



Universidad
de Huelva

Tema 4

Máquinas de Turing

- 4.1 Límites de los autómatas
- 4.2 Definición de Máquina de Turing
- 4.3 Ejemplos
- 4.4 Máquina de Turing con N cintas
- 4.5 Máquina de Turing no determinista
- 4.6 Máquina de Turing Universal
- 4.7 Representación de autómatas con Máquinas de Turing
- 4.8 Lenguajes aceptados por Máquinas de Turing

4.1 Límites de los autómatas

4.2 Definición de Máquina de Turing

4.3 Ejemplos

4.4 Máquina de Turing con N cintas

4.5 Máquina de Turing no determinista

4.6 Máquina de Turing Universal

4.7 Representación de autómatas con Máquinas de Turing

4.8 Lenguajes aceptados por Máquinas de Turing

- Los límites de los autómatas finitos y los autómatas de pila vienen descritos por los lemas de bombeo.

- Ejemplo de lenguaje que no puede reconocer un autómata finito

$$a^n b^n$$

- Ejemplo de lenguaje que no puede reconocer un autómata de pila

$$a^n b^n c^n$$

4.1 Límites de los autómatas

4.2 Definición de Máquina de Turing

4.3 Ejemplos

4.4 Máquina de Turing con N cintas

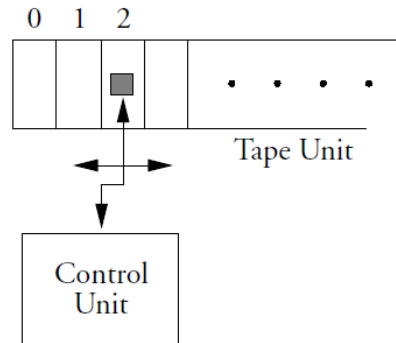
4.5 Máquina de Turing no determinista

4.6 Máquina de Turing Universal

4.7 Representación de autómatas con Máquinas de Turing

4.8 Lenguajes aceptados por Máquinas de Turing

- Una **Máquina de Turing** está formada por una unidad de control, que puede describirse mediante un autómata finito, una cinta de lectura y escritura, que tiene un comienzo a la izquierda y se extiende indefinidamente hacia la derecha, y un cabezal, que indica la posición de la cinta sobre la que trabaja la máquina en cada paso.



- Inicialmente la cinta contiene la cadena de entrada de la máquina (w), seguida de celdas en blanco (β).
- En cada paso, la máquina lee el contenido de la celda sobre la que se encuentra el cabezal, realiza un cambio de estado de la unidad de control, y escribe un dato sobre la celda o mueve el cabezal a izquierda o derecha. La máquina termina al alcanzar un estado especial llamado estado de parada (*halt*).



- Formalmente, una máquina de Turing es una sex-tupla $TM=(\Gamma, \beta, Q, \delta, s, h)$ donde:
 - Γ es el alfabeto de la cinta
 - β es el símbolo blanco de la cinta
 - Q es el conjunto de estados de la máquina
 - $\delta : Q \times (\Gamma \cup \{\beta\}) \rightarrow (Q \cup \{h\}) \times (\Gamma \cup \{\beta\} \cup \{L,R\})$ es la función de transición
 - s es el estado inicial de la máquina
 - h es el estado de parada de la máquina.



- Se dice que una máquina de Turing (M) *acepta una cadena de entrada* $w \in \Gamma^*$, si partiendo del estado inicial s y en la cinta se encuentra la cadena w seguida de blancos (β) la máquina M procesa la cadena y termina en el estado de parada (h).
- Se define el lenguaje aceptado por una máquina de Turing, $L(M)$, como el conjunto de todas las cadenas w aceptadas por la máquina.
- Los lenguajes aceptados por las máquinas de Turing se denominan *recursivamente enumerables*.

- Una función $f: \Gamma^* \rightarrow \Gamma^* \cup \{\perp\}$ se dice que es *parcial* si existe alguna entrada w tal que $f(w) = \perp$
- El símbolo \perp se lee *bottom* e indica que un determinado valor no está definido.
- Una función $f: \Gamma^* \rightarrow \Gamma^* \cup \{\perp\}$ se dice que es *total* si no existe ninguna entrada w tal que $f(w) = \perp$, es decir, si está definida sobre todo el conjunto de entrada.
- El **valor computado por máquina de Turing** para una entrada w se define como el contenido de la cinta cuando la máquina alcanza el estado de parada.
- Puesto que pueden existir entradas w para las que la máquina de Turing no alcance nunca un estado de parada, la **función computada por una máquina de Turing** es una función parcial.

- Una máquina de Turing puede ser utilizada como un reconocedor de lenguaje. Dada una cadena w , la máquina puede decidir si pertenece al lenguaje o no, generando la salida adecuada.
- Se dice que un lenguaje es *decidible* si se puede construir una máquina de Turing que alcance siempre el estado de parada y reconozca el lenguaje.
- Un lenguaje puede ser aceptable pero no decidible, es decir, que si $w \in L$ la máquina de Turing alcanza siempre el estado de parada, pero si $w \notin L$ podría no parar.

