



Universidad
Carlos III de Madrid
www.uc3m.es

La Máquina de Turing como precusora de la Teoría de la Computación (III)

M^a Araceli Sanchis de Miguel

Grupo de Control y Aprendizaje de Sistemas



Este obra está bajo una
[licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 3.0 España.](https://creativecommons.org/licenses/by-nc-sa/3.0/es/)



Motivación e interés

- ¿Puede tener mente un computador?
- ¿Existe algún algoritmo que describa el funcionamiento del cerebro humano? \Rightarrow la IA lo busca.
- ¿Por qué no se ha encontrado aún?
 - Límites tecnológicos (temporales)
 - Forma de razonar es No-algoritmizable
- ¿Dónde están, si existen, los límites de la computación?





MT Universal (I)

- Es una MT “programable”
 - Dependiendo del programa, puede simular a cualquier otra MT.
 - Lee y ejecuta programas almacenados en su cinta
 - El programa de una MTU es una versión codificada de una MT que lleva a cabo la tarea que se desea que ejecute la MTU.
- Para construir una MTU que desempeñe una tarea:
 - Hay que diseñar una MT genérica para esa tarea y
 - codificar dicha MT genérica en la cinta de la MTU.
 - Se tratará de una MT con alfabeto binario y cinta limitada en un sentido.



codificar la cadena de entrada.

[licencia de Creative Commons Reconocimiento-NoComercial-CompartirIgual 3.0 España.](https://creativecommons.org/licenses/by-nc-sa/3.0/es/)



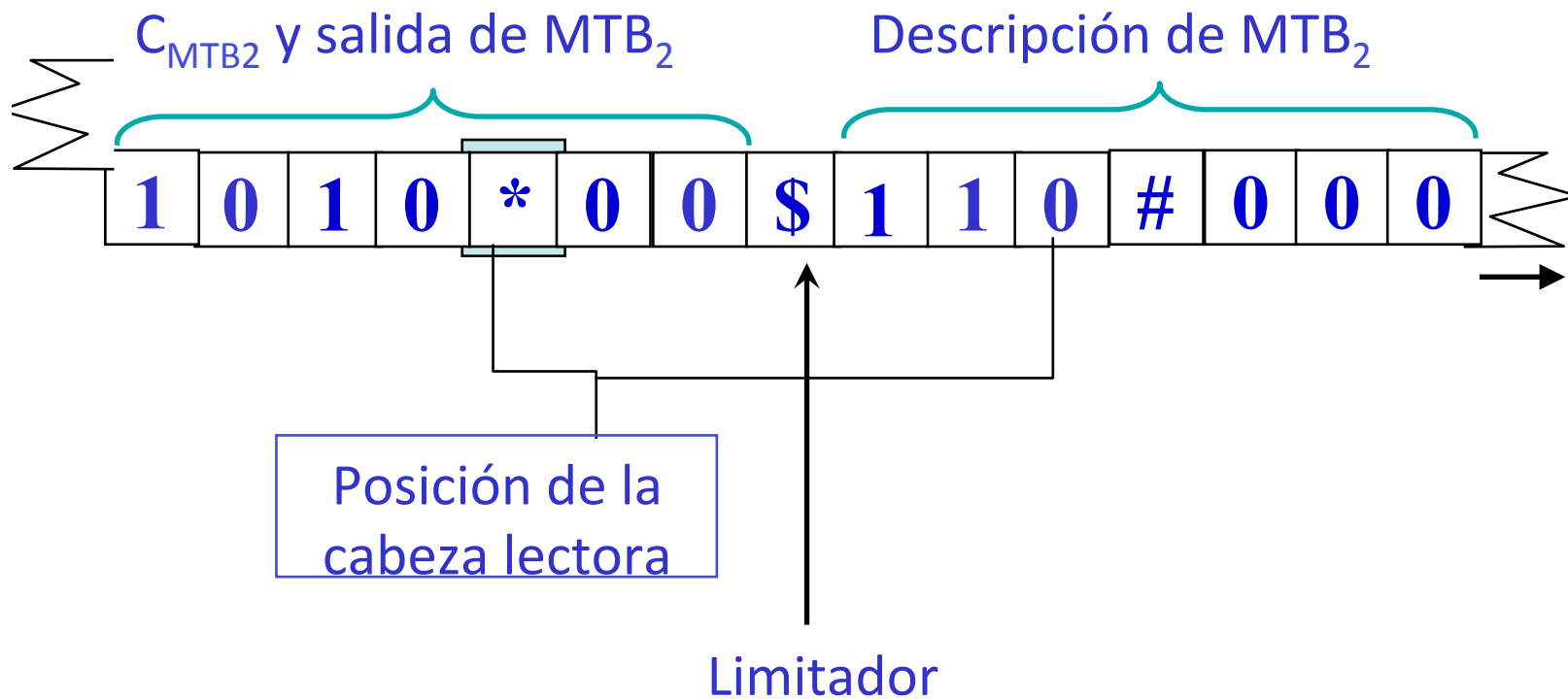
MT Universal (II)

