

Asociación estudiantil de ingeniería física(AEIF)

Pagina web



Divulgación

---

El problema de P vs NP.

---

Por Iván Reyna 02/12/2023

La fundación Clay de matemáticas en el año 2000 fundó “Los 7 problemas del milenio”, los cuales tienen una recompensa de un millón de dólares para la persona que logre resolver un problema. En los que se encuentran: El problema de P vs NP, La conjetura de Birch y Swinnerton-Dyer, Yang-Mills y el salto de masa, Las ecuaciones de Navier-Stokes, La conjetura de Hodge, La hipótesis de Riemann y la Conjetura de Poincaré.

En este artículo se profundizará el problema de P igual a NP. Problema del milenio el cual tiene el propósito de la creación de algoritmos computacionales, cuya solución sea rápida al igual que su comprobación.

## El problema de P vs NP como problema del milenio.

La interrogante inicial acerca de la relación entre las clases de complejidad NP y P fue planteada por el científico computacional Stephen Cook. Hasta la fecha, la teoría de la complejidad computacional no ha logrado proporcionar una respuesta definitiva. En esencia, la cuestión central, formulada como <<¿es P = NP completo?>>, indaga si la capacidad de “verificar” soluciones de manera rápida (característica de problemas NP) implica necesariamente la habilidad de “obtener” respuestas con la misma rapidez (atributo de problemas P). En este contexto, “rápidamente” significa “Tiempo polinómico”.

Encontrar una solución a este problema planteado hace 50 años podría desbloquear innumerables retos computacionales o demostrar que están fuera de nuestro alcance para siempre. Analizamos su historia y su enorme impacto, que lo ha convertido en uno de los siete “Problemas del Milenio”.

Shiobhan Roberts.

La informática se encuentra ante la complejidad de diversos desafíos que marcan su evolución. Entre estos desafíos, destacan:

- **Almacenamiento:** Con el crecimiento exponencial de los datos, la gestión efectiva de grandes conjuntos de datos, su almacenamiento, procesamiento y análisis se ha convertido en un desafío clave.
- **Tiempo:** A través de una aproximación al número de pasos de ejecución que un algoritmo utiliza para resolver un problema.

## Problemas del tipo P.

Conjunto de problemas que un ordenador puede resolver en un tiempo razonable. En particular, los problemas de la clase P son aquellos cuyo tiempo para encontrar una solución puede expresarse mediante una fórmula polinómica, como  $n^2$ . En algoritmos de tiempo polinómico, donde  $n$  es el tamaño del problema o del input, el abordaje de este problema ocurre a una tasa de tiempo razonable.

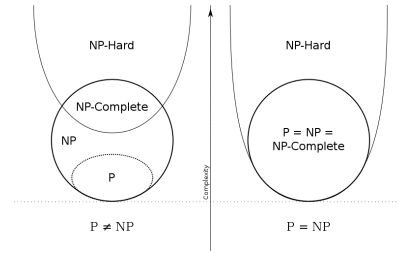
Podemos expresar lo anterior como:  $P(x) = ax^2 + bx + c$ , siendo así un tipo de problema que no supera un polinomio de grado 2.

Algunos ejemplos son:

- Ordenar una lista.
- Buscar un elemento de un texto.
- Multiplicación de números enteros.

## Problemas del tipo NP.

Conjunto de problemas complejos y tardíos. En este caso nos referimos que son problemas de forma exponencial  $n^x$ , donde  $x$  es cualquier valor mayor que 2; más sin embargo, la comprobación del problema se puede linealizar de forma  $x \leq 2$ . Es decir un problema del tipo  $x \in R|x \geq 2$ , con comprobación  $x \in R|x \leq 2$ .



Algunos ejemplos son:

Sudoku.

Rompecabezas.

Problema del viajero.

Visto desde la maquina de Turing.

Sea  $L$  un lenguaje definido sobre un alfabeto finito,  $\Sigma$ . Si existe una relación binaria donde  $R \subset \Sigma^* \times \Sigma^*$  y un entero positivo  $k$ , tal que para todo  $x \in \Sigma^*$ , se satisfacen las siguientes condiciones:

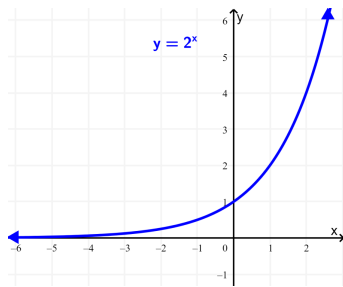
$$x \in L \leftrightarrow \exists y \in \Sigma^* \text{ tal que } (x,y) \in R \text{ y } |y| \in O(|x|^k).$$

$LR = \{x\#y : (x,y) \in R\}$  en  $\Sigma \cup \{\#\}$ , donde es decidible por una maquina de Turing en tiempo polinomial.

Entonces, la máquina de Turing que decide  $L_R$  es llamada el Verificador para  $L$  y es llamado el Certificado de membresía de  $x$  en  $L$ . Finalmente,  $L$  se encuentra en  $NP$  "si y solo si"  $L_R$  corre en tiempo polinómico.

## ¿P=NP?

Entonces solo existen dos caminos  $P=NP$  o  $P \neq NP$ .



Para  $P \neq NP$ , se representa como:  $P \cap NP$ . Donde si  $P \neq NP$ , entonces queda demostrado que la mayoría de problemas se tendrían que resolver de forma tardía.

Para  $P=NP$ , se representa como:  $P \subset NP$ . Donde si  $P=NP$ , entonces se tiene un problema de resolución rápida y efectiva, y a su vez, una comprobación de la misma forma.

Comprobar dichas propiedades es lo que no han logrado demostrar los matemáticos ni los grandes programadores, pero ya se han tenido grandes avances de dicho estudio.

## Teorema de Cook-Levin y el SAT.

El teorema de Cook-Levin propuesto en 1971. Establece que un problema de decisión, se puede reducir a un problema de satisfacción booleana (SAT). Esto quiere decir que problemas de decisión se pueden resolver mediante este sistema.

Considerando un problema NP-completo, podemos reducir la complejidad del mismo mediante el SAT. Dando mayor facilidad para la resolución del problema. A pesar de que reduce el problema, "no logra resolverlo por completo".