

# Variantes de Máquinas de Turing

Teoría de Algoritmos I (75.29 / 95.06)

Ing. Víctor Daniel Podberezski

✉ [vpodberezski@fi.uba.ar](mailto:vpodberezski@fi.uba.ar)

# Variantes de las Maquinas de turing

## Existen múltiples variantes de las maquinas de Turing

Con posibilidad de no avanzar ademas de retroceder o avanzar el cabezal

Con múltiple cinta

No deterministica

Con impresora ( cinta output)

...

## Todos estos

Se han demostrado que tienen el mismo poder de computo

(pueden reconocer los mismos lenguajes)

# Turing Machine multicinta

## Corresponde a una variante de una TM

Con múltiples cintas

### Cada cinta

Tiene un cabezal para leer y escribir

### Inicialmente

La primer cinta contiene el input

El resto de las cintas se encuentra en blanco

# Turing Machine multicinta (cont.)

## La función de transición

Permite leer, escribir y moverse en alguna o todas las cintas

## Formalmente, con $k$ cintas

$$\delta: Q \times \Gamma^k \rightarrow Q \times \Gamma^k \times \{L, R, S\}^k$$

(S implica que el cabezal no se mueve)

**Por ejemplo  $\delta(q_i, a_1, \dots, a_k) = (q_j, b_1, \dots, b_k, L, R, \dots, L)$**

Se lee “si la maquina esta en el estado  $q_i$  y leo en la cinta  $x$  el símbolo  $a_x$

Entonces se realiza una transición al estado  $q_j$ , se graba en la cinta y el símbolo  $b_y$  y en cada cinta muevo al cabezal según lo que se indica

# Poder de cómputo de la TM multicinta

**Si una TM multicinta puede reconocer mas lenguajes que una TM**

Entonces tiene mas poder de computo o es “más robusta”

**Si demostramos que dado una maquina**

se puede construir otra equivalente

Entonces reconocen los mismos lenguajes

**Mediante la construcción teórica de cualquier TM multicinta**

En su equivalente TM probaremos que estos modelos son equivalentes

# Equivalencia entre TM y TM multicinta

## Sea una TM de K cintas (MTM)

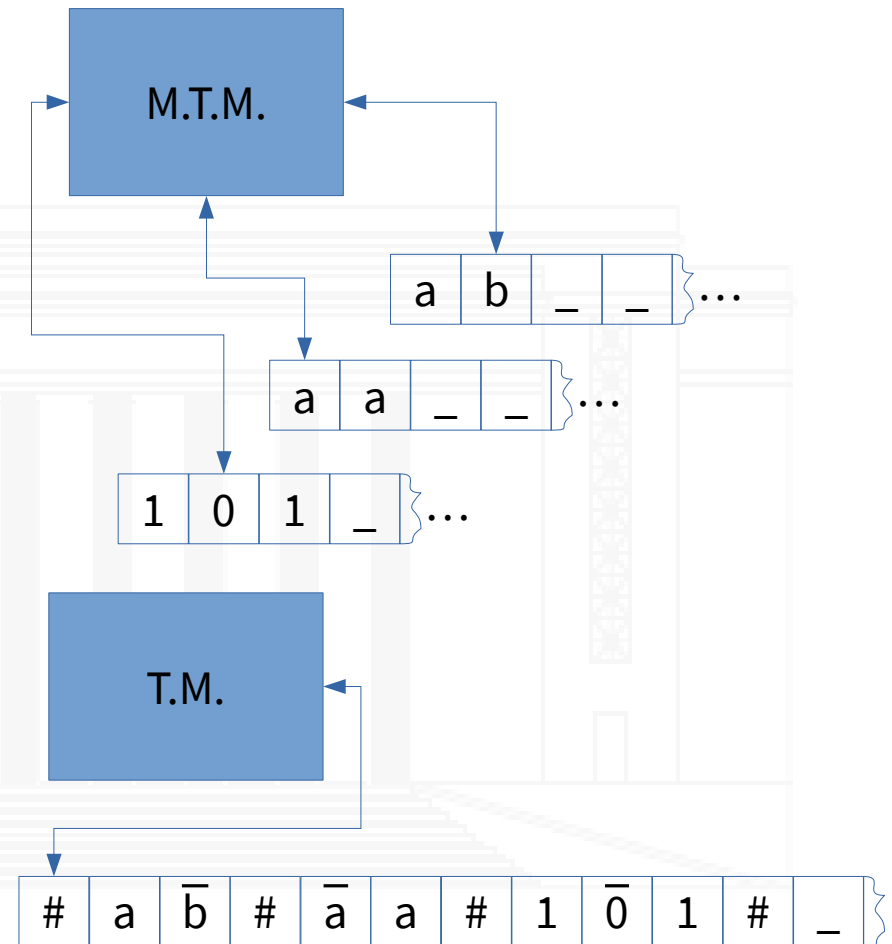
Generaremos una TM equivalente

### Pasos

Creamos Un nuevo carácter especial “#” de separacion de cintas

Concatenamos cada input de la cinta de la MTM a la la cinta de la TM separados por el “#”

