

DD 2441 Seminars on Theoretical Computer Science

LECTURE 1

Course practicalities

Jakob Nordström jakobn@kth.se
 www.csc.kth.se / ~ jakobn

Massimo Lauria lauria@kth.se
 www.csc.kth.se / ~ lauria

Course webpage

~~www~~.csc.kth.se/utbildning/kth/kurser/DD2441/semteor2

Look at course webpage regularly for updates
 (especially during first weeks)

This will be a demanding course

- Requires work in between lectures
- Requires reading up on stuff you know but have forgotten
- Requires you to fill in some details (won't have time to do everything in class)

Need to decide on

- schedule
- scribing
- and possibly more (Piazza.com for student-instructor interaction)

Will do this tomorrow Now focus on CC.

General communication problems

System needs to perform some task

Depends on info distributed among different parts
 of system (processors / parties / players)

Thus, players need to communicate to solve task.

(2)

Simplified abstract scenario

Yao's model '79

- Two parties / players (Alice & Bob)
- Each player gets fixed part of input
- Only care about communication
- Compute prespecified function of input

Very simple model, but applicable in many scenarios (as we shall see)

X, Y, Z arbitrary finite sets

$f: X \times Y \rightarrow Z$ arbitrary function

Alice gets $x \in X$ Bob $y \in Y$

Want to compute $f(x, y)$

Need to communicate

Use fixed protocol P (depending only on f)
for sending bits to each other

Protocol P must specify

- communication terminated?
- if so, what is the answer? ($f(x, y)$)
- if not, who sends bit next? [depends only on bits communicated so far]
- what is next bit sent [depends on communication so far + private input]

Only interested in communication
 ignore internal computations of Alice & Bob
 i.e., unlimited computational powers

Cost of P on $(x, y) = \# \text{ bits communicated}$

Cost of $P = \text{worst case, i.e., max over all } (x, y)$

Communication cplx of $f = \text{cost of best protocol for } f$

[In general, function will be

$$f_n: X_n \times Y_n \rightarrow Z_n \text{ for } n \rightarrow \infty$$

and interested in asymptotic analysis. This usually suppressed from notation]

Formally

DEF 1 Protocol P over domain $X \times Y$ with range Z : binary tree. Each internal node v labelled by $a_v: X \rightarrow \{0, 1\}$ or $b_v: Y \rightarrow \{0, 1\}$. Each leaf labelled by $z \in Z$.

Value of P on $(x, y) = \text{element } z \text{ in leaf reached, walking left if } a_v / b_v \text{ gives } 0 \text{ and right otherwise}$

Cost of P on $(x, y) = \text{length of path}$

Cost of $P = \text{max over all } (x, y)$

DEF 2 (Deterministic) communication cplx of $f: X \times Y \rightarrow Z$ is min cost of any P computing f .

Notation $D(f)$ [or $CC_D(f)$]

Variants

- ① Only one of players knows answers
[changes cplx by at most $\log_2 |Z|$]
NOT IMPORTANT
- ② Order of communication arbitrary or take turns
[factor of 2]
NOT IMPORTANT
- ③ function vs. relation
e.g. for sets $x, y \subseteq U$, find $z \in x \cap y$ & will return to this later]
- ④ # players 2 or more
[will return to this later]
- ⑤ Must Alice & Bob be deterministic or can they flip coins [return to this in 2lec]

Prop 3 For any $f: X \times Y \rightarrow Z$

$$D(f) \leq \log_2 |X| + \log_2 |Z|$$

$$D(f) \leq \log_2 |Y| + \log_2 |Z|$$

One of players sends over input, the other the answer

(5)

Ex 4 subsets $x, y \subseteq \{1, 2, \dots, n\}$

Compute $AVG(x, y) =$ arithmetic mean of multiset $x \cup y$

Alice sends $\sