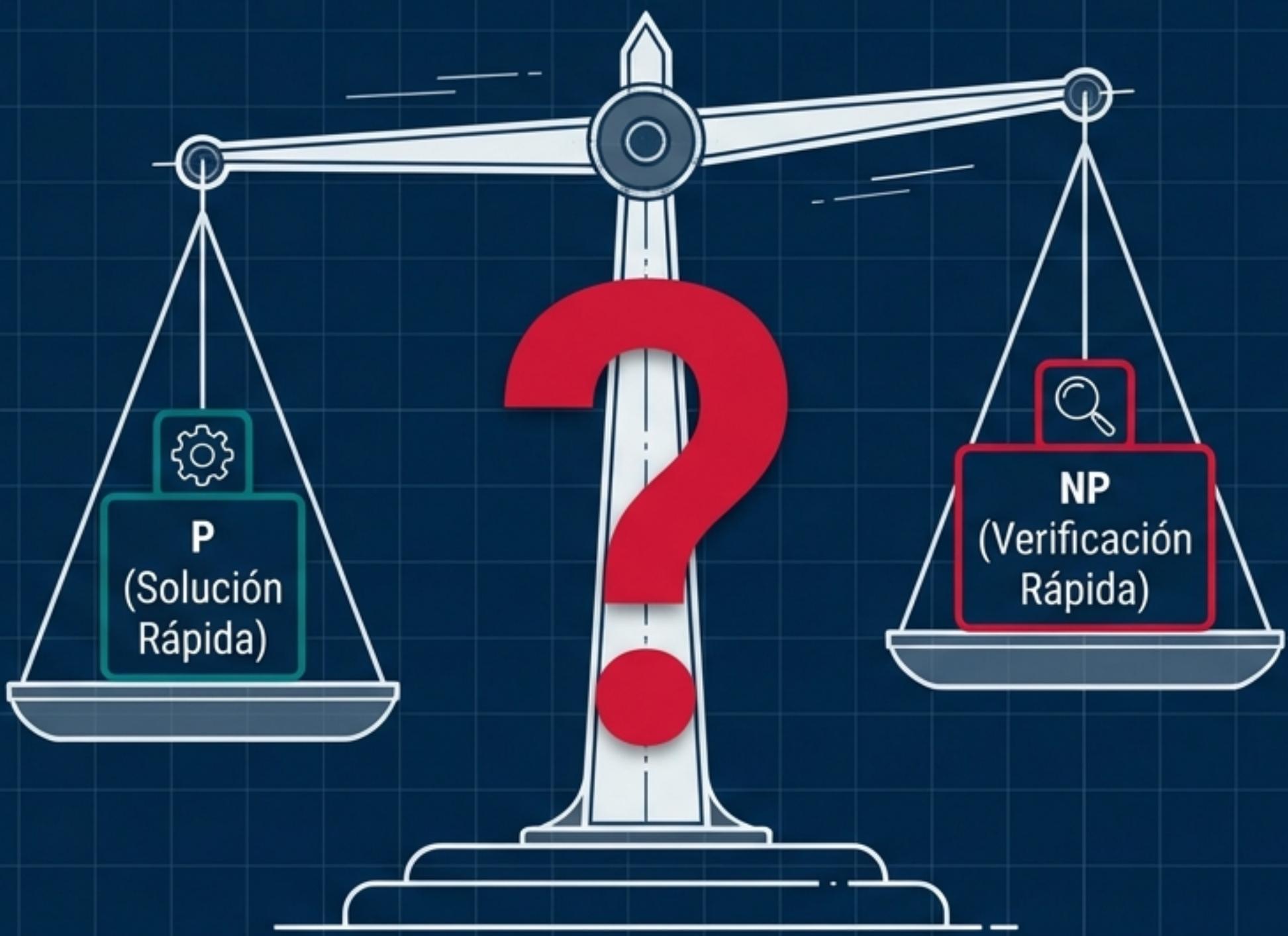
Clase NP (Non-deterministic Polynomial)

Problemas que pueden ser VERIFICADOS en tiempo polinomial.