- Recibe como entrada, codificada en la cinta, la descripción completa de otra MTB2.
- Ejecuta la MTB2 que tiene en su cinta (simula el comportamiento de MTB2).
- En la cinta de MTU está MTB2 y la transcripción de la cinta de MTB2, CMTB2
- Al final, en la parte donde estaba CMTB2 aparece la salida que produciría MTB2 actuando sobre CMTB2 .





MT Universal





MT Universal. Cinta

- Información sobre:
 - función de transición
 - estado inicial
 - posición inicial de la cabeza
 - contenido inicial de la cinta
- Dividida en dos partes por el \$:
 - izda del \$: CMTB2
 - dcha del \$: codificación de MTB2
- Cabeza lectora: ¿tiene que leer en ambos lados del \$?
 - Donde está la cabeza: *
 - contenido reemplazado por *: en la posición “l” a la dcha del \$.
- A la dcha del \$, estructura análoga a MT Transcriptor de Información , con:
 - longitud de la dirección y etiquetas, “l” = $E(\log_2 |Q|) + 1$
 - longitud de los registros, “m” = $2 \times l + 1$





MT y Computabilidad

- Turing demostró con su MT (y sus extensiones) que todo problema computable (resoluble), lo es en una MTU.
- Complejidad computacional:
 - Se basa en tratar de dar respuesta a la siguiente pregunta:
 - *¿Qué hace a algunos problemas computacionalmente difíciles y a otros sencillos?*
 - Estudia el orden de complejidad de un algoritmo que resuelve un problema *decidible*.
 - Para ello, considera los 2 tipos de recursos requeridos durante el cómputo para resolver un problema:
 - **Tiempo:** Número de pasos base de ejecución de un algoritmo para resolver un problema.
 - **Espacio:** Cantidad de memoria utilizada para resolver un problema.





Funciones NO computables (I)

- ¿Cuáles son los problemas/funciones no computables?
- Equivalente a los “no-decidibles”:
 - **Función de Rado:**
 - **Secuencias aleatorias**
 - **El teorema de Fermat:** $a^n + b^n = c^n$ / $a, b, c \neq 0$ y n entero > 2
 - **Los primos pares:**





Funciones NO computables (II)

- ¿Cuáles son los problemas/funciones no computables?
- Equivalente a los “no-decidibles”:
 - **Función de Rado**: El Problema del Castor Afanoso: ¿Cuál es el número máximo de 1's que pueden ser escritos por una máquina de Turing de N estados (donde N no incluye el estado final) que termina en parada, y que comienza con una cinta inicialmente en blanco? Este número, que varía en función del número de estados de la máquina, se denota $\sigma(N)$. Una máquina que produce $\sigma(N)$ celdas no en blanco se denomina Castor Afanoso.
 - **Secuencias aleatorias**
 - **El teorema de Fermat**
 - **Los primos pares**





