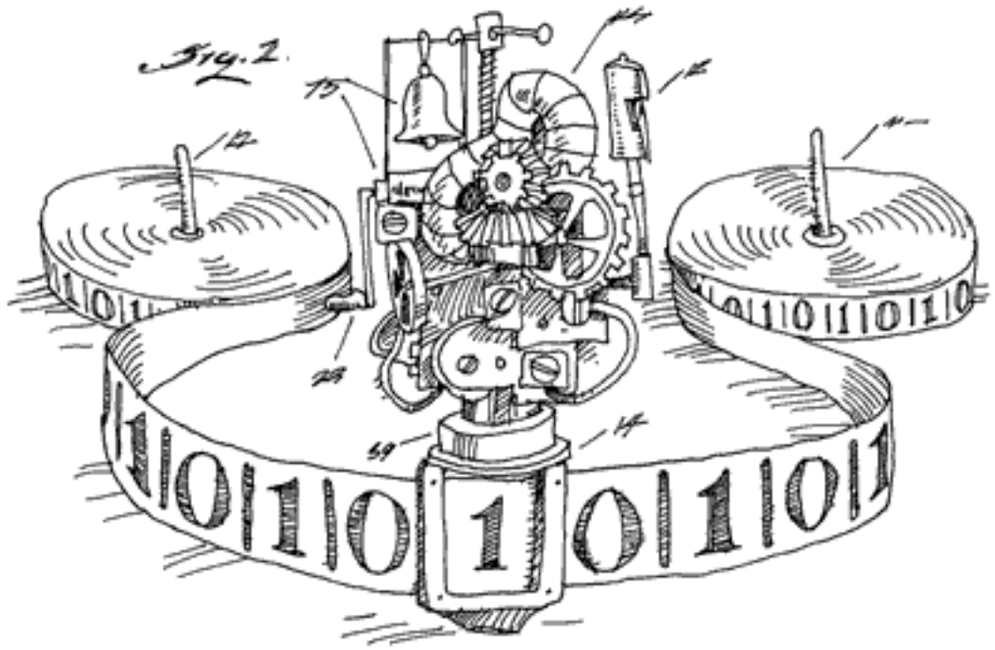
CSC4010: Introduction to Theoretical Computer Science



Lecture 17

Recap: Simulating Turing Machines as Circuits

Theorem: Let $T : \mathbb{N} \to \mathbb{N}$ be any time constructible function (time bound). Then, TIME(T) $\subseteq$ SIZE(T^2) . In particular, $P \subseteq$ PSIZE .

Key Observation:

Locality of computation: Given a Turing Machine M, let $B = b_1 b_2 \cdots b_l$ be the current configuration and let NEXT(B) $= b'_1 b'_2 \cdots b'_l$ then, b'_i depends on $b_{i-1}, b_i, b_{i+1}, b_{i+2}$.

Recap: Simulating Turing Machines as Circuits

Theorem: Let $T : \mathbb{N} \to \mathbb{N}$ be any time constructible function (time bound). Then, TIME(T) $\subseteq$ SIZE(T^2) . In particular, $P \subseteq$ PSIZE .

Let $M = (Q, \Sigma, q_{\text{acc}}, q_{\text{rej}}, q_0, \sqcup, \delta)$. Then, for any configuration $B = b_1 b_2 \cdots b_l$, where $b_i \in Q \cup \Sigma$. Suppose $|Q \cup \Sigma| = r$