No conocemos un procedimiento rápido para resolverlos, pero si nos dan la solución, podemos comprobarla rápido.

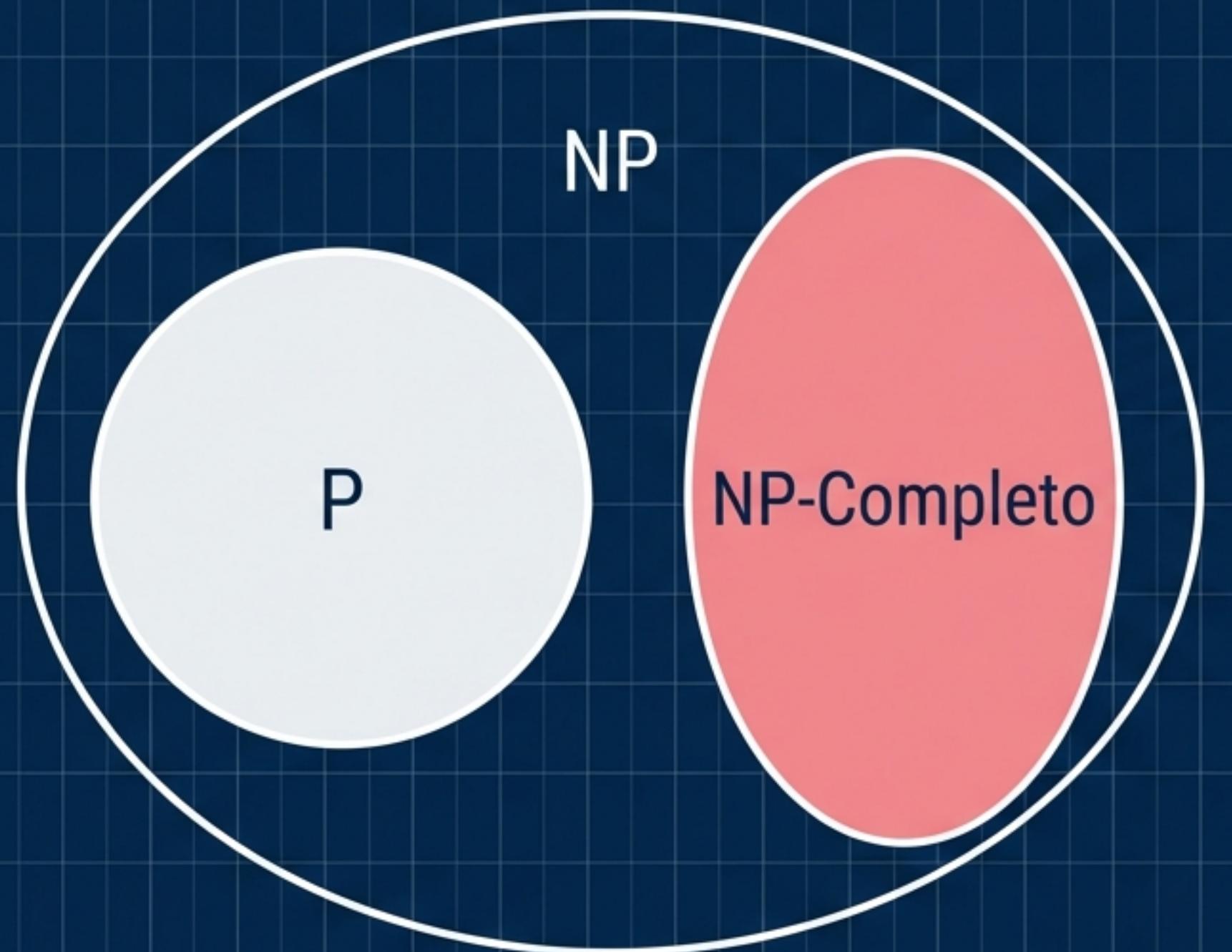
$P \subseteq NP$ (Todo problema fácil de resolver es fácil de verificar).

La Pregunta del Millón: ¿Es $P = NP$?



- ¿Implica una verificación rápida que existe una solución rápida?
- Pregunta abierta desde 1971.
- Consenso actual: ¡Hoy en día no se ha podido demostrar que $P=NP$!