4.1 Límites de los autómatas

4.2 Definición de Máquina de Turing

4.3 Ejemplos

4.4 Máquina de Turing con N cintas

4.5 Máquina de Turing no determinista

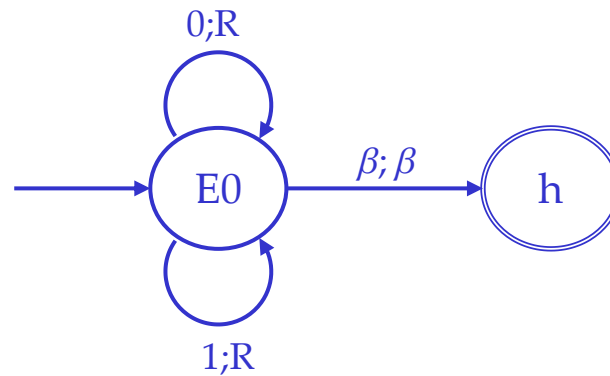
4.6 Máquina de Turing Universal

4.7 Representación de autómatas con Máquinas de Turing

4.8 Lenguajes aceptados por Máquinas de Turing

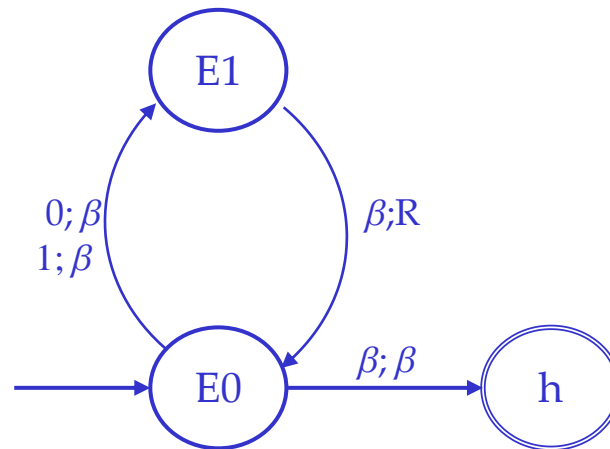
- Ejemplos
 - Recorrer la cadena hasta el final.
 - Borrar la cadena.
 - Buscar un 1.
 - Comprobar que dos cadenas son iguales ($w#w$).
 - Abrir un hueco en la primera posición ($w \rightarrow #w$).
 - Reconocer el lenguaje $a^n b^n c^n$

- Recorrer la cadena hasta el final.



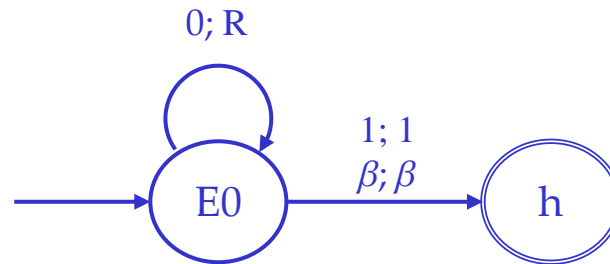
Estado	0	1	β
E0	R ; E0	R ; E0	β ; h

- Borrar la cadena.



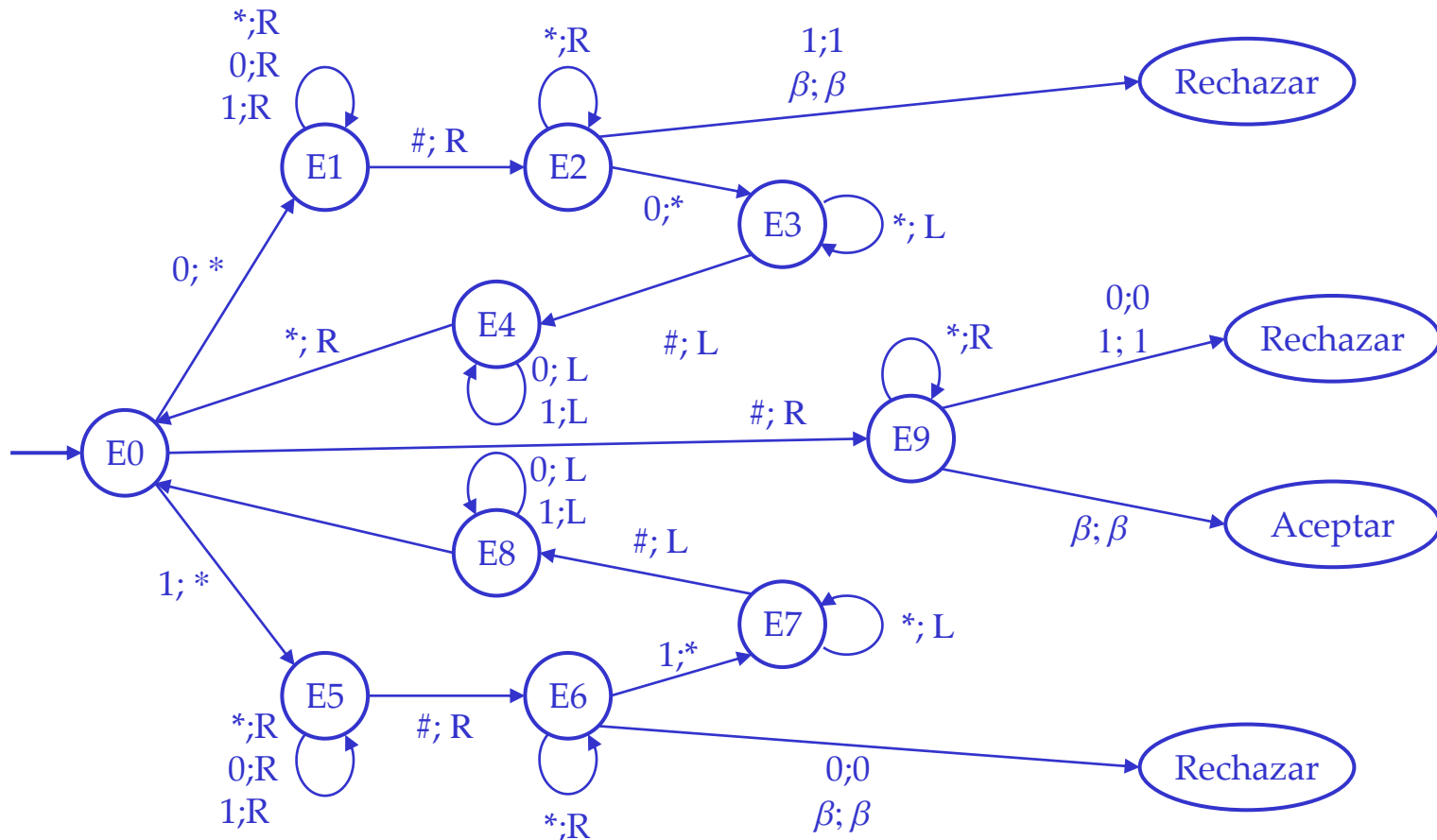
Estado	0	1	β
E0	$\beta ; E1$	$\beta ; E1$	$\beta ; h$
E1	$R ; E0$	$R ; E0$	$R ; E0$