Funciones NO computables (III)

– Función de Rado:

Num. estados	Num max. 1's impresos	Cota inferior para el valor de σ
3	$\sigma(3)$	6
4	$\sigma(4)$	12
5	$\sigma(5)$	17
6	$\sigma(6)$	35
7	$\sigma(7)$	22.961
8	$\sigma(8)$	3^{92}
9	$\sigma(9)$	3^{92+1}
10	$\sigma(10)$	$((a^a)^a)\dots a = a^a$





Funciones NO computables (V)

- ¿Cuáles son los problemas/funciones no computables?
- Diferentes de los “no-decidibles”:
 - **Función de Rado:**
 - **Secuencias aleatorias**
 - **El teorema de Fermat:** $a^n + b^n = c^n$ / $a, b, c \neq 0$ y n entero > 2
 - **Los primos pares:**





Funciones NO computables (VI)

- ¿Cuáles son los problemas/funciones no computables?
- Diferentes de los “no-decidibles”:
 - **Función de Rado:**
 - **Secuencias aleatorias**
 - **El teorema de Fermat:**
 - **Los primos pares:** Existe un número infinito de primos p tales que $p + 2$ también es primo. Ejs: 3 y 5 son primos pares, 11 y 13 ó 29 y 31. según los primos son más grandes la frecuencia de aparición de pares va disminuyendo, pero siguen surgiendo pares de primos gemelos aun entre números de tamaños enormes.





El problema de la Parada (I)

NO existe una Máquina de Turing que pueda decidir si una Máquina de Turing se va a parar: problema de la parada.

- Prueba por reducción al absurdo:
 - Supongamos que si existe esa MT y llamémosla **A**
 - **A** tiene codificado en su cinta (parecido a como ocurre en MTU) una MT (**MT_p**) y su cinta (**C**). A la salida hará:

$$A(MT_p, C) = \begin{cases} 1 & \text{si } MT_p \text{ con } C \text{ se para} \\ 0 & \text{si } MT_p \text{ con } C \text{ no se para} \end{cases}$$

- La MT **A** cuando escribe un 1, entra en un bucle ∞ (no se para)

En torno a la figura de Alan Turing





El problema de la Parada (II)

- MT_p : es la p -ésima MT
 - La cinta C puede ser binaria, de forma que C se corresponderá con un número binario (q).
- Supongamos que escribimos una MT_1 que:
 - Recibe el número en binario k ,
 - lo duplica y
 - aplica al resultado $A(MT_k, k)$.
- Dicha máquina será una MT determinada, por ejemplo la n -ésima.

En torno a la figura de Alan Turing





El problema de la Parada (III)

- *¿Qué ocurrirá con $A(MT_n, n)$?:*
 - Si $A(MT_n, n)$ se para \rightarrow $MT_n(n)$ no acaba
 - Si $MT_n(n)$ no se para \rightarrow $A(n, n)$ acaba
 - Si $A(n, n)$ no se para \rightarrow $MT_n(n)$ acaba
 - Es decir, si $MT_n(n)$ no se para \rightarrow $MT_n(n)$ se para, ABSURDO!
- Por otra parte:
 - Si $A(MT_n, n)$ no se para \rightarrow $MT_n(n)$ acaba
 - Si $MT_n(n)$ se para \rightarrow $A(n, n)$ no acaba
 - Si $A(n, n)$ se para \rightarrow $MT_n(n)$ no acaba
 - Es decir, si $MT_n(n)$ se para \rightarrow $MT_n(n)$ no se para, ABSURDO!
 - Como hemos llegado a un absurdo, la hipótesis de partida es falsa.





El problema de la Parada (IV)

PRIMERO: Proponemos la existencia de un programa que:

Dado el número de Gödel de un programa

**Programa
propuesto**

Se detendrá con la variable $x = 1$ si la entrada representa un programa autoterminante o $x = 0$ si no es así.