NP-Completo (NPC)

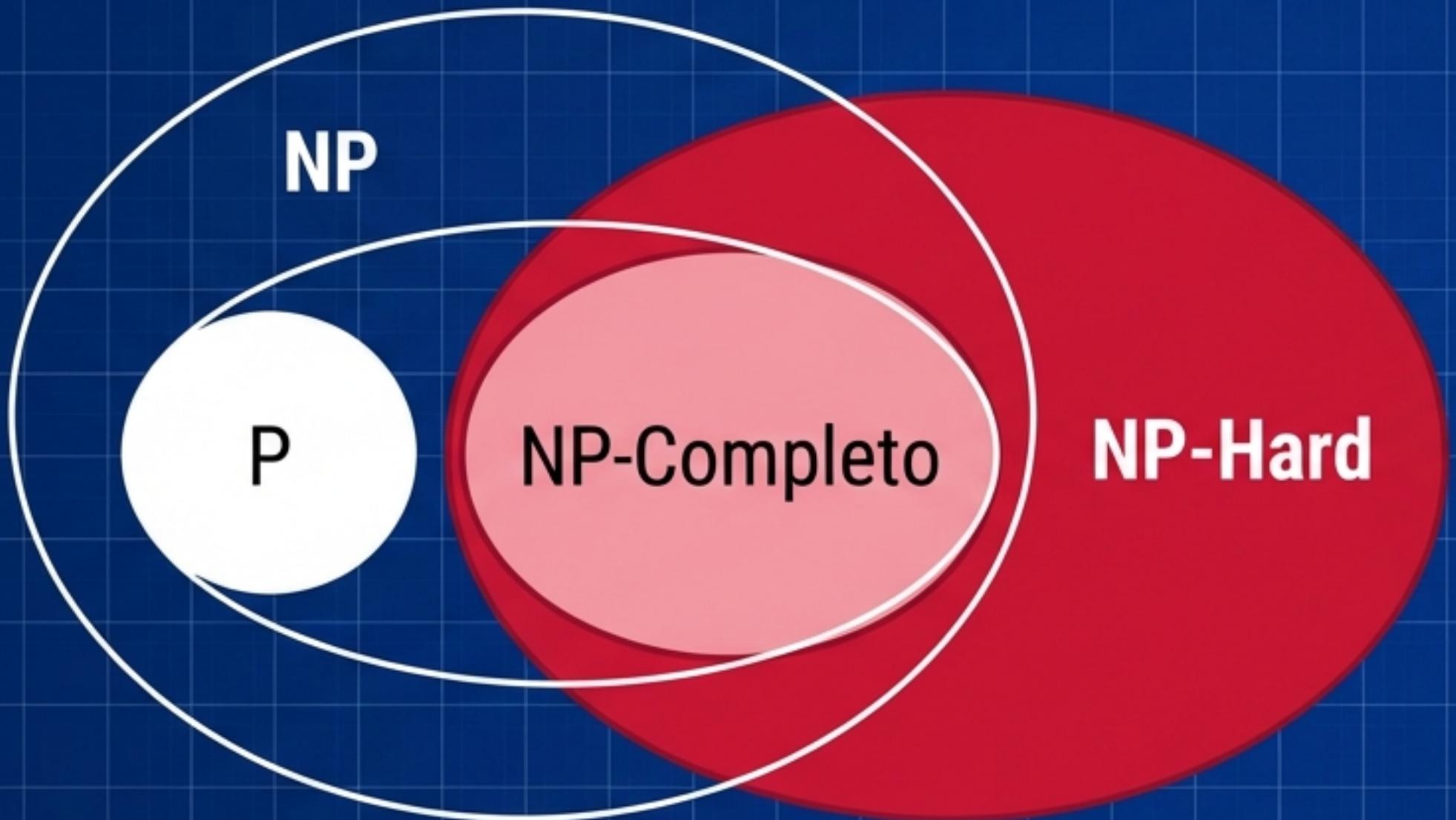


- **Definición:** Un problema es NPC si cualquier problema en NP se puede transformar en él en tiempo polinomial.