- Buscar un 1.

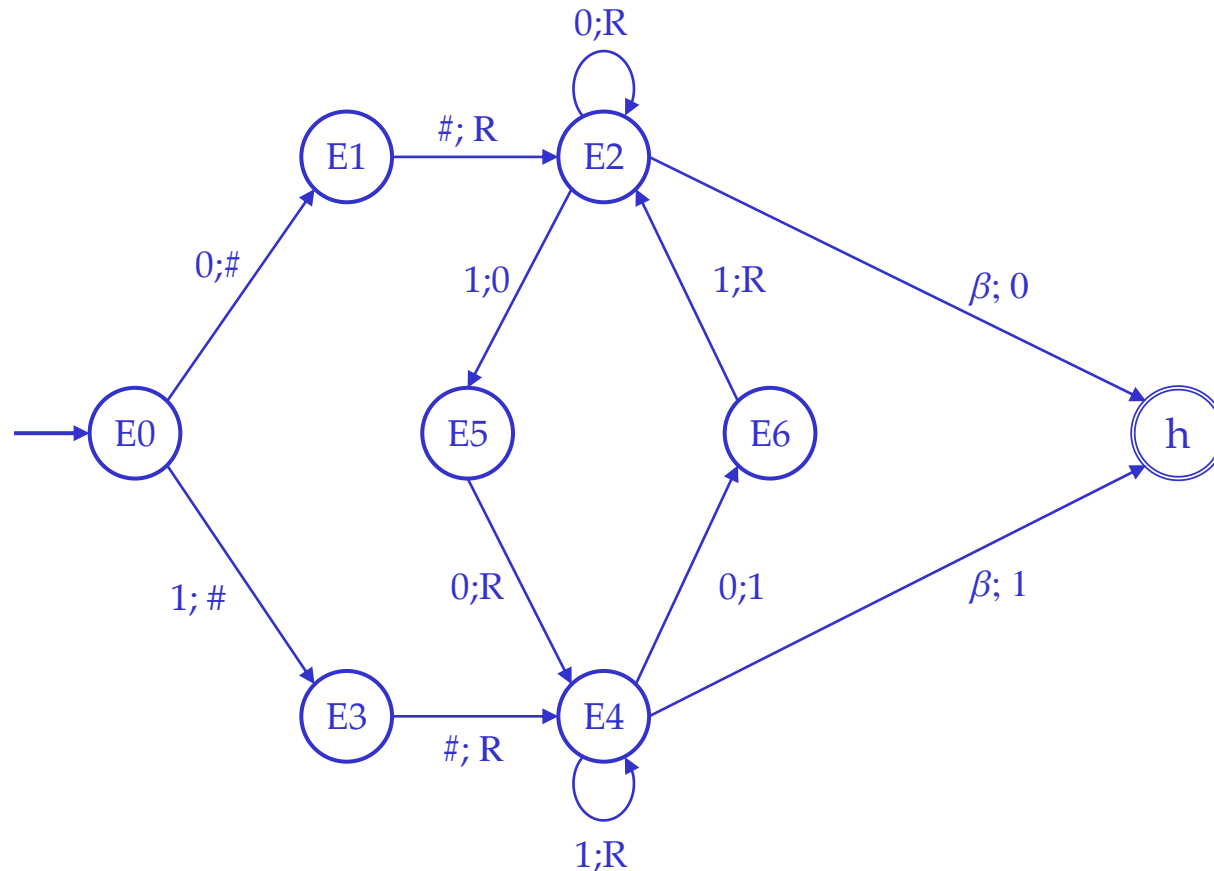


Estado	0	1	β
E0	R; E0	1; h	β ; h

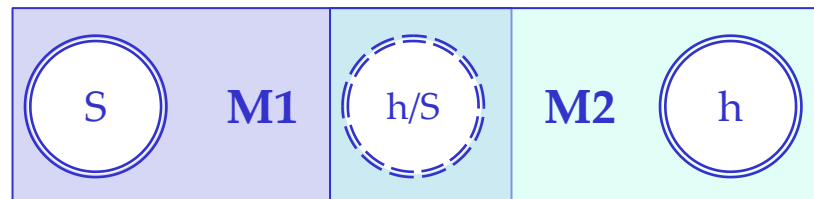
- Comprobar que dos cadenas son iguales ($w#w$).



- Abrir un hueco en la primera posición ($w \rightarrow \#w$).



- Toda máquina de Turing tiene un único estado inicial y un único estado de parada.
- Esto permite tratar las máquinas de Turing como rutinas que pueden componerse. Es decir, se puede construir una máquina de Turing M combinando dos máquinas de Turing $M1$ y $M2$. Para eso se sustituye el estado de parada de $M1$ por el estado inicial de $M2$. El resultado es una máquina de Turing que ejecuta $M1$ y, a continuación, ejecuta $M2$.



4.1 Límites de los autómatas

4.2 Definición de Máquina de Turing

4.3 Ejemplos

4.4 Máquina de Turing con N cintas

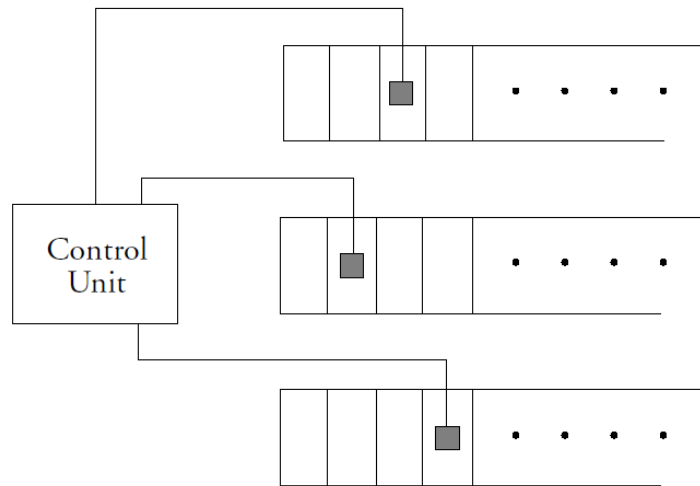
4.5 Máquina de Turing no determinista

4.6 Máquina de Turing Universal

4.7 Representación de autómatas con Máquinas de Turing

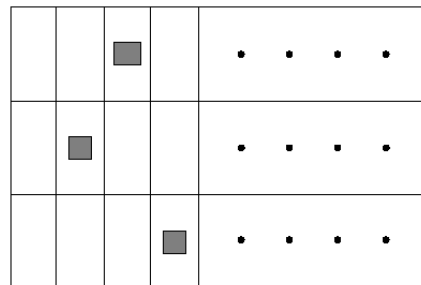
4.8 Lenguajes aceptados por Máquinas de Turing

- Una máquina de Turing con N cintas es un modelo extendido de máquina de Turing en el que existen N cintas con N cabezales. En cada paso, la unidad de control lee el contenido de las N cintas, realiza un cambio de estado y realiza una acción (escritura, desplazamiento a la izquierda o desplazamiento a la derecha) sobre cada cinta.