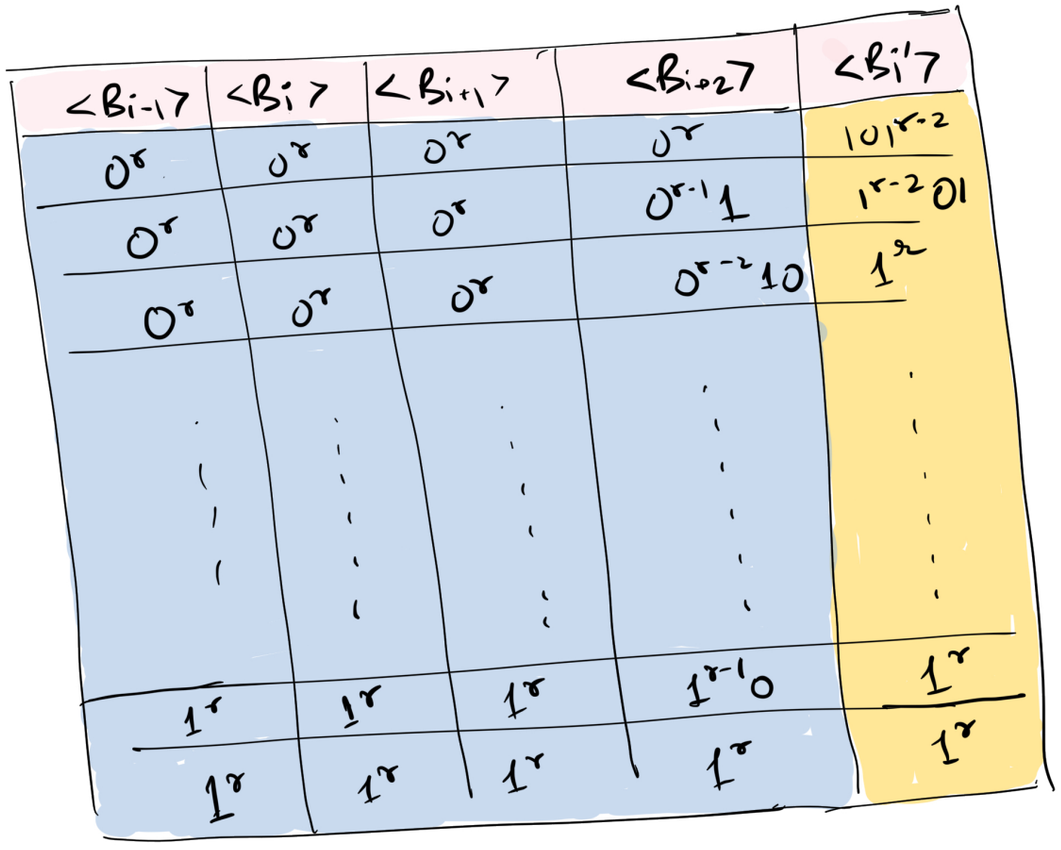
$\langle B \rangle = \langle b_1 \rangle \langle b_2 \rangle \cdots \langle b_l \rangle$ and $|\langle b_i \rangle| = r$.

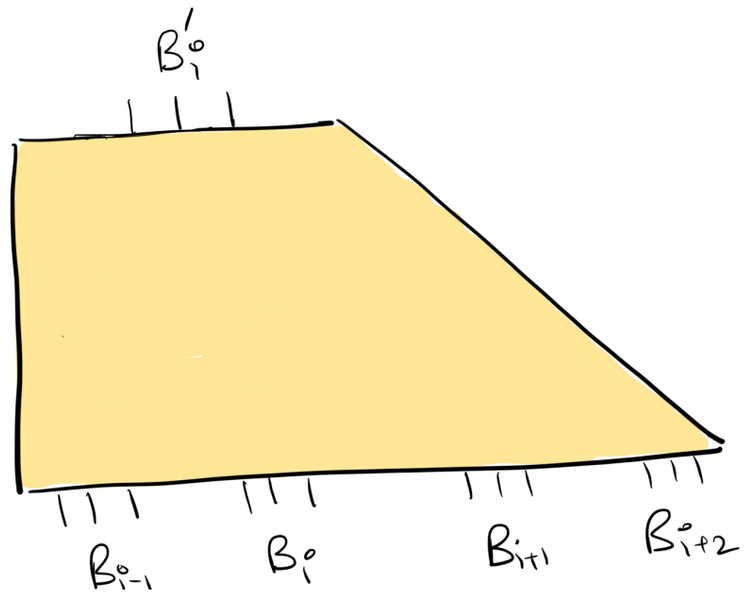
There is a function $\Delta_M : \{0,1\}^{4r} \to \{0,1\}^r$ such that for every configuration B, if $\langle B \rangle = b_1 b_2 \cdots b_l$ and $\langle \text{NEXT}(B) \rangle = b'_1 b'_2 \cdots b'_l$ then for every i, we have,