Por cada símbolo en la MTM creamos uno equivalente marcado que usaremos para identificar la posición virtual de un cabezal



# Equivalencia entre TM y TM multicinta (cont.)

## Para simular una iteración

La TM explora la cinta desde el primer # al  $(k+1)$ -ésimo # para determinar los símbolos marcados y con eso la posición de las cabezas virtuales

Luego realiza una segunda pasada para actualizar las cintas de acuerdo a lo que dicta la función de transición de la M.T.M.

## Si en alguna iteración

Desde el cabezal virtual llega a la marca “#” que limita el contenido de esa cinta

Se debe mover todos los símbolos siguientes 1 posición creando una celda nueva en blanco “\_”

Luego continua la simulación

# Turing machine no determinística

## Una TM no determinística (NDTM)

Permite en un punto del computo transicionar en simultaneo a varias posibilidades

## Podemos describir su función de transición como

$$\delta: Q \times \Gamma \rightarrow P(Q \times \Gamma \times \{L, R\})$$

## El cómputo de una TM no determinística

Corresponde a un árbol de cuyas ramas corresponden a diferentes posibilidades de la máquina

Si al menos 1 rama termina en el estado de aceptación, entonces el resultado es de aceptación



# Equivalencia entre TM y NDTM

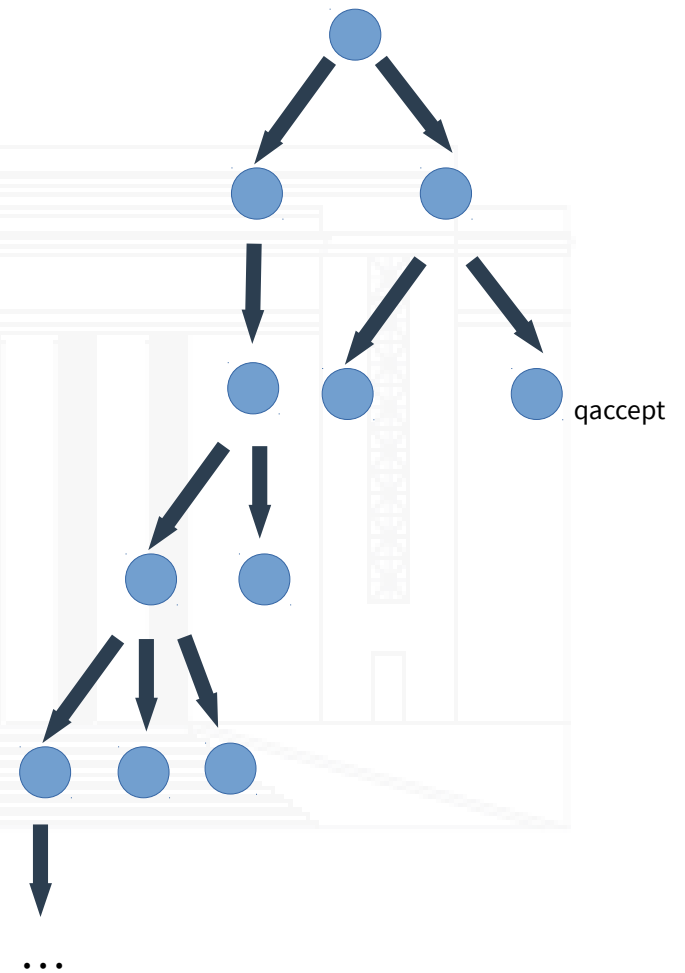
## Podemos simular una TM no deterministica

Para eso realizaremos una nueva TM que recorra cada nodo del árbol de la computación no deterministica

Si encuentra un estado de aceptación termina, sino continua (pudiendo looppear por siempre)

El recorrida del arbol debe realizarse mediante Breadth first search

(usar DFS podria ocasionar seleccionar una rama que loopea para siempre )