- Las máquinas de Turing con N cintas tienen la misma capacidad de cómputo que las máquinas de Turing con una sola cinta.
- Teorema: Para cada máquina de Turing con N cintas, M_N , existe una máquina de Turing con una única cinta, M_1 , tal que, una computación de M_N que finalice en T pasos se puede simular mediante M_1 en $O(T^2)$ pasos.

- Demostración:
 - Sea Γ el alfabeto utilizado en M_N . Se define Γ' el alfabeto de M_1 como un alfabeto formado por $(2 \cdot |\Gamma|)^N$ símbolos. Esto permite modelar las N cintas por medio de una única cinta en la que en cada celda se almacena la información de N celdas paralelas. El factor 2 permite además añadir una marca a los símbolos, de manera que la posición de los N cabezales de M_N se simulará con una marca en la celda correspondiente.



- Demostración:
 - M_1 simula el comportamiento de M_N en dos pasadas. En la primera pasada se recorre la cinta buscando las marcas para identificar la entrada equivalente de M_N . En la segunda pasada se ejecuta la acción de M_N desplazando o escribiendo el dato que corresponda en cada cinta.
 - Si M_N realiza la ejecución en T pasos, la distancia entre las marcas será como mucho de T posiciones (suponiendo que un cabezal se quede quieto y otro siempre avance a la derecha). Por tanto, cada transición de M_N se puede simular con $O(T)$ transiciones en M_1 . La ejecución completa en M_1 será de orden $O(T^2)$.

