- Rejects an input (says NO) if ALL branches reject.
- Accepts if ANY one branch accepts.
- What does "nondeterminism" mean?

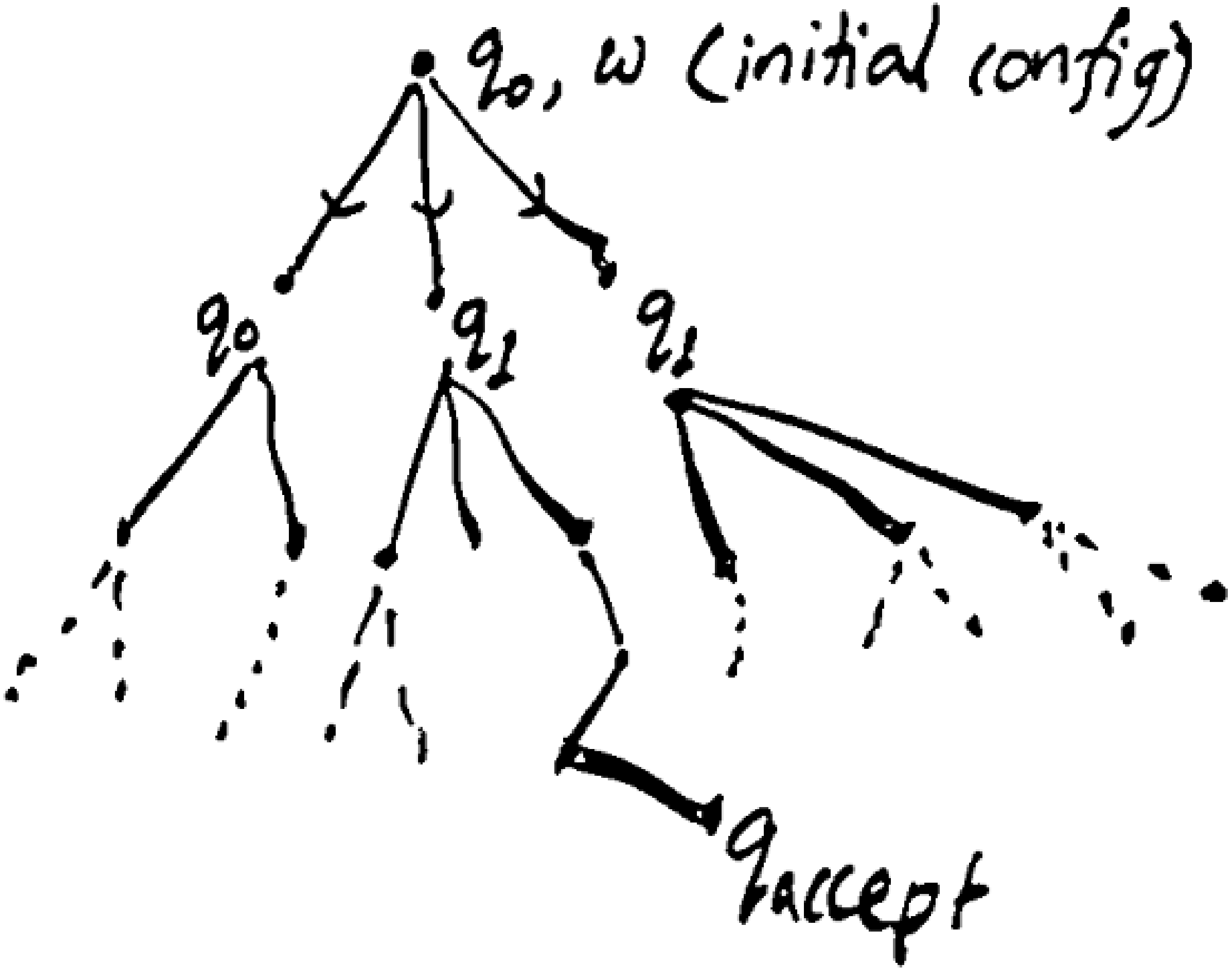
It "guesses" the next move.

Theorem: A non-deterministic TM is not 'more powerful' as far as the computing power is concerned. i.e, for every non-det TM N, there is an equivalent deterministic TM M, that recognizes (or decides) the same language.

- We prove it by simulating N on a deterministic TM M.
- For any input w:

  N accepts $w \Longleftrightarrow M$ accepts w.

- How do we simulate the moves of (non-deterministic) N on a deterministic M?
- What move(s) do we carry out first, which ones next?


q_0, w (initial config)
q_0
q_1
q_1
q_{accept}

- We execute moves in the order of a breadth-first traversal (bfs) of this tree.
- we may use a multitape TM as M. (we know no: of tapes do not matter)

Tape 1: Input (always)

ω	␣	␣

Tape 2: current branch of computation.

As a number, in binary.

1, 2, 3, (1,1), (1,2), (1,3), (2,1), (2,2)...

Tape 3: Actual computation, in this branch.

0	0	1	0	x	0	1	␣	␣	...

(SUDEEP)