Formal Language & Automata Theory

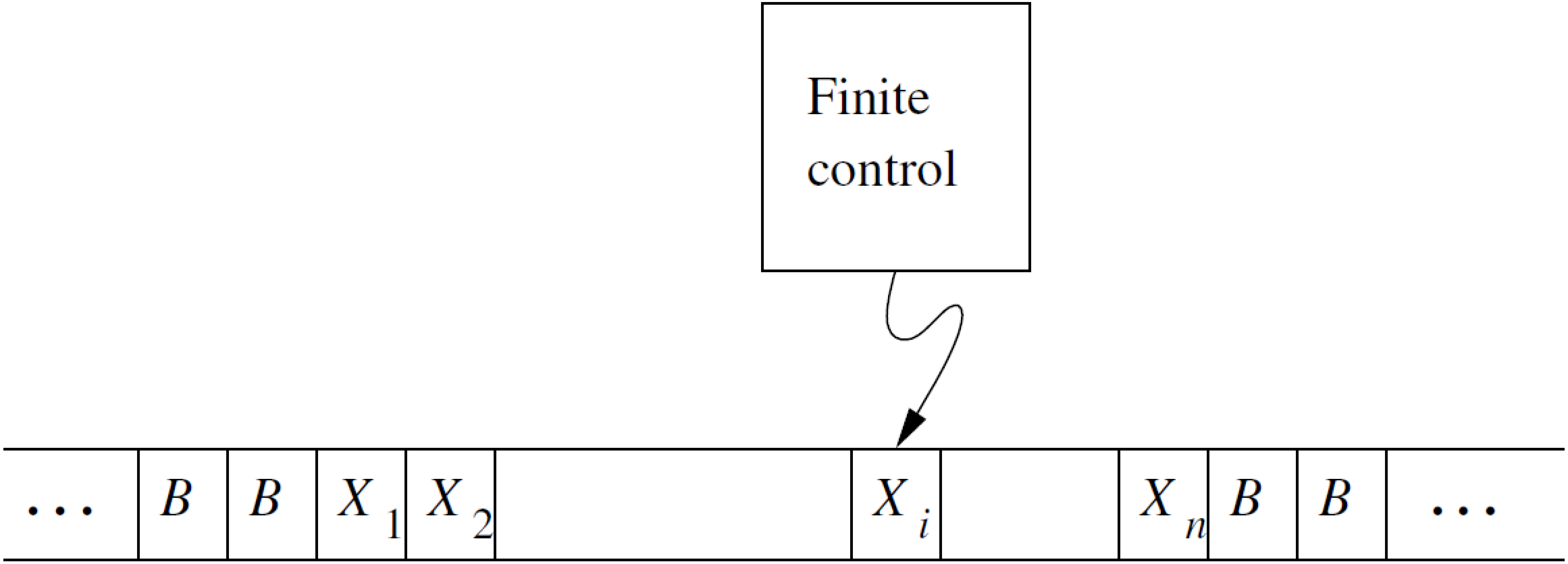
Prof. Sankhadeep Chatterjee

Turing Machine

A TM is described by:

- A finite set of states (Q).
- An input alphabet (Σ).
- A tape alphabet (Γ; contains Σ).
- A transition function (δ).
- A start state (q_0, in Q).
- A blank symbol (B, in Γ- Σ).
    - All tape except for the input is blank initially.
- A set of final states ($F \subseteq Q$).

Turing Machine



Transition Function of TM

- Takes two arguments:
  - A state, in Q.
  - A tape symbol in Γ.
- δ(q, Z) is either undefined or a triple of the form (p, Y, D).
  - p is a state.
  - Y is the new tape symbol.
  - D is a direction, L or R.
- If δ(q, Z) = (p, Y, D) then, in state q, scanning Z under its tape head, the TM:
  - Changes the state to p.
  - Replaces Z by Y on the tape.
  - Moves the head one square in direction D.
    - D = L: move left; D = R; move right.