ENTONCES: si existe tal programa, podemos modificarlo.

Añadiéndole una estructura mientras / fin

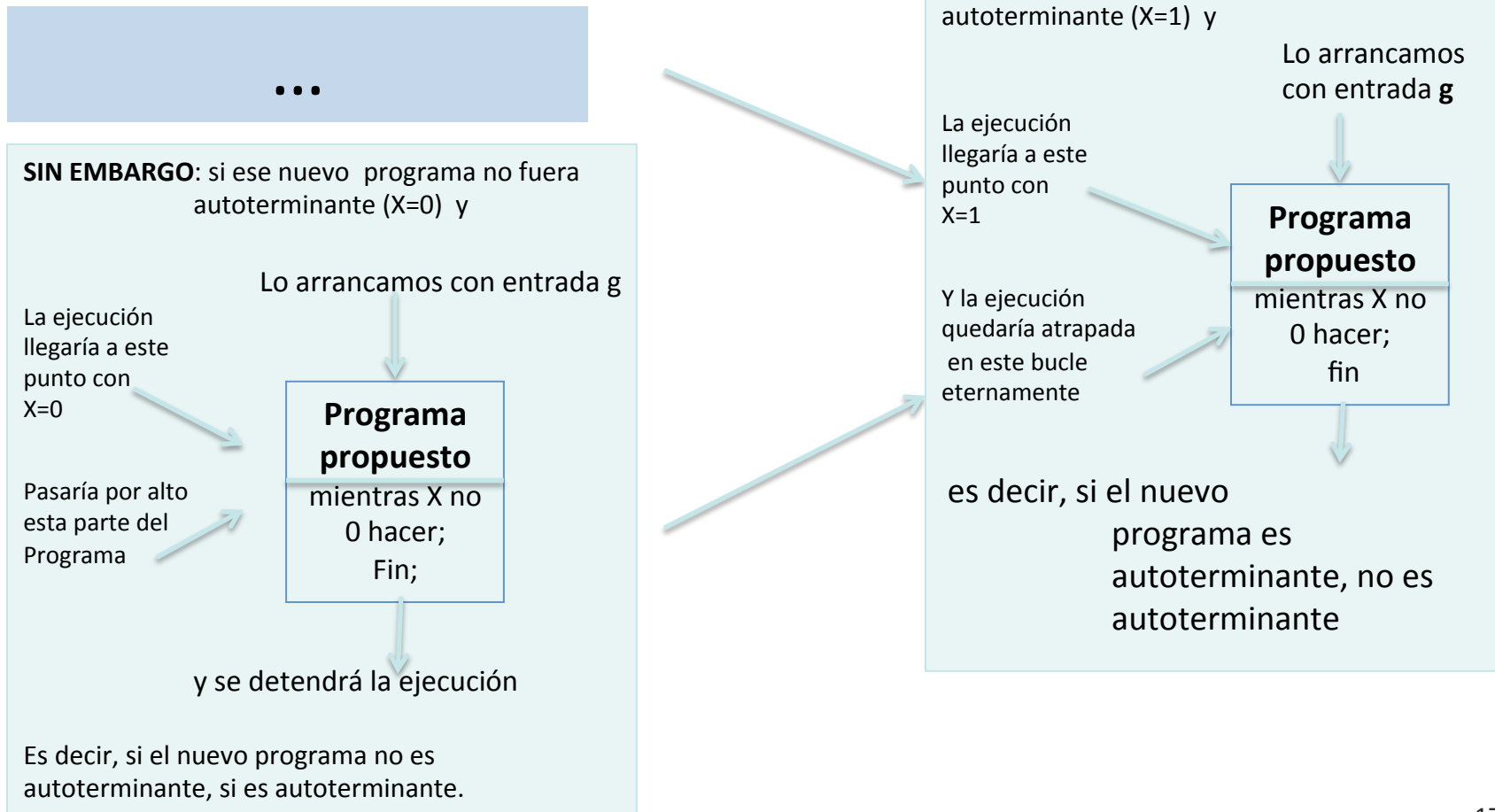
**Programa
propuesto**

mientras X no
0, hacer; fin;