4.1 Límites de los autómatas

4.2 Definición de Máquina de Turing

4.3 Ejemplos

4.4 Máquina de Turing con N cintas

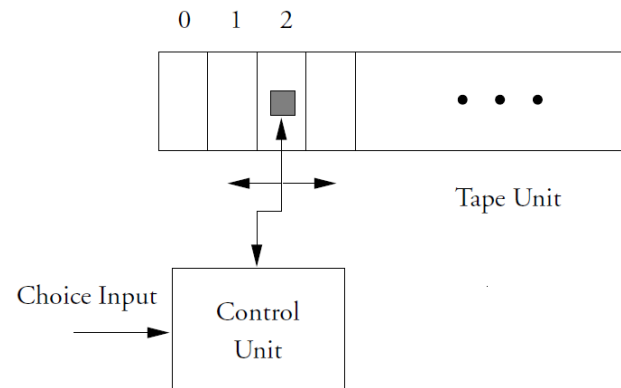
4.5 Máquina de Turing no determinista

4.6 Máquina de Turing Universal

4.7 Representación de autómatas con Máquinas de Turing

4.8 Lenguajes aceptados por Máquinas de Turing

- Una máquina de Turing no determinista (*NDTM*) es una máquina de Turing en la que pueden existir varias transiciones a partir del mismo estado y lectura del cabezal. Esto significa que, dado un estado y un símbolo de entrada, es posible elegir la transición a efectuar entre varias opciones.
- Una *NDTM* se puede modelar como una máquina de Turing con una entrada adicional que permite seleccionar la transición a efectuar entre las varias posibilidades de cada paso.

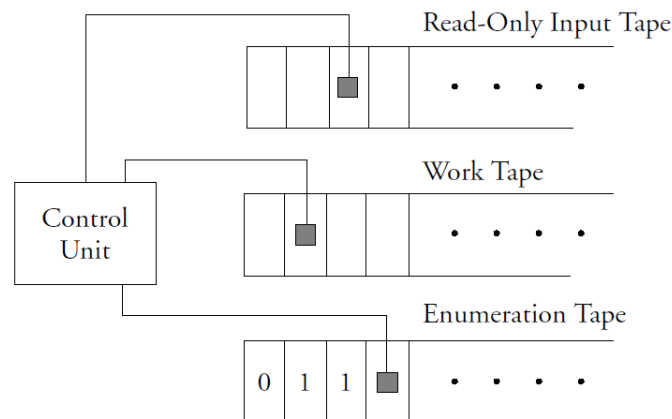




- Se pueden reescribir las transiciones añadiendo nuevos estados de manera que en cada caso la elección se efectue solo entre dos opciones, con lo que el selector podría ser una señal de un bit.
- Dada una cadena de entrada w , se dice que la *NDTM* acepta la cadena si existe algún camino que permita aceptarla, es decir, si existe una cadena de selección c (*choice string*) que lleve la máquina desde el estado inicial al estado de parada.
- Una cadena w no es aceptada por la *NDTM* si no existe ninguna cadena de selección que permita aceptarla.
- El lenguaje $L(M)$ aceptado por la *NDTM* es el lenguaje formado por todas las cadenas aceptadas.

- Formalmente, una máquina de Turing no determinista es una sep-tupla $\text{NDTM}=(C, \Gamma, \beta, Q, \delta, s, h)$ donde:
 - C es el alfabeto del selector.
 - Γ es el alfabeto de la cinta.
 - β es el símbolo blanco de la cinta.
 - Q es el conjunto de estados de la máquina.
 - $\delta : Q \times (\Gamma \cup \{\beta\}) \times C \rightarrow (Q \cup \{h\}) \times (\Gamma \cup \{\beta\}) \cup \{L, R\}$ es la función de transición.
 - s es el estado inicial de la máquina.
 - h es el estado de parada de la máquina.

- Teorema: Todo lenguaje aceptado por una máquina de Turing no determinista puede ser aceptado por una máquina de Turing determinista.
- Demostración:
 - Dada una NDTM es posible construir una TM determinista formada por 3 cintas. La primera cinta contiene la cadena de entrada w y se utilizará en modo sólo lectura. La segunda cinta será una cinta de trabajo en la que se computará la NDTM. La tercera cinta servirá para enumerar los códigos de la cadena de selección.



- El comportamiento de la TM consiste en simular el comportamiento de la NDTM sobre la cinta de trabajo, asumiendo como entrada de selección el contenido de la tercera cinta. Los valores de esta cinta se generan en orden lexicográfico: 0, 1, 00, 01, 10, 11, 000, ...
- Si el resultado de la simulación no es correcto (es decir, si se alcanza un punto en el que no existe transición o se alcanza el final de la cadena de selección) se incrementará el valor del código de selección, se copiará de nuevo la entrada w sobre la cinta de trabajo y se volverá a simular la NDTM con la nueva cadena de selección.
- Si en algún valor de la cadena de selección c se consigue aceptar la cadena de entrada w entonces se saltará al estado de parada de la TM, aceptando la entrada.



