Simulating NTMs

Use 3-tape TM

1) tape of original NTM
2) holds input string $w$
3) coin flips

If original NTM has some path that halts on $w$
our new deterministic machine has to halt on $w$ too
- need to find the right path (or the right coin flips to send us on that path)

Simulate NTM by
1) for each sequence of coin flips in lexicographic order
   write coin flips to tape 3
   simulate NTM step-by-step
   copy input to tape 1
   look at next coin flip(s) in case of nondeterminism
   from tape 2 to tape 1
   to decide what to do, give up if no flips left
   if any machine halts, halt simulation read at least 1, even if no nondeterminism
If \( \text{NTM} \ M \text{ decides } L \text{ in poly steps} \), is there also a standard \( \text{TM} \ M' \) that decides \( L \) in poly steps?

\[
NP = P ?
\]

\[
\uparrow \quad \left\{ \begin{array}{l}
\text{set of languages decidable in poly time on standard TM}
\end{array} \right.
\]

\[
\downarrow \quad \left\{ \begin{array}{l}
\text{set of languages decidable in poly time on NTM}
\end{array} \right.
\]

\[
\text{Set of languages } L \text{ s.t. } \exists \text{ standard } \text{TM}, s.t. w \in L \text{ if and only if there exists certificate } x \text{ s.t., } |x| \text{ is poly in } |w| \text{ and } M(x,w) = T.
\]
Universal Turing Machines

one machine that can simulate any other TM

UTM takes input \( M \), \( w \) and simulates behavior of \( M \) on \( w \)

3 types: encoding of \( M \) need to encode as a string

type of \( M \) current state of \( M \) encodes state, symbols

\[
\begin{align*}
\delta \in &
\begin{array}{lll}
(0, 0, \rightarrow) & 0 & 000 \\
(0, 1, \rightarrow) & 1 & 001 \\
(0, 2, \rightarrow) & 1 & 011 \\
(0, 3, \rightarrow) & 0 & 100 \\
(0, 4, \rightarrow) & 0 & 000 \\
(1, 0, \rightarrow) & 0 & 001 \\
(1, 1, \rightarrow) & 0 & 010 \\
\end{array}
\end{align*}
\]
"M": \[000a000a000a010; 001a000a100a000; \ldots\]

Input \( w = 010a \quad "w" = a001a000a100 \)

Min state \( q_0 \quad 000 \)

Simulate one step of \( M \) by

Finding a transition on tape 1 whose first two components match \( w \).

If none exist, then \( M \) was in a halting state, so halt.

One transition is found, copy 3rd component to state tape.

Copy 4th component to current location on 2nd tape unless 4th comp is \( 010 \) or \( a011 \) (\( \rightarrow \) or \( \leftarrow \)) — in these cases, scan 2nd tape for next or previous \( a \) (if see \( w \), write \( 0000 \) in its place).