Complexity Reading Group 10/22/2015
Brown University

1 Boolean Circuits

Definition 1: A *circuit* is a directed acyclic graph with n source nodes and 1 sink node. Each of the remaining nodes are logical gates. The size of the circuit is the number of total nodes

Definition 2: P_{poly} is the class of languages L, decidable by polynomial size circuit families

Alternatively, can think of P_{poly} as the class of Turing Machines with advice. Suppose:

$$L \in P_{poly} \text{ if } \exists \{\alpha_n\} \text{ with } \alpha_n \in \{0,1\}^{p(n)}, \text{ TM } M, \text{ s.t. } x \in L \iff M(x, \alpha_n) = 1 \tag{1}$$

Theorem 6.19 (Karp Lipton): If $NP \subseteq P_{poly}$, then $PH = \Sigma_2^P$. I.e.:

$$PH = \cup_i \Sigma_i^P = \cup_i \Pi_i^P \tag{2}$$

Proof: Note that it is sufficient to show that $\Pi_2^P = \Sigma_2^P$, which we can do by showing $\Pi_2\text{SAT} \in \Sigma_2^P$

Recall:

Definition 3: Let Π_2SAT be:

$$\phi \in \Pi_2 SAT \iff \forall_u \exists_v : \phi(u, v) = 1 \tag{3}$$

$(\phi, u) \in \Pi_2 SAT \iff \exists v : \phi(u, v) = 1$

Suppose $NP \subseteq P_{poly}$.

Then $\exists$ polynomial size $\{c_n\}$ to solve $\Pi_2 SAT$.

So we can generate a witness using this circuit family in linear time. I.e. $\exists \{c_{n'}\}$ which outputs the witness $c_{n'}(\phi, u) = v$, (where v is the witness).

Can rewrite this statement as:

$$\exists_w : w \in \{0,1\}^{q(n)^2} \forall_u : u \in \{0,1\}^n \phi(u, c_{n'}(\phi, u)) = 1 \tag{4}$$

Where $c_{n'}(\phi, u)$ generates the witness for ϕ.

So what we want to show is that Equation 4 is true iff $\phi \in \Pi_2 SAT$

Now, note that Equation 4 can be verified in Σ_2^P. (just look at the quantifiers) □

Note: If we show $NP \not\subseteq P_{poly} \rightarrow P \neq NP$

Q: What are the limitations on this model of computation?

Theorem 6.21:

$$\forall_n : \exists_f : f\{0,1\}^n \rightarrow \{0,1\} : \text{ can't be computed by a circuit } C \text{ s.t. } |C| = \frac{2^n}{10n} \tag{5}$$

Proof: Given a circuit of size $\leq S$, it can be represented by $\leq c * S^2$ bits.

If this is our restriction on circuit size, then we can have s^{3s} possible circuits (number of possible DAGs with in-degree 2).

Suppose $S = \frac{2^n}{10n}$, then clearly far fewer circuits than 2^{2^n}. □

I.e. you cannot compute a huge space of functions with polynomial circuits.

Theorem 6.23 (Non-Uniform Hierarchy Theorem):

$$\forall_T, T' : \mathbb{N} \rightarrow \mathbb{N} \text{ with } \frac{2^n}{n} > T'(n) > 10T(n) > n \tag{6}$$

I.e. $\textsc{Size}(T(n)) \subsetneq \textsc{Size}(T'(n))$

Intuition: With a larger circuit, you can compute strictly more functions.

1.1 Gate Elimination Method

Suppose we have a function f that we're trying to compute with a circuit.:

(1) Assigning variables in f to maintain properties of f.

(2) Eliminate gates in our circuit by storing variables, preserve same properties of f via the new circuit.

Definition 5: Let $Q_{2,3}^{(n)}$ be the class of functions where:

$$f : \{0,1\}^n \to \{0,1\} \in Q_{2,3}^{(n)} \iff \forall_{(x_i,x_j)}\ f \text{ has at least 3 distinct sub functions as } (x_i, x_j) \text{ range} \quad (7)$$

Also: $\forall x_i : \exists c_i$ s.t. $f_{x_i=c_i} \in Q_{2,3}^{(n-1)}$

Recursion bottoms out at $Q_{2,3}^{(3)}$, since $n = 3$ is the smallest input space for which there can be 3 distinct sub functions.

Example: $f_c^{(n)}(x_1, \ldots, x_n) = ((\sum_i x_i) \mod 3) \mod 2$, for $c \in \{0, 1, 2\}$.

Now we're going to prove a bound for this class of functions:

Theorem 9.3.2 (from John Hughes' book): If $f \in Q_{2,3}^{(n)}, C(f) \geq 2n-3$, where $C()$ is a circuit.

Proof:

Part One: f depends on each var x_i. □

Part Two: Some input vertex x_i has fan-out ≥ 2. (suppose we're dealing with a more general class of circuits now).

Consider a gate g that has the maximum possible length to the output. Since we're in a DAG, g has to be directly receiving input nodes. Suppose x_i and x_j are the variables that feed in to g., and they both don't feed anywhere else.

If we fix x_i, then there are only two possible sub functions left, which is a contradiction. Therefore, there must be at least one input variable that feeds in to more than one input node (i.e. fan-out ≥ 2).

By induction: $C(f_{n-1}) \geq 2(n-1) - 3$, $C(f_n) \geq 2(n-1) - 3 + 2 \geq 2n - 3$

2 NC and AC classes

Definition 6: $L \in NC^d$ if:

- L can be decided by poly size $\{C_n\}$
- Depth $\mathcal{O}log^d(n)$

- Each gate having bounded (2) fan in

Definition 7: $L \in AC^d$ if:

- L can be decided by poly size $\{C_n\}$
- Depth $\mathcal{O}log^d(n)$
- Each gate having unbounded fan in

Note: $NC^0 \subsetneq AC^0 \subsetneq NC^1$

And more generally: $NC^i \subseteq AC^i \subseteq NC^{i+1}$

Theorem 1 (Ajtai '83): $\textsc{Parity} \notin AC^0$