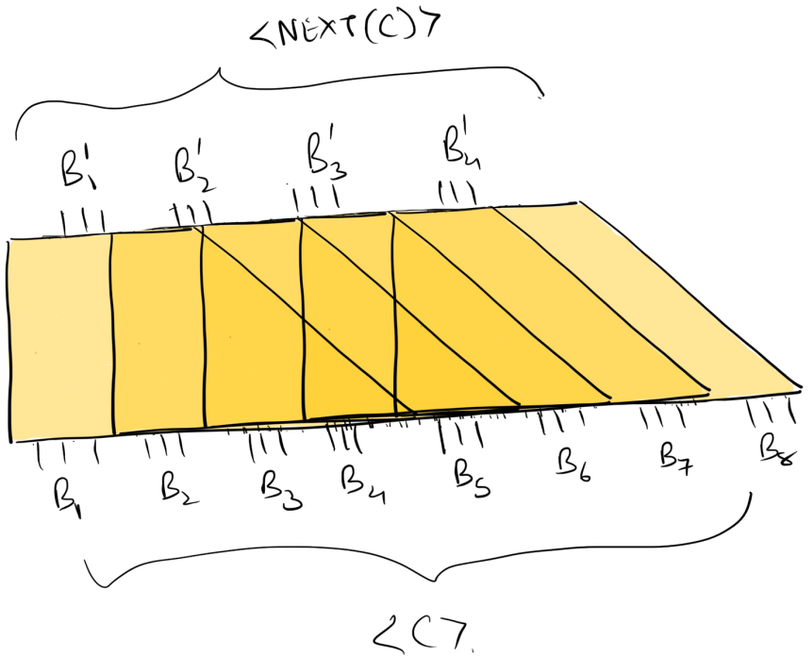
$$\Delta_M(b_{i-1} b_i b_{i+1} b_{i+2}) = b'_i$$

Recap: Simulating Turing Machines as Circuits

Time for computing the circuit for Δ_M:

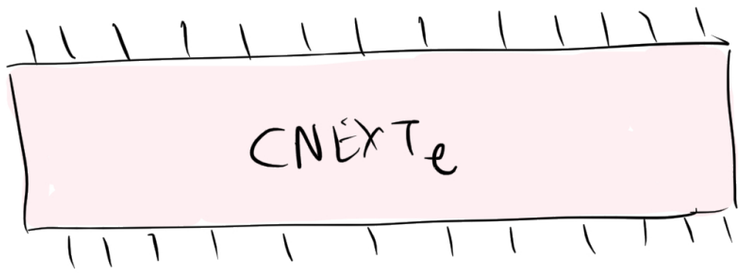




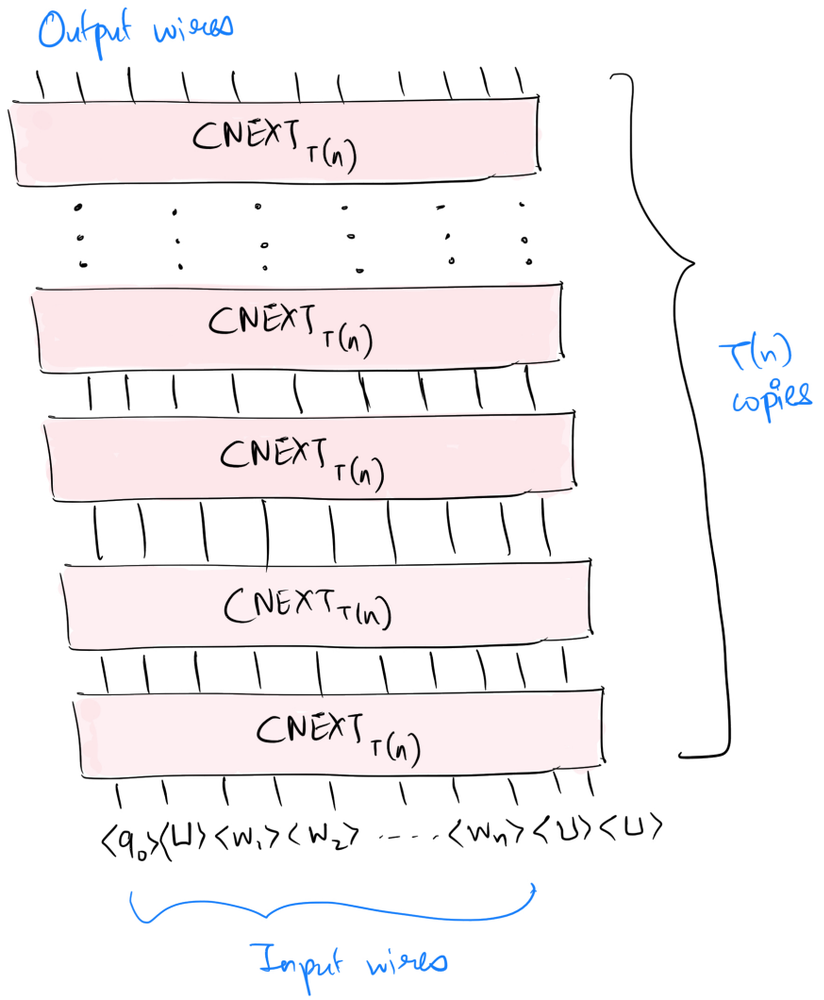


Recap: Simulating Turing Machines as Circuits

- Let M be a Turing Machine that decides L with time complexity $T(n)$.
- We get a circuit of size $O(T(n)^2)$ computing $L_n = \{0,1\}^n \cap L$.



Time for computing the circuit for L_n :



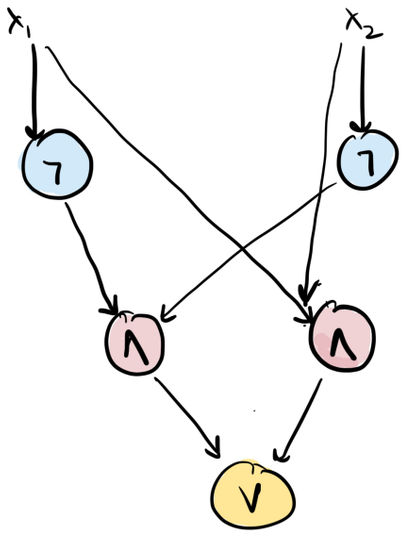
CIRCUIT-SAT is NP-Complete

Circuit Satisfiability

We say that circuit C (on n inputs) is satisfiable if there exists $x \in \{0,1\}^n$ such that $C(x) = 1$