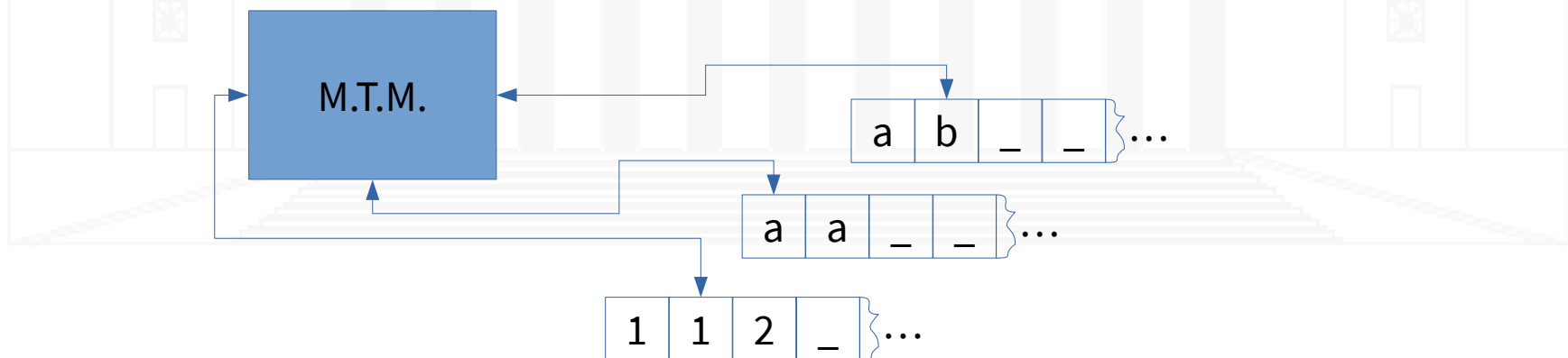
# Equivalencia entre TM y NDTM (cont.)

## Utilizaremos un MTM con 3 cintas

La cinta 1 contendrá el input del NDTM

La cinta 2 se utilizara para simular el computo del input hasta una rama del árbol de computación no determinístico

La cinta 3 realiza un seguimiento de la ubicación en el cálculo del arbol de computo



# Equivalencia entre TM y NDTM (cont.)

1. inicialmente la cinta 1 contiene el input y la cinta 3 el string vacio (raiz)
2. Repetir hasta aceptación (cada testeo es una rama del árbol)

copiar el contenido de la cinta 1 a la cinta 2

usar la cinta 2 para simular la NDTM con el input

En cada iteración

Aplicar la función de transición usando la opción de transición que indica el cabezal en la cinta 3 entre las posibles.

Si la rama a elegir no existe, o se llega a un estado de rechazo seguir con próxima rama

Si se llega a estado de aceptación terminar todo el proceso “aceptando”.

Sino, mover a la derecha el cabezal de cinta 3 y repetir proceso de iteración del cabezal

Asignar a la cinta 3 la etiqueta de la próxima rama a simular

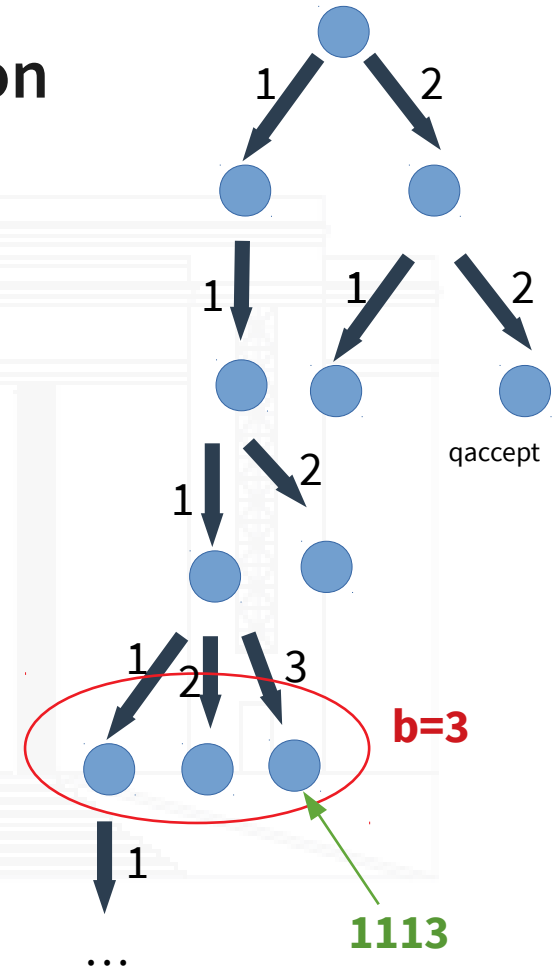
## Equivalencia entre TM y NDTM (cont.)

**En el ejemplo, la secuencia de ramas a probar son**

E, 1, 2, 3, 11, 12, 13, 21, 22, 23, 31, 32, 33, 111, 112 ...

(en azul aquellas “direcciones” invalidas)

**El proceso terminará al simular la rama “22”**





Presentación realizada en Julio de 2020