- Observación: Todo lenguaje aceptado por una NDTM puede ser aceptado por una TM determinista, pero eso no quiere decir que el número de recursos que necesita cada una sea del mismo orden.

4.1 Límites de los autómatas

4.2 Definición de Máquina de Turing

4.3 Ejemplos

4.4 Máquina de Turing con N cintas

4.5 Máquina de Turing no determinista

4.6 Máquina de Turing Universal

4.7 Representación de autómatas con Máquinas de Turing

4.8 Lenguajes aceptados por Máquinas de Turing

- Una *Máquina de Turing Universal*, *UTM*, es una máquina de Turing capaz de simular el comportamiento de cualquier máquina de Turing sobre cualquier cadena de entrada.
- Para poder construir una UTM es necesario definir una codificación de las máquinas de Turing, de manera que podamos indicar la máquina de Turing a simular por medio de su código.
- La codificación canónica de una máquina de Turing es una cadena formada mediante el alfabeto $\Lambda = \{ <, >, [,], \#, 0, 1, \beta, R, L \}$.

- Los valores de los estados se representan en codificación unaria:

1, 11, 111, 1111, 11111, ...

- El estado inicial es el “1” y el estado de parada es la cadena vacía.
- Las transiciones $\delta(p,a)=(q,z)$ se representan mediante la cadena $\langle z\#q \rangle$, donde el estado q se representa en unario y el símbolo $z \in \{0, 1, \beta, R, L\}$:

$\langle 0\#11 \rangle$, $\langle L\#1111 \rangle$, $\langle R\# \rangle$

- Para un estado p , las transiciones $\delta(p,0)$, $\delta(p,1)$ y $\delta(p,\beta)$ se representan entre corchetes:

$[\langle R\#1111 \rangle \langle L\#11 \rangle \langle 0\#11111 \rangle]$

- La codificación canónica de una máquina de Turing consiste en la enumeración ordenada de las transiciones de cada estado.

- Por ejemplo, la codificación de la máquina de Turing que inserta un símbolo β a la izquierda y mueve la cadena de entrada una posición a la derecha es la siguiente:

$[\langle\beta\#11\rangle\langle\beta\#111\rangle\langle\beta\#\rangle]$

$[\langle R\#1111\rangle\langle R\#1111\rangle\langle R\#1111\rangle]$

$[\langle R\#11111\rangle\langle R\#11111\rangle\langle R\#11111\rangle]$

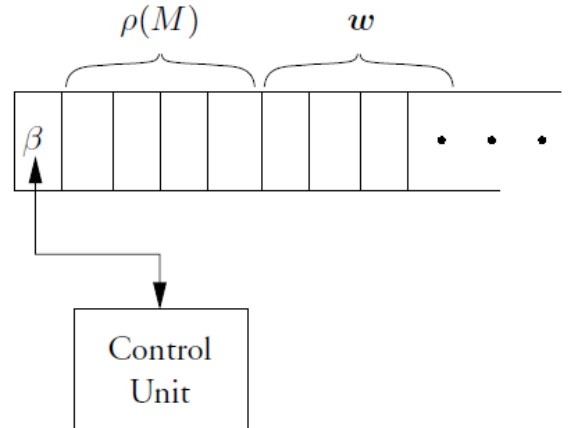
$[\langle 0\#11\rangle\langle 0\#111\rangle\langle 0\#\rangle]$

$[\langle 1\#11\rangle\langle 1\#111\rangle\langle 1\#\rangle]$

- La *Máquina de Turing Universal (UTM)* es una máquina de Turing que es capaz de simular cualquier máquina de Turing (descrita en codificación canónica) sobre cualquier cadena de entrada.
- Para construir una UTM vamos a considerar un alfabeto de 20 símbolos, los utilizados en la representación canónica con marca o sin marca:

$$\underline{\Delta} = \{ <, >, [,], \#, 0, 1, \beta, R, L \} \cup \{ \leq, \geq, \lfloor, \rfloor, \#, \underline{0}, \underline{1}, \underline{\beta}, \underline{R}, \underline{L} \}$$

- Para simular una máquina de Turing (TM) sobre una cierta entrada (w) consideraremos que el contenido inicial de la cinta será un símbolo β , seguido de la codificación canónica de TM y a continuación la cadena w .



- El funcionamiento de la UTM es el siguiente:
 1. Para indicar la posición del cabezal de la máquina simulada se utiliza una marca sobre el símbolo correspondiente de w .
 2. Para indicar el estado en que se encuentra la máquina simulada se utiliza una marca sobre el símbolo $[$ de comienzo de sus transiciones.
 3. En cada paso de simulación de la máquina TM hay que identificar el símbolo sobre el que se encuentra el cabezal (haciendo un recorrido buscando 0 , 1 o β).
 4. A continuación hay que desplazarse hacia la izquierda buscando el símbolo $[$ de comienzo del estado actual de TM.
 5. Una vez seleccionado el comienzo del estado, se busca la transición correspondiente al símbolo del cabezal (primer ' \leftarrow ' si es 0 , segundo ' \leftarrow ' si es 1 o tercer ' \leftarrow ' si es β) y se marca ' $\underline{\leftarrow}$ '.