CIRCUIT-SAT $= \{\langle C \rangle \mid \exists x \text{ s.t. } C(x) = 1\}$

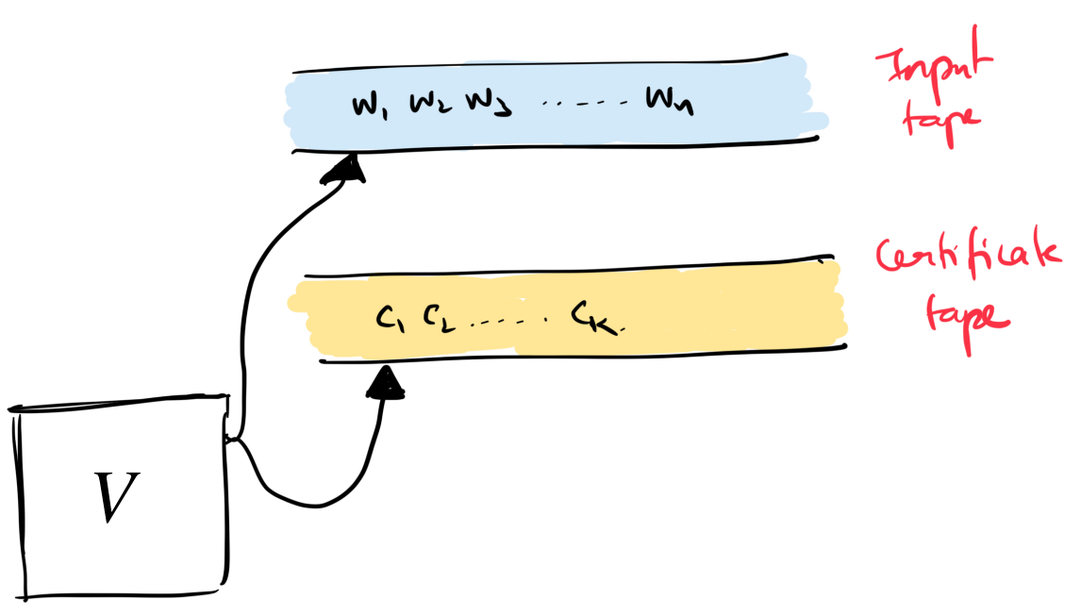
CIRCUIT-SAT $\in NP$



CIRCUIT-SAT is NP-Hard

CIRCUIT-SAT is NP-Hard

- Consider $Y \subseteq \{0,1\}^*$ such that $Y \in NP$:
  - There exists verifier V such that $w \in Y \iff \exists k_1, k_2 \geq 0$ such that $V(w, c) = 1$ for some $|c| \leq |w|^{k_1}$ and V runs in time at most $|w|^{k_2}$.



CIRCUIT-SAT is NP-Hard

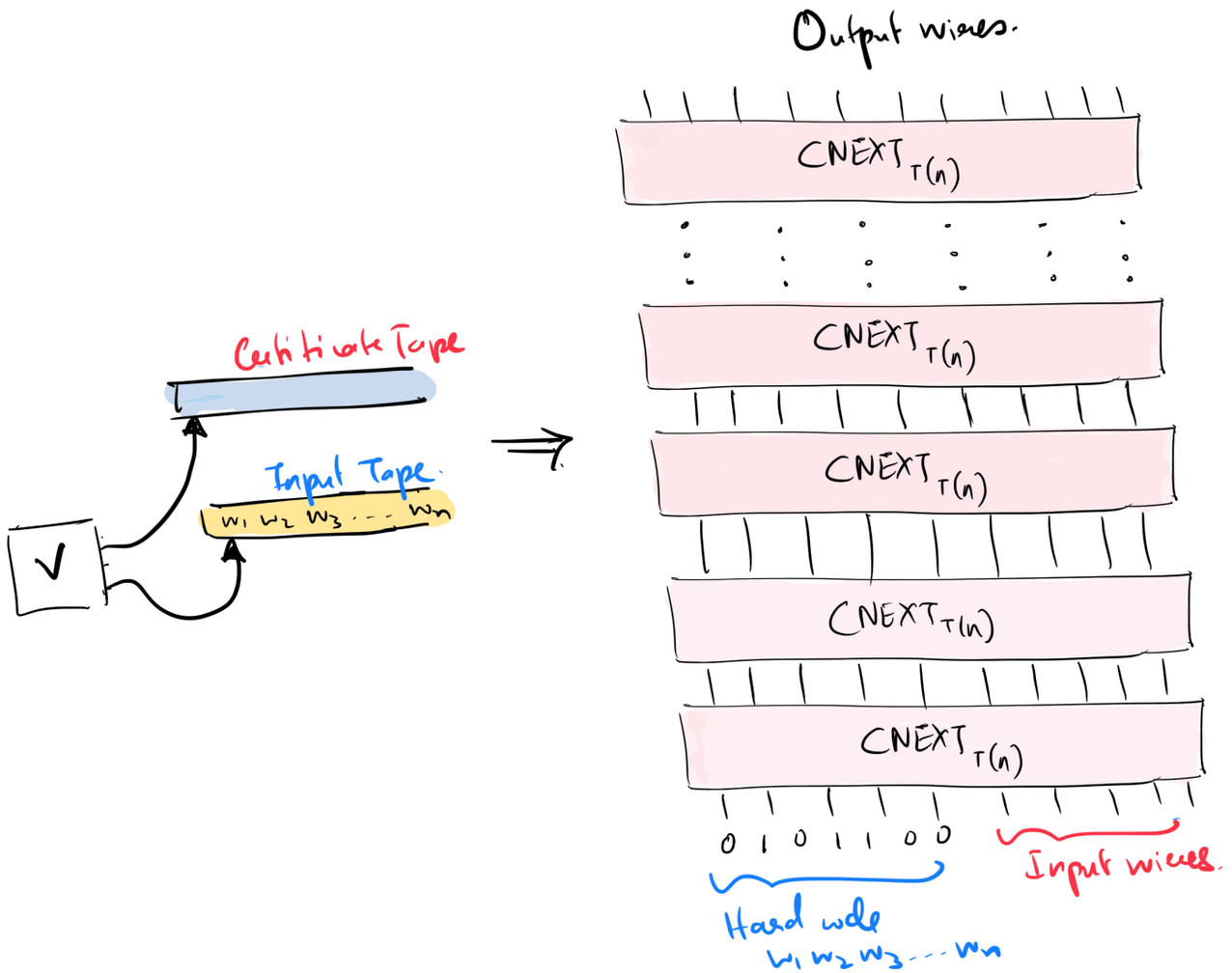
Theorem: CIRCUIT-SAT is NP-Hard.