Implicación: Si un solo problema NPC se resuelve eficientemente, **TODOS los problemas NP** se resuelven.

Ejemplo: Problema del viajante de comercio.

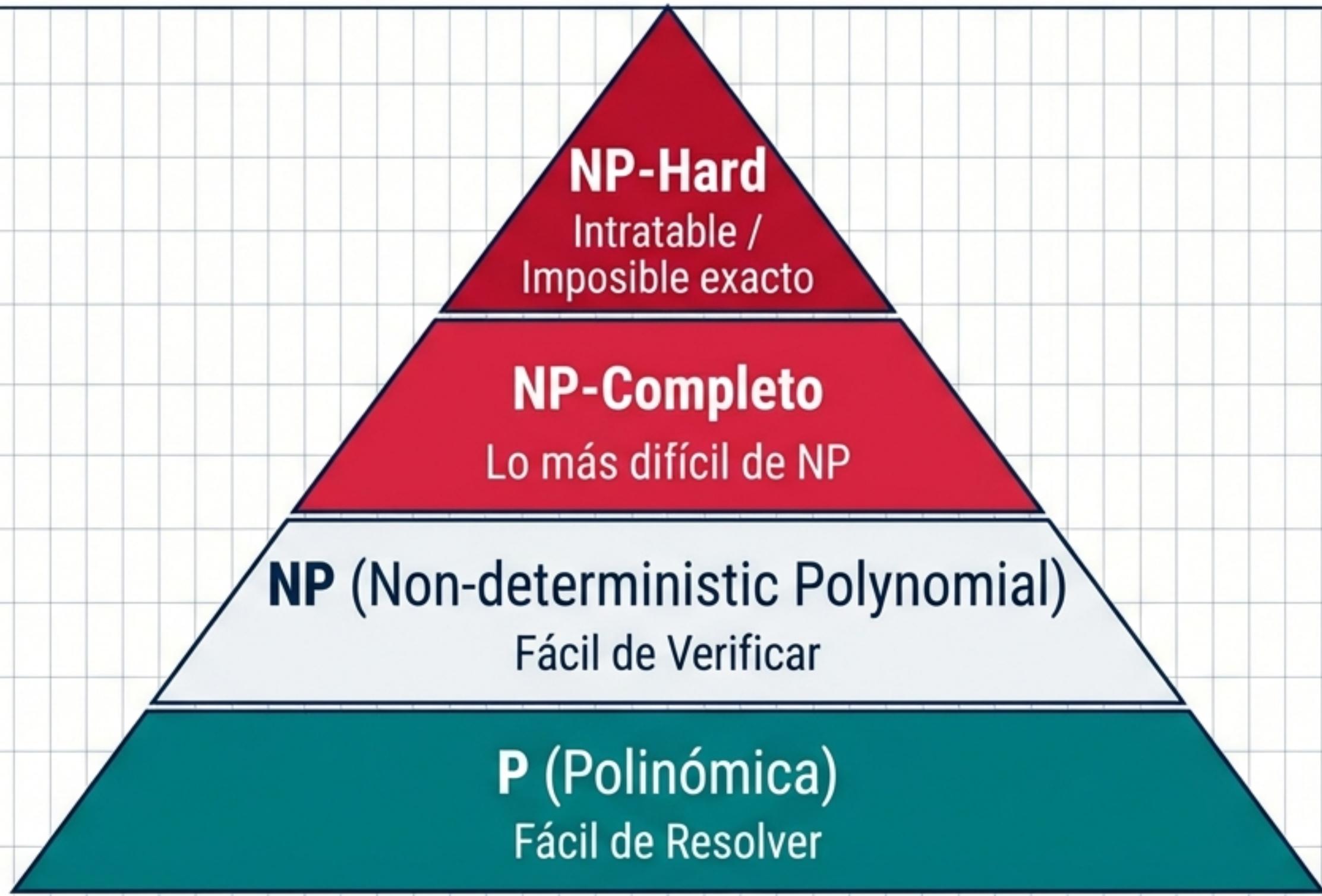
NP-Hard (NP-Duro)



⌚ Problemas que son “como mínimo tan difíciles” como los NP-Complejos.