- Funcionamiento de la UTM (sigue):
 6. Para ejecutar la transición se lee la acción asociada $(0,1,\beta,R,L)$ y se desplaza a la derecha buscando el cabezal de TM para ejecutar la acción: escribir $\underline{0}$, escribir $\underline{1}$, escribir β , desmarcar el símbolo y marcar el de la derecha o desmarcar el símbolo y marcar el de la izquierda.
 7. El siguiente paso es colocar TM en el estado resultado de la transición. Para ello hay que desplazarse a la izquierda buscando el símbolo \leq , desmarcarlo y desplazarse tres celdas a la derecha buscando el símbolo siguiente de #.
 8. Si el siguiente de # es el símbolo $>$ entonces se trata de una transición al estado de parada de TM, por lo que hay que saltar al estado de parada de UTM.
 9. Si el siguiente de # es el símbolo 1 se trata del comienzo de un estado. Hay que marcar 1, desplazarse al comienzo de la representación canónica, marcar el [y volver al símbolo $\underline{1}$, desmarcarlo y desplazarse a la derecha.

- Funcionamiento de la UTM (sigue):
 10. Cada vez que nos encontremos un 1 hay que marcarlo, desplazarse a la izquierda buscando \lfloor , desmarcarlo, desplazarse a la derecha buscando \lfloor , marcarlo, desplazarse a la derecha buscando $\underline{1}$, desmarcarlo y desplazarse una celda a la derecha.
 11. Cuando encontremos el símbolo $>$ habremos dejado marcado con \lfloor el comienzo del estado actual de TM y podremos desplazarnos a la derecha buscando el símbolo marcado que indica la posición del cabezal de TM, volviendo al punto 3 de este algoritmo.

4.1 Límites de los autómatas

4.2 Definición de Máquina de Turing

4.3 Ejemplos

4.4 Máquina de Turing con N cintas

4.5 Máquina de Turing no determinista

4.6 Máquina de Turing Universal

4.7 Representación de autómatas con Máquinas de Turing

4.8 Lenguajes aceptados por Máquinas de Turing

- Autómata Finito
 - Se puede desarrollar un Autómata Finito mediante una Máquina de Turing con dos cintas, una cinta de solo lectura (para leer la entrada) y una cinta de solo escritura (para escribir la salida). Los estados de la Máquina de Turing corresponden a los estados del Autómata.
 - Cada transición $\langle p, a; q, b \rangle$ del Autómata Finito corresponde a una transición de la Máquina de Turing en la que se lee el símbolo 'a' de la cinta de entrada, se desplaza la cinta de entrada a la derecha, se escribe el símbolo 'b' en la cinta de salida, se desplaza la cinta de salida a la derecha y se cambia del estado 'p' al estado 'q'.
 - Los estados de aceptación del Autómata se traducen en transiciones al estado de parada de la Máquina de Turing.

- Autómata de Pila
 - Se puede desarrollar un Autómata de Pila mediante una Máquina de Turing con dos cintas, una cinta de solo lectura (para leer la entrada) y una cinta para simular la pila.
 - Las acciones de apilar un símbolo 'a' se traducen en una transición que escriba 'a' en la pila seguida de una transición que mueva el cabezal de la pila a la derecha.
 - Las acciones de desapilar un símbolo 'a' se traducen en una transición que escriba un carácter blanco ' β ' en la pila seguida de una transición que mueva el cabezal a la izquierda.
 - Cada transición del Autómata de Pila que consume un carácter de la cadena de entrada se traduce en una transición de la Máquina de Turing que mueva la cinta de lectura hacia la derecha.

- Máquina de Turing
 - Una Máquina de Turing puede simularse mediante un autómata de dos pilas.
 - En la primera pila se almacena el contenido de la cinta desde el inicio hasta la posición del cabezal.
 - En la segunda pila se almacena el contenido de la pila desde la celda a la derecha del cabezal hasta que comiencen los símbolos blancos.
 - Una transición a la izquierda corresponde a desapilar un símbolo de la primera pila y apilarlo en la segunda.
 - Una transición a la derecha corresponde a desapilar un símbolo de la segunda pila (si no se ha llegado al final) y apilarlo en la primera. Si la segunda pila ha llegado al final se apila un símbolo blanco ' β ' en la primera pila.
 - Para escribir un símbolo ' a ' en la cinta se desapila la primera pila y se apila el símbolo ' a '.