Instantaneous Description of TM

- The following string represents a ID of a TM

$$X_1 X_2 \dots X_{i-1} q X_i X_{i+1} \dots X_n$$

1. q is the current state of TM
2. Tape head is scanning the i^{th} symbol from left
3. $X_1 X_2 \dots X_n$ is the portion of the tape between the leftmost and the right- most nonblank.

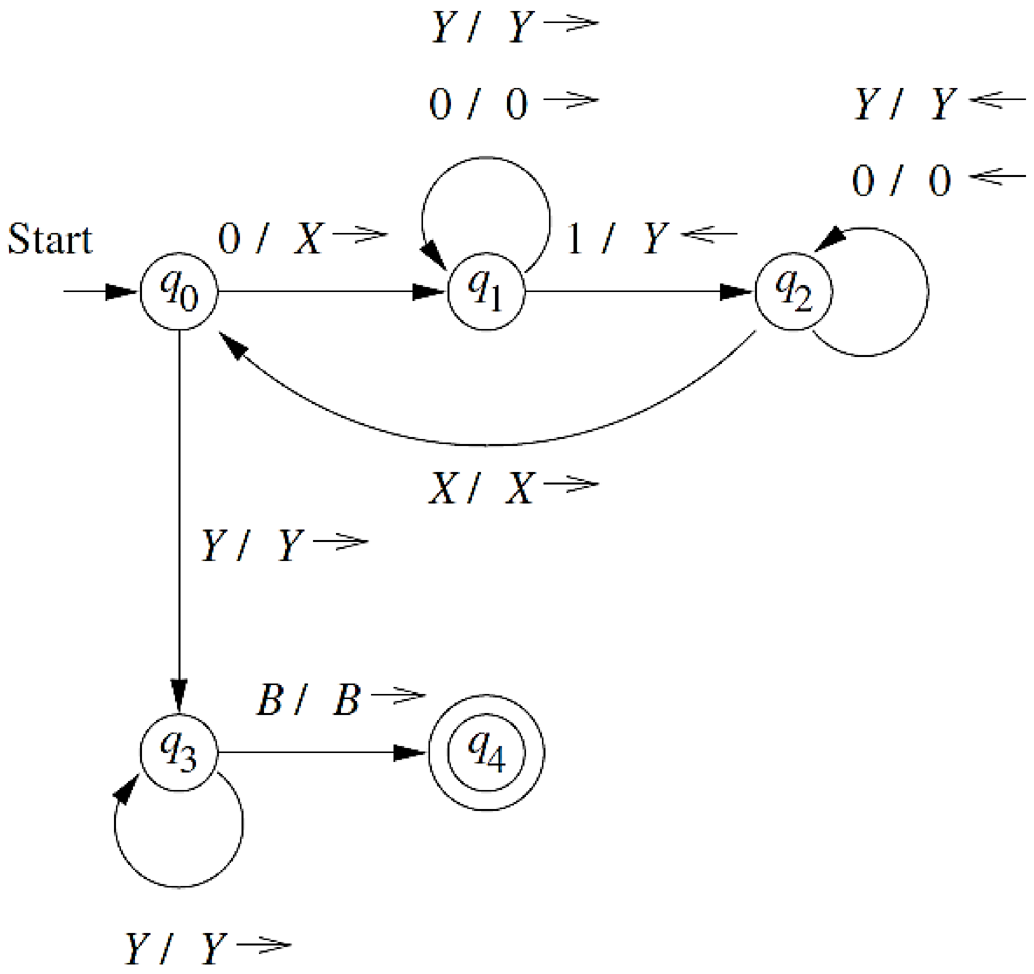
- Assume that $\delta(q, X_i) = (p, Y, L)$
- Then the operator $\vdash$ is used to represent one move of TM as follows:

$$X_1 X_2 \dots X_{i-1} q X_i X_{i+1} \dots X_n \vdash X_1 X_2 \dots X_{i-2} \boldsymbol{p} \boldsymbol{X_{i-1}} \boldsymbol{Y} X_{i+1} \dots X_n$$

