



CS 579: Computational Complexity. Lecture 4

Boolean Circuits

Alexandra Kolla

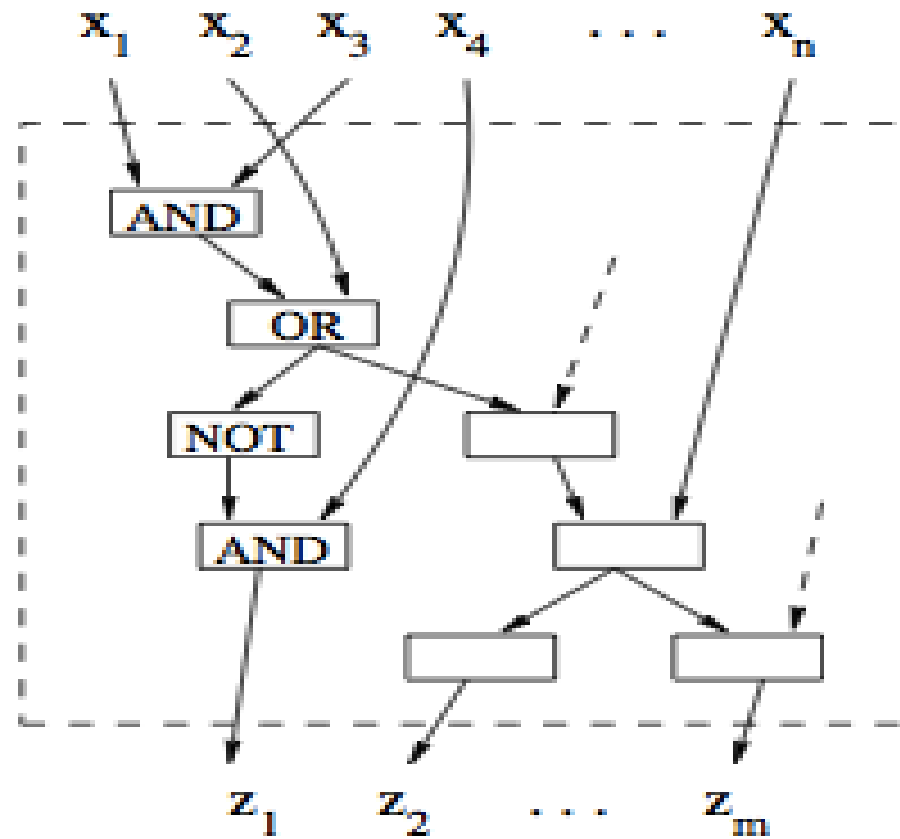
Today

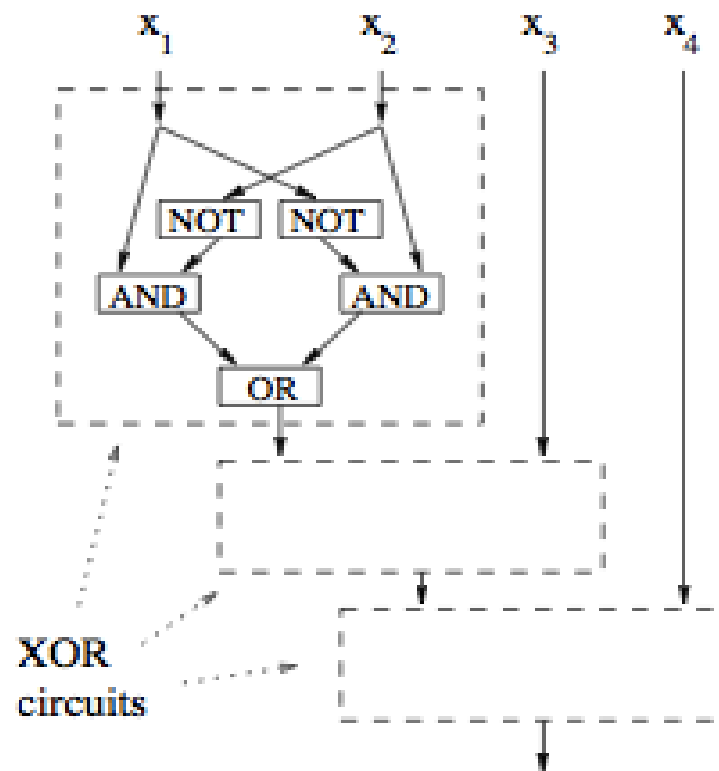
- Boolean circuits
- Poly size circuits can simulate poly computations
- Relations between complexity classes
- Karp-Lipton