Para producir un programa nuevo, con número de Gödel g



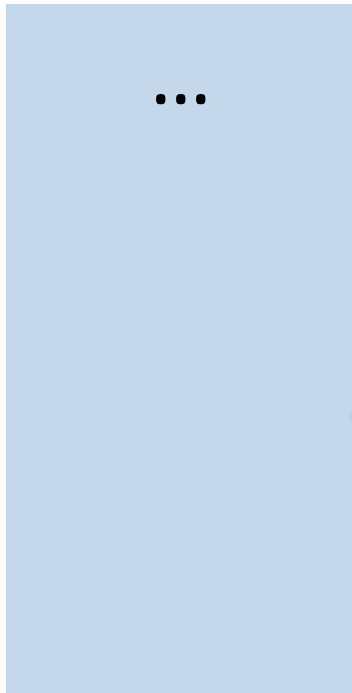
El problema de la Parada (V)



En torno a la figura de Alan Turing



El problema de la Parada (VI)



En CONSECUENCIA:

La existencia del Programa

la existencia de un nuevo programa propuesto

Programa propuesto



Conduciría a

Programa propuesto

mientras X no
0 hacer;
fin

que no es autoterminante ni no autoterminante.

Así que la existencia del programa propuesto es imposible.

¿Hay límites en la computación Turing?

- Tecnología actual
- Limitaciones
- Soluciones futuras:
 - ADN
 - Computación Cuántica
 - Otras...





¿Hay límites en la computación Turing?

- Problemas de construcción / tecnología:
 - Límites de las computadoras convencionales (2 generaciones más de miniaturización)
- Las computadoras del futuro
 - ADN
 - Moleculares
 - Cuánticas





Tecnología Actual

- Basada en silicio
 - 1 Chip de $0,5 \text{ cm}^2$ contiene millares de transistores de menos de 1μ de diámetro
 - Si: conduce la electricidad
 - Óxido de Si: es aislante
 - Fotolitografía: sobre las capas de Si o de Si_2O_3 película de polímero (P4HS) con el esquema del circuito deseado.
 - El patrón del circuito se graba en la película de polímero exponiéndolo a la luz a través de una máscara.
 - Después se aplican sustancias que corroen el material de silicio no protegido.





Limitaciones (I)

- Limitación \Rightarrow Longitud de onda empleada para fijar el patrón (1/5 de micra)
- Para crear componentes electrónicos de menor tamaño:
 - Cambiar a radiaciones de menor λ :
 - Luz UV
 - Rayos X (litografía de RX)
 - Haces de electrones de alta energía
 - Todavía no hay acuerdo. Colaboraciones entre grandes fabricantes: IBM, Motorola, Lucent Technologies y Lockheed Martin





Limitaciones (II)

- Límites de la miniaturización no sólo fotolitográficos
- Transistor al mínimo tamaño: aislante de la puerta tendrá un diámetro de 4-5 átomos: ¿seguirá proporcionando aislamiento?



David Miller y sus compañeros de Lucent Technologies \Rightarrow NO, fin del aislante





Futuro

- Soluciones:

- Computadoras “tipo organismo”
- Computadoras “tipo cubeta de laboratorio”
- Computadoras con nanotubos (dispositivos de carbono que pueden medir millonésimas de mm, nanómetro, menos de 1/10 parte de los cables que se pueden grabar en la actualidad en un circuito)
 - Además nanotubos son conductores
 - Pero, ¿quién procesará los bits?⇒moléculas que procesan bits no como impulsos eléctricos sino como fotones o como otros componentes moleculares.
 - No se sabe cómo conectar moléculas a un circuito fiable y complejo que sirva para calcular ⇒ **ordenador molecular**





Futuro. ADN

- **Leonar Adleman (U. California, 1994)**
 - Uso de moléculas para resolver problemas
 - Análogamente a cómo hacen las células
 - ADN: cadenas con 4 bases o símbolos distintos para codificar información
 - Usando técnicas de biotecnología: programar distintas cadenas de bases en hebras sintéticas de ADN
 - Las hebras se pueden generar (copiar) cortar y ensamblar en grandes cantidades

¿Se podría convencer al ADN de que resolviera problemas complejos de cálculo?