Example: TM

$L = \{0^n 1^n | n \geq 1\}$

State	Symbol 0	1	X	Y	B
q_0	(q_1, X, R)	–	–	(q_3, Y, R)	–
q_1	$(q_1, 0, R)$	(q_2, Y, L)	–	(q_1, Y, R)	–
q_2	$(q_2, 0, L)$	–	(q_0, X, R)	(q_2, Y, L)	–
q_3	–	–	–	(q_3, Y, R)	(q_4, B, R)
q_4	–	–	–	–	–



Example: TM

- Consider input 0011 to the TM
- Initial ID is $q_0 0011$
- $q_0 0011 \vdash Xq_1 011 \vdash X0q_1 11 \vdash Xq_2 0Y1 \vdash q_2 X0Y1 \vdash Xq_0 0Y1 \vdash XXq_1 Y1 \vdash XXYq_1 1 \vdash XXq_2 YY \vdash Xq_2 XYY \vdash XXq_0 YY \vdash XXYq_3 Y \vdash XXYYq_3 B \vdash XXYYBq_4 B$

	Symbol				
State	0	1	X	Y	B
q_0	(q_1, X, R)	—	—	(q_3, Y, R)	—
q_1	$(q_1, 0, R)$	(q_2, Y, L)	—	(q_1, Y, R)	—
q_2	$(q_2, 0, L)$	—	(q_0, X, R)	(q_2, Y, L)	—
q_3	—	—	—	(q_3, Y, R)	(q_4, B, R)
q_4	—	—	—	—	—

Language of Turing Machine

Let M = $(Q, \Sigma, \Gamma, \delta, q_0, B, F)$ be a Turing machine. Then,

$$L(M) = \{w \in \Sigma^* \mid q_0 w \vdash* \alpha p \beta\}$$

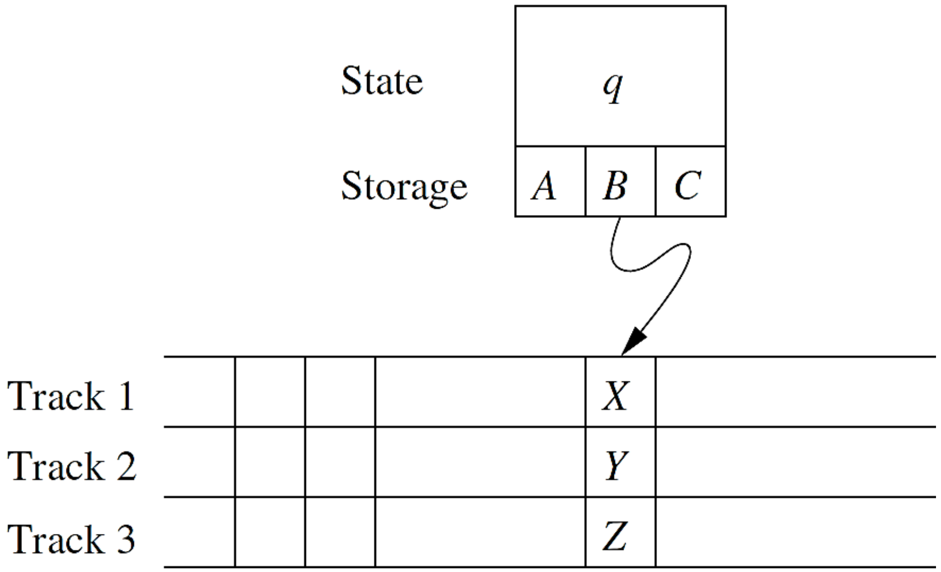
for some state p ∈ F and tape strings α and β

- The set of languages accepted by a Turing machine is often called the recursively enumerable languages or RE languages.
- **Acceptance by halting**: There is another notion of "acceptance" that is commonly used for Turing machines.
- We say a TM halts if it enters a state q, scanning a tape symbol X, and there is no move in this situation i.e. $\delta(q, X)$ is undefined.