- **Proof:** Given any $Y \in$ NP, show that $Y \leq_P$ CIRCUIT-SAT .

CIRCUIT-SAT is NP-Hard

Theorem: CIRCUIT-SAT is NP-Hard.

- **Proof:** Given any $Y \in$ NP, show that $Y \leq_P$ CIRCUIT-SAT .



Cook-Levin Theorem

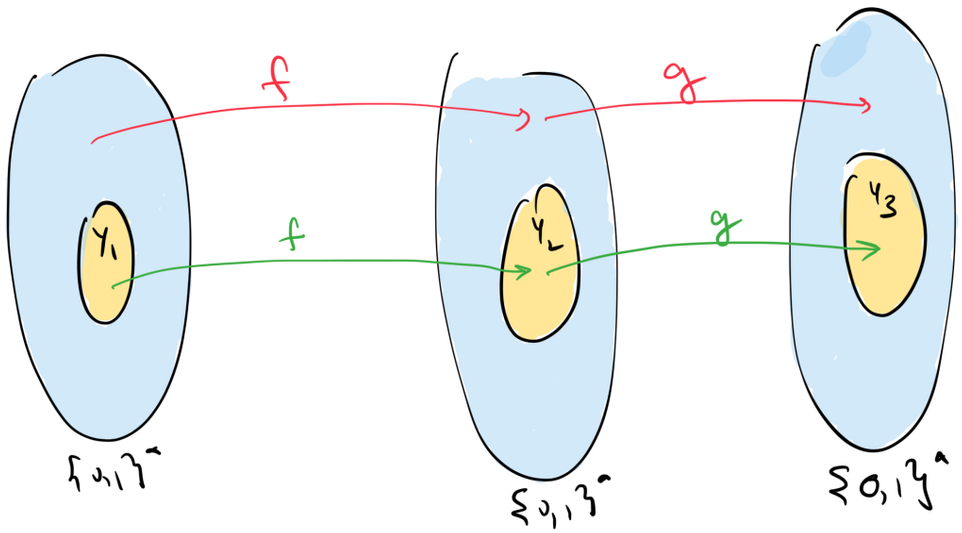
SATISFIABILITY

Definition: SAT $= \{\langle F \rangle \mid F$ is a satisfiable boolean formula$\}$.

Cook-Levin Theorem: SAT is NP-Complete.

Chaining Reductions

Claim: If $Y_1 \leq_P Y_2$ and $Y_2 \leq_P Y_3$ then $Y_1 \leq_P Y_3$.



SATISFIABILITY

Definition: SAT $= \{\langle F \rangle \mid F$ is a satisfiable boolean formula$\}$.

Cook-Levin Theorem: SAT is NP-Complete.

SATISFIABILITY

Definition: SAT $= \{\langle F \rangle \mid F$ is a satisfiable boolean formula$\}$.

Cook-Levin Theorem: SAT is NP-Complete.

Summary

- We saw some explicit NP-Complete Languages:
  - CIRCUIT-SAT is NP-Complete
  - SAT is NP-Complete
  - In HW, you will prove other languages such as 3SAT, CLIQUE, etc. are NP-Complete.