Circuits

- Circuit C has n inputs, m outputs and is constructed with AND, OR, NOT gates.
- Each gate has in-degree 2 except the NOT gate which has in-degree 1
- Circuit C computes function $f_C: \{0,1\}^n \rightarrow \{0,1\}^m$
- $\text{SIZE}(C)$ =number of AND and OR gates (we don't count NOT gates)

Circuits





A circuit computing the boolean function $f_C(x_1x_2x_3x_4) = x_1 \oplus x_2 \oplus x_3 \oplus x_4$

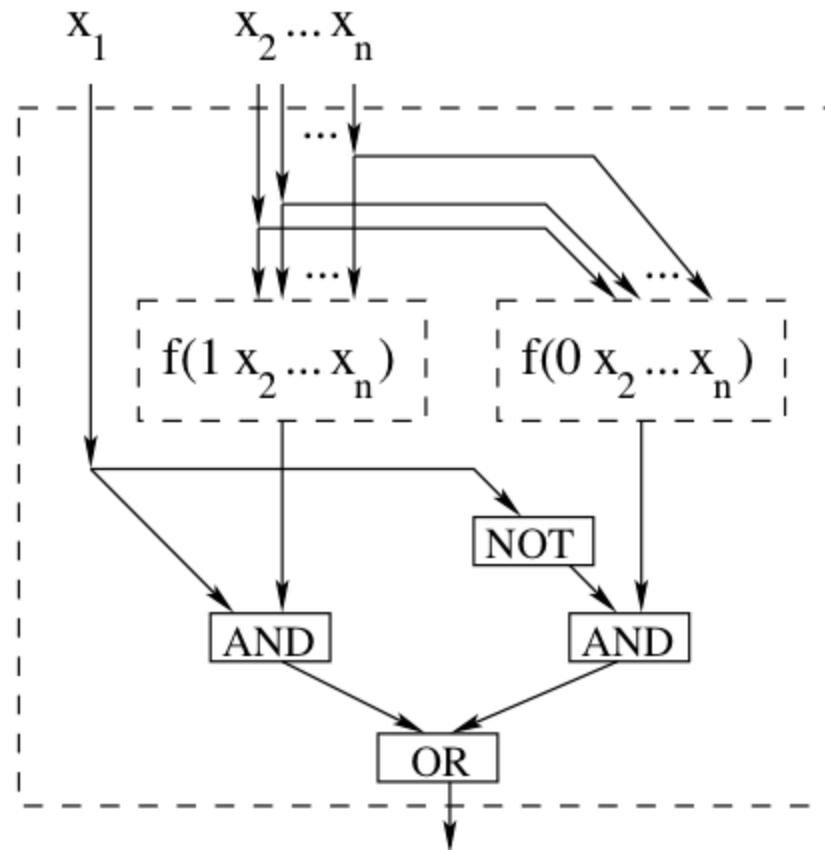
Circuits

- To be compatible with other complexity classes, need to extend the model to arbitrary input sizes:
- **Definition 1.** Language L is solved by a family of circuits $\{C_1, C_2, \dots, C_n, \dots\}$ if for every $n \geq 1$ and for every x s.t. $|x|=n$
$$x \in L \iff f_{C_n}(x) = 1$$
- **Definition 2.** Language $L \in \text{SIZE}(s(n))$ if L is solved by a family of circuits $\{C_1, C_2, \dots, C_n, \dots\}$ where C_i has at most $s(i)$ gates.