Futuro. ADN

- **Leonar Adleman (U. California, 1994)**
 - SI, ADN muy adecuado para resolver problemas de minimización
 - Ejemplo: el camino hamiltoniano
 - Si cada posible solución está codificada en una hebra de ADN, en un tubo puede haber millones y solo es cuestión de separar la mejor solución
 - Todo ocurre de forma paralela: todas las soluciones son creadas y examinadas al mismo tiempo
 - Todavía no está claro si se trata de computación “SUPERTURING” o más bien de computación eficiente de problemas intratables.





Futuro. Cuántica

- En los 60 algunos científicos se dieron cuenta de que la miniaturización conducía al reino de la física cuántica
- En los 80 se empezó a pensar cómo podría operar un ordenador bajo la influencia de la mecánica cuántica
 - 1ª ventaja \Rightarrow velocidad
 - Mezcla de estados: **qbit** objetos cuánticos superpuestos (gato de Schrödinger)
 - Así una serie de interruptores cuánticos -objetos en estados cuánticos bien definidos, como átomos en diferentes estados de excitación- posee bastantes más configuraciones de qubits que la correspondiente serie clásica de bits





Futuro. Cuántica

- ... cómo podría operar un ordenador bajo la influencia de la mecánica cuántica
 - una memoria clásica de tres bits puede almacenar sólo una de las ocho configuraciones posibles de unos y ceros, la correspondiente serie cuántica puede almacenar las ocho, en una superposición de estados
 - plasmar estas ideas en un dispositivo físico supone un reto descomunal: una superposición cuántica de estados es una cosa muy delicada, y difícil de mantener, sobre todo si está extendida por un enorme conjunto de elementos lógicos. Una vez que esta superposición empieza a interactuar con su entorno, comienza a desplomarse y la información cuántica se pierde por los alrededores.





Futuro. Cuántica

- ... cómo podría operar un ordenador bajo la influencia de la mecánica cuántica
 - Algunos investigadores creen que este problema volverá la informática cuántica a gran escala -en la que grandes cantidades de datos son manipulados en multitud de pasos- imposiblemente delicada y difícil de manejar
 - Pero el problema ha sido parcialmente solucionado en los últimos años por el desarrollo de algoritmos que permitirán funcionar a los ordenadores cuánticos, a pesar de los pequeños errores introducidos por este tipo de pérdidas.

En torno a la figura de Alan Turing





Futuro. Computadoras Ópticas

- Basadas en la luz
 - Construcción de un dispositivo óptico
 - Aprovechamiento de la velocidad de la luz
 - Aprovechamiento de la capacidad de transportar información de la luz
- Problema: el silicio emite energía térmica, no luminosa
- Construcción de trampas a escala atómica en el Si para atrapar electrones y lograr que emitan energía luminosa.
- Ventaja: a temperatura ambiente.





Futuro. Computadoras Neuroelectrónicas

- Basadas en la interacción del Si con tejidos vivos:
 - Primer neurochip: fusión de un microchip con células de caracol.





Futuro. Sin embargo....

- La capacidad de los procesadores sigue aumentando
- Trabajan en una estrategia de desarrollo basada en tecnología de múltiples núcleos, consistente en la introducción de dos o más procesadores en un sólo chip.
- Otras opciones:
 - Distribución del procesamiento y del almacenamiento: nubes