4.1 Límites de los autómatas

4.2 Definición de Máquina de Turing

4.3 Ejemplos

4.4 Máquina de Turing con N cintas

4.5 Máquina de Turing no determinista

4.6 Máquina de Turing Universal

4.7 Representación de autómatas con Máquinas de Turing

4.8 Lenguajes aceptados por Máquinas de Turing

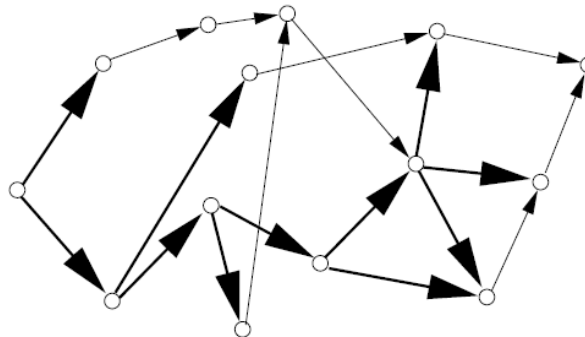
- Los lenguajes aceptados por las Máquinas de Turing son lenguajes con estructura de frase, es decir,
 - a) Dado un lenguaje recursivamente enumerable (L_M), reconocido por una Máquina de Turing (M), existe una Gramática con estructura de frase (G) que reconoce el mismo lenguaje ($L_G=L_M$).
 - b) Dada una Gramática con estructura de frase (G), existe una Máquina de Turing (M) que reconoce el lenguaje descrito por la gramática ($L_M=L_G$).
- NOTA: Los lenguajes con estructura de frase son aquellos cuyas reglas de producción son del tipo: $\alpha \rightarrow \beta$
 - $\alpha \in (N \cup \Sigma)^* N (N \cup \Sigma)^*$
 - $\beta \in (N \cup \Sigma)^*$

- Configuración de una Máquina de Turing de una cinta en un instante:
 - El estado de ejecución de una Máquina de Turing en un instante dado puede representarse como

$$[x_1 x_2 \dots \mathbf{p} x_j \dots x_n]$$

indicando que el contenido de la cinta es $(x_1 x_2 \dots x_n)$, que la máquina se encuentra en el estado \mathbf{p} y que el cabezal se encuentra sobre la celda j -ésima.

- Grafo de configuración:
 - Un grafo de configuración $G(M,w)$ de una Máquina de Turing (determinista o no determinista) es un grafo dirigido cuyos vértices son configuraciones de la Máquina de Turing.
 - La raíz del grafo corresponde a la configuración inicial en el que en la cinta de entrada se encuentra la cadena w , el cabezal se encuentra al comienzo de la cinta y la Máquina de Turing se encuentra en el estado inicial.
 - Existe un conexión de un vértice v_0 a v_1 si la Máquina de Turing puede cambiar de la configuración inicial a la final en un único paso.



- Lenguaje con estructura de frase asociado a una Máquina de Turing
 - Dada una Máquina de Turing (M), que acepta un lenguaje recursivamente enumerable (L_M), existe una gramática G que describe el mismo lenguaje.

DEMOSTRACIÓN:

- Vamos a suponer, sin pérdida de generalidad, que al aceptar una cierta cadena del lenguaje (w) el contenido inicial de la cinta es $(\beta w_1 w_2 \dots w_n \beta \beta \dots)$, y el contenido final de la cinta es $(\beta 1 \beta \beta \dots)$.
- La configuración inicial de la Máquina es $[s \beta w_1 w_2 \dots w_n \beta]$ y la configuración final es $[h \beta 1 \beta]$.
- La gramática G tiene como símbolos terminales los mismos que la máquina de Turing (para reconocer el mismo lenguaje) y como símbolos no terminales los estados de M , el símbolo inicial S y los símbolos auxiliares $[]$ y β .

- Lenguaje con estructura de frase asociado a una Máquina de Turing

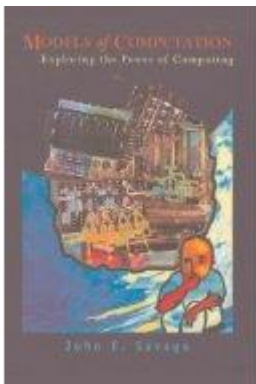
DEMOSTRACIÓN (SIGUE):

- Las reglas de la gramática G son las siguientes:
- $S \rightarrow [h \beta 1 \beta]$
- $\beta \beta] \rightarrow \beta]$
- $[s \beta \rightarrow \epsilon$
- $\beta] \rightarrow \epsilon$
- $x \mathbf{q} \rightarrow \mathbf{p} x$ para cada transición de la forma $\langle \mathbf{p}, x; \mathbf{q}, \mathbf{R} \rangle$
- $\mathbf{q} z x \rightarrow z \mathbf{p} x$ para cada transición de la forma $\langle \mathbf{p}, x; \mathbf{q}, \mathbf{L} \rangle$
- $\mathbf{q} y \rightarrow \mathbf{p} x$ para cada transición de la forma $\langle \mathbf{p}, x; \mathbf{q}, y \rangle$

- Máquina de Turing asociada a un Lenguaje con estructura de frase
 - Dada una gramática con estructura de frase, G , se puede construir una Máquina de Turing que reconoce el lenguaje $L(G)$.
- DEMOSTRACIÓN:
 - Una forma es construir una Máquina de Turing No Determinista con dos cintas. En la primera cinta se encuentra la cadena de entrada y en la segunda cinta se parte del símbolo inicial de G .
 - Las transiciones de la Máquina de Turing sobre la segunda cinta ejecutan las reglas de G . Como a partir de una forma sentencial es posible ejecutar varias reglas, el comportamiento de la Máquina de Turing es no determinista.
 - Cuando se genera una cadena a partir del símbolo inicial, la Máquina de Turing la compara con la cadena de la primera cinta. Si coincide, la cadena se acepta.



Bibliografía



- Savage, John E. (1998). “Models Of Computation: Exploring the Power of Computing”. Capítulo 5