Lecture #24

CS170
Spring 2021

Lower Bounds

- Complexity of a problem P is a function $T(P)$ that measures its cost (time/memory/...) as a function $f(n)$ of its input size n
- An algorithm for P gives an upper bound on $f(n)$
  - Eg P = sorting, $T(P)$ = # comparisons
  - Insertion Sort $\Rightarrow$
  - Merge Sort $\Rightarrow$
- A lower bound for P is a proof that $f(n) = \Omega(g(n))$
  - Holds for any algorithm (in a class)
  -

Lower Bounds

- Recall NP-complete problems: 3SAT, ILP, ...
    - widely believed lower bound:
    - Best known lower bounds:
    -
- CS 172 preview
    - "Time Hierarchy Thm"

Examples of Lower Bounds
for specific classes of algorithms

1) Circuit Complexity:
   -

2) Cell Probe Model
   -

3) Branching Program
   -

4) Communication Complexity
   -

Circuit Complexity (11 13)

- Problem: Given $f : \{0,1\}^n \to \{0,1\}$, how big a circuit do you need to evaluate f?
  - Circuit = DAG of and, or, not gates
  - Size could be # wires, depth
- Ex $f : \{0,1\}^{10} \to \{0,1\}^{10}$ could multiply $x \cdot y$ where x = first 5 bits of input, y = last 5 bits
- What is known?
  -
  -
  -
  -
  -

Circuit Complexity (2/3)

- Problem: Given $f : \{0,1\}^n \to \{0,1\}$, how big a circuit do you need to evaluate f?
  - Circuit = DAG of and, or, not gates
  - Size could be # wires, depth
- Most $f : \{0,1\}^n \to \{0,1\}$ need exponentially many wires

Circuit Complexity (313)

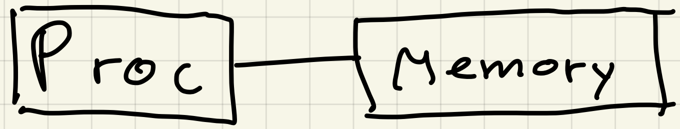
- Problem: Given $f:\{0,1\}^n \to \{0,1\}$, how big a circuit do you need to evaluate f?
  - Circuit = DAG of and, or, not gates
  - Size could be # wires, depth
- Connection to NP-Completeness
  -

  -

  -

  -

Cell Probe Model

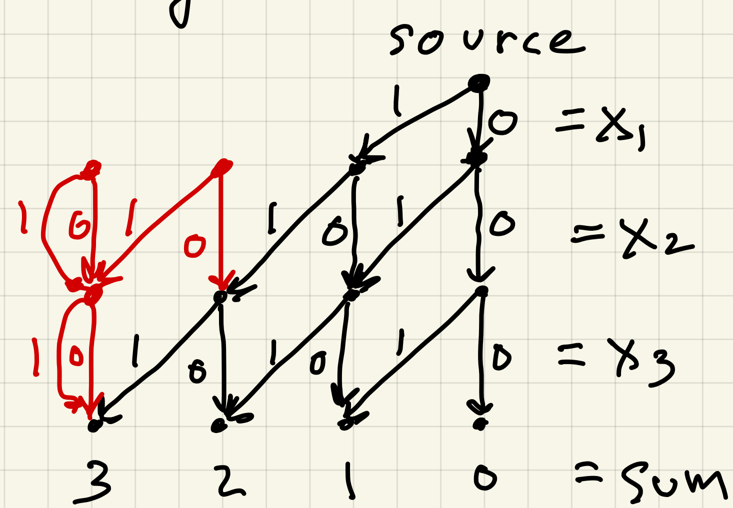
- Algorithm is allowed to perform the following ops:
    - 

    - Processor can read a word (w bits) from memory location i, or write a word
    -
-

Branching programs

- DAG to compute $f:\{0,1\}^n \to Y$
  - one source node
  - one sink node per element of Y
  - each non-sink nodes has 2 outgoing edges labelled $x_i=0$ or $x_i=1$, where x=input

- Adding 3 bits: $Y=\{0,1,2,3\}$



Width $=4=2^{\#\text{bits needed}}$

#Layers $=3=$ #steps to compute answer $\approx$ runtime

-

Communication Complexity (1/8)

- Alice and Bob both want $f:(X,Y) \to \{0,1\}$
    - Alice only knows X, Bob only knows Y
    - They exchange messages, last one to receive a message announces $f(X,Y)$
    - Goal: minimize #bits Alice and Bob exchange
- Different kinds of algorithms allowed:
    -
    -
    -
    -

Communication Complexity (2/8)

- Alice knows X, Bob knows Y, want $f(X,Y) \in \{0,1\}$ while minimizing # bits exchanged
- Def: $D(f)$ = minimum # bits with deterministic alg
- Def: $R_{pub}(f)$ = minimum # bits with public coin randomness
- Def: $R_{priv}(f)$ = minimum # bits with private coin randomness
-
-
-
-

Communication Complexity (3/8)

- Alice knows X, Bob knows Y, want $f(X,Y) \in \{0,1\}$
- Def: $\vec{D}(f)$ = minimum #bits with deterministic alg where Alice sends one message to Bob

Communication Complexity (4/8)

- Alice knows X, Bob knows Y, want $f(X,Y) \in \{0,1\}$
- Def: $\vec{D}(f)$ = minimum #bits with deterministic alg where Alice sends one message to Bob
- Claim: $\vec{D}(EQ) \geq n = |X|$
- Def: DE = counting # distinct elements
- Claim: Any exact deterministic alg A for DE requires $\Omega(n)$ bits of memory (FM was random)

proof: Show that if A solves DE with s bits, we can use A to solve EQ with $s + \log n$ bits, so

$$s + \log n \geq \vec{D}(EQ) \geq n \Rightarrow s \geq n - \log n \stackrel{?}{=} \Omega(n)$$

Communication Complexity (5/8)

- Claim: Any exact deterministic alg A for DE requires $\Omega(n)$ bits of memory
-

Communication Complexity (6/8)

- Claim: Any approximate deterministic alg for DE ($|\tilde{t}-t| \le .01t$) also requires $\Omega(n)$ bits of memory

Proof: EQ where $X, Y \in B \subseteq \{0,1\}^n$, $D^{\rightarrow}(EQ^B) \ge \log_2 |B|$

Claim: (no proof!) $\exists$ B such that

$|B| \ge 2^{cn}$, all $X \in B$ have same #1s $= r$,

and if $X, Y \in B$, $X \ne Y$, then #1s X and Y share $\le \frac{r}{10}$

Show that if A_{ap} solves DE with s bits, 1% error, we can use A_{ap} to solve EQ^B with s bits

$\Rightarrow$ $s \ge D^{\rightarrow}(EQ^B) \ge \log_2 |B| \ge cn = \Omega(n)$

Communication Complexity (7/8)

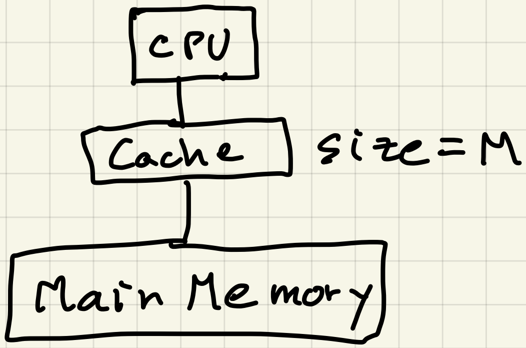
- Summary of counting distinct elements (DE)

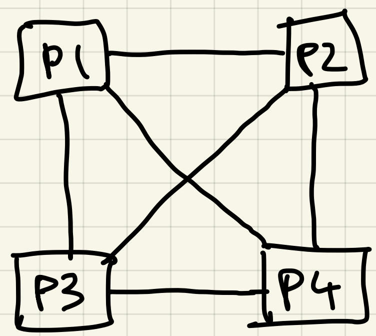
  No exact, deterministic alg with $o(n)$ memory

  No approx, deterministic alg with $o(n)$ memory

Communication Complexity (8/8)

- Goal: minimize communication between main memory and cache, or between processors connected over a network





- Thm (Hong, Kung, 81) Any execution of $\Theta(n^3)$ matrix multiply moves $\Omega(n^3/\sqrt{M})$ words between Cache and main memory
- Attained by "looptiling", widely implemented
- Extends to rest of linear algebra, any code that looks like nested loops accessing arrays