- A efectos prácticos, son imposibles de resolver de forma exacta para tamaños grandes.
- Nota: No tienen por qué pertenecer a la clase NP.

Resumen: Jerarquía de Complejidad



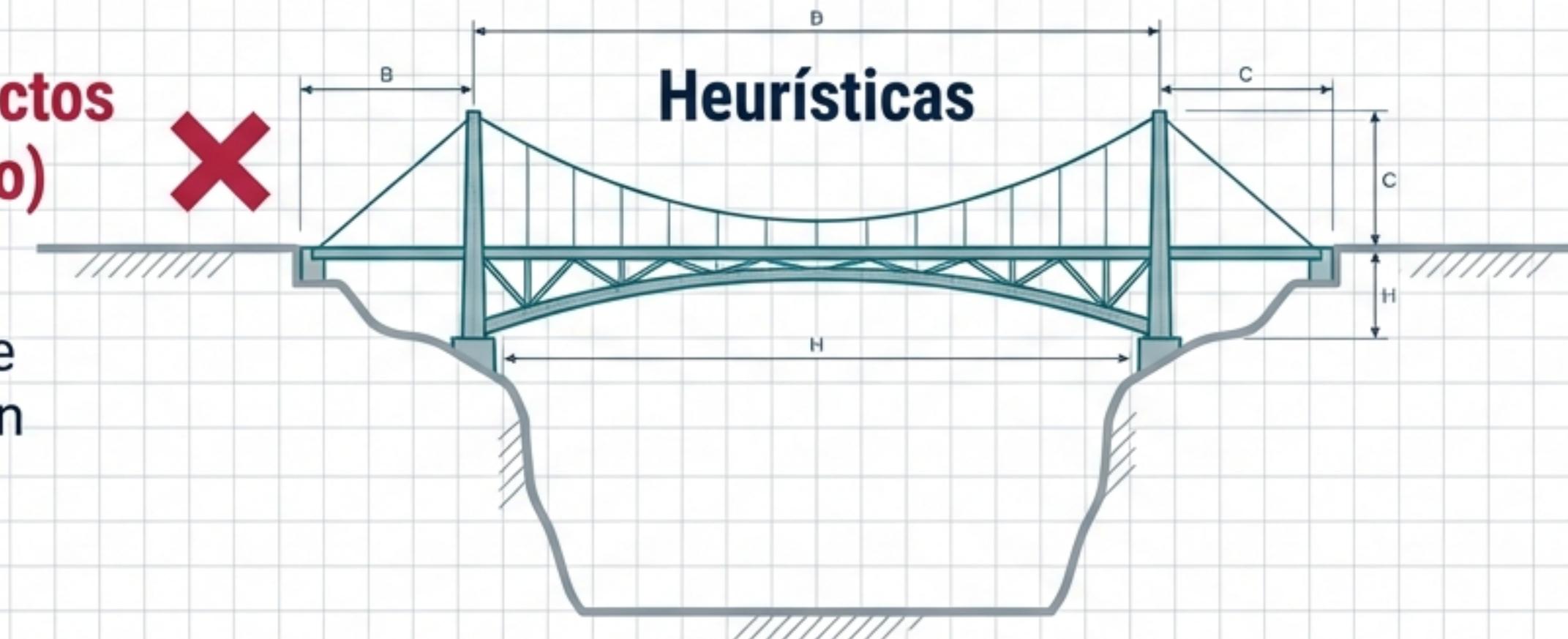
La Necesidad de una Nueva Vía

**Métodos Exactos
(Bloqueado)**

Muchos problemas de ingeniería son NP-Hard.



Heurísticas



**Solución
Práctica**

Resolución exacta inviable.

**Por tanto... necesitamos Algoritmos
Aproximados (Heurísticas)**

Referencias y Créditos



Material basado en el curso:
**Modelos predictivos y de optimización de
estructuras de hormigón**

Prof. Dr. Ing. V. Yepes

Departamento de Ingeniería de la Construcción y PIC
Universitat Politècnica de València