Relation to other complexity classes

- Unlike other complexity classes where there are languages of arbitrarily high complexity, the size complexity of a problem is always at most exponential
- **Theorem.** For every language L ,
$$L \in \text{SIZE}(O(2^n))$$

Relation to other complexity classes

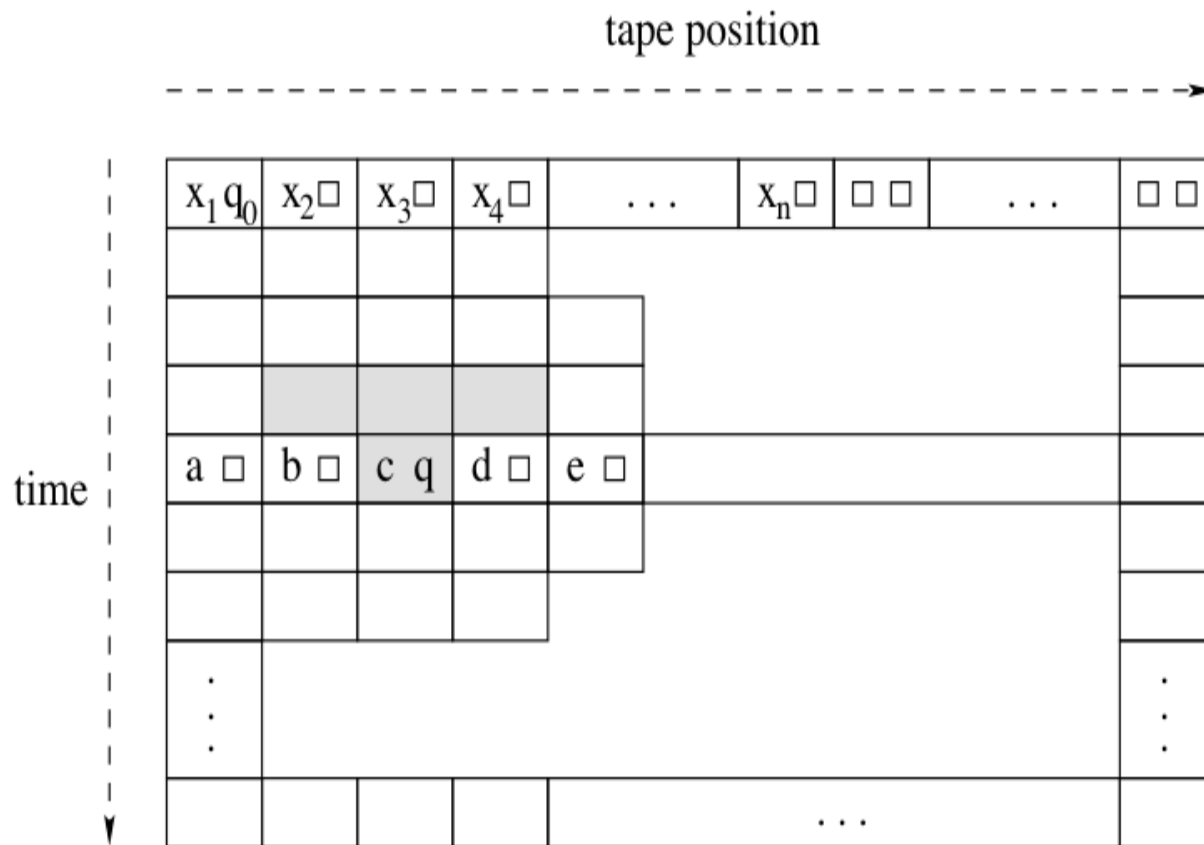


Relation to other complexity classes

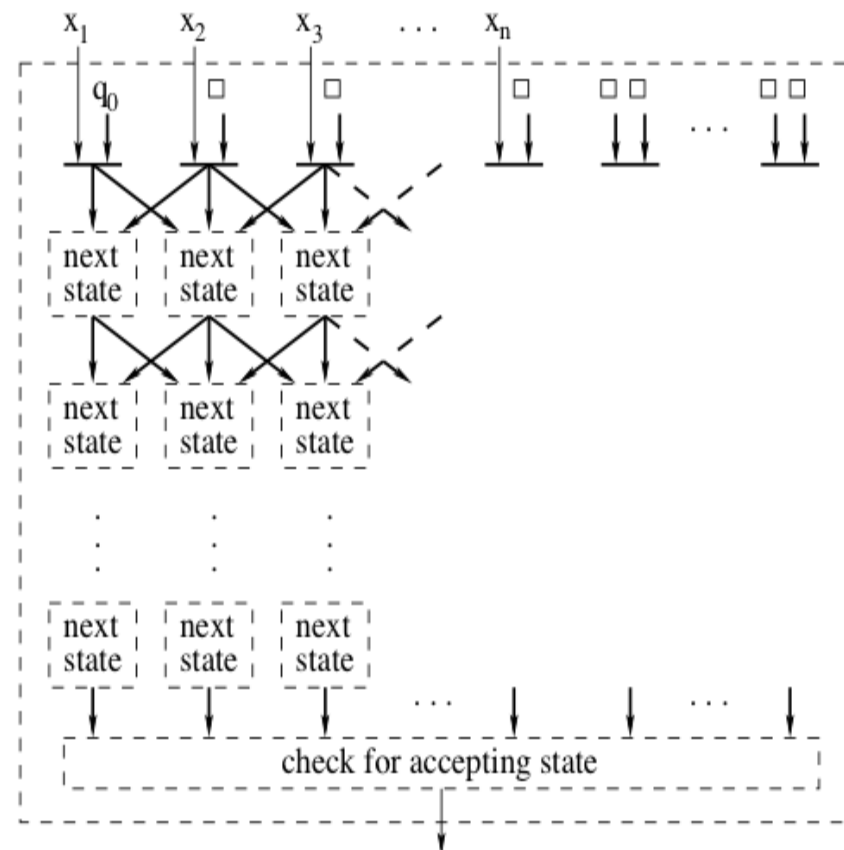
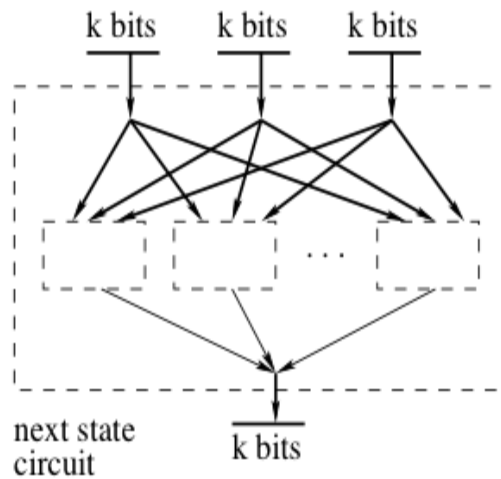
- Exponential bound is nearly tight
- **Theorem.** There are languages L such that $L \notin \text{SIZE}(2^{o(n)})$. In particular, for every $n \geq 11$, there exists $f : \{0,1\}^n \rightarrow \{0,1\}$ that cannot be computed by a circuit of size $2^{o(n)}$.

Relation to other complexity classes

- Efficient computations can be simulated by small circuits
- **Theorem.** If $L \in \text{DTIME}(t(n))$, then $L \in \text{SIZE}(O(t^2(n)))$



Relation to other complexity classes



Relation to other complexity classes

- Efficient computations can be simulated by small circuits
- **Theorem.** If $L \in \text{DTIME}(t(n))$, then $L \in \text{SIZE}(O(t^2(n)))$
- **Corollary.** $P \subseteq \text{SIZE}(n^{O(1)})$
- However, $P \neq \text{SIZE}(n^{O(1)})$. In fact, there are undecidable languages in $\text{SIZE}(O(1))$ (ex)

Karp-Lipton-Sipser

- **Theorem.** If $NP \subseteq SIZE(n^{O(1)})$ then $PH = \Sigma_2$

Karp-Lipton-Sipser

- **Theorem.** If $NP \subseteq SIZE(n^{O(1)})$ then $PH = \Sigma_2$