Language of Turing Machine

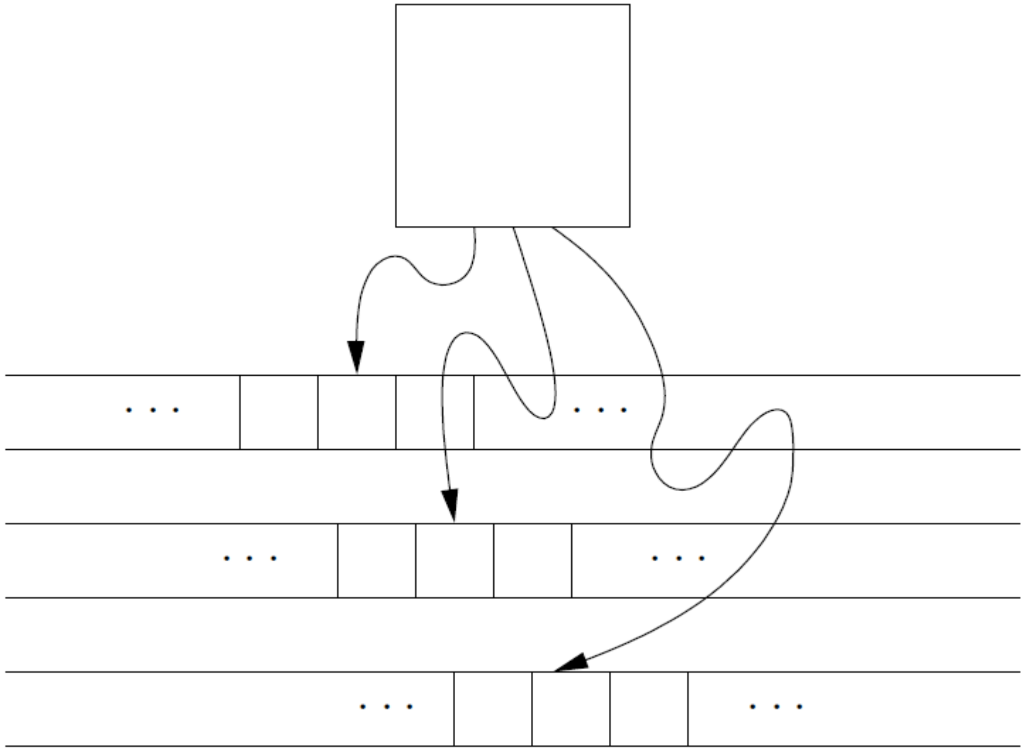
- We assume that a TM always halts when it is in an accepting state.
- However, TM may halt even if it does not accept.
- Those languages with Turing machines that do halt eventually, regardless of whether or not they accept, are called recursive language
- Turing machines that always halt, regardless of whether or not they accept, are a good model of an "algorithm"
- If an algorithm (or Turing Machine) to solve (or accept) a given problem (or language) exists, then we say the problem is "decidable"

Multiple Tracks Turing Machines



- State contains ‘control’ state and data elements
- The finite control consisting of not only a ‘control’ state q, but three data elements A, B, and C i.e. $[q, A, B, C]$
- The tape of the Turing machine is composed of several tracks.
- Each track can hold one symbol, and the tape alphabet of the TM consists of tuples, with one component for each “track”
- The cell scanned by the tape head contains the symbol $[X, Y, Z]$
- All other operations are same as that of the original TM

Multitape Turing Machines



- The device has a finite control (state), and some finite number of tapes.
- States and tapes are same as TM
- The input, a finite sequence of input symbols, is placed on the first tape.
- All other cells of all the tapes hold the blank
- The finite control is in the initial state. The head of the first tape is at the left end of the input. All other tape heads are at some arbitrary cell.
- A move of the multitape TM depends on the state and the symbol scanned by each of the tape heads.
- Tape heads makes a move, which can be either left, right, or stationary

Nondeterministic Turing Machines

- A nondeterministic Turing machine (NTM) differs from the deterministic variety in terms of transition function only.

- Transition function δ for each state q, tape symbol X is defined as follows:

$$\delta(q, X) = \{(q_1, Y_1, D_1), (q_2, Y_2, D_2), \ldots, (q_k, Y_k, D_k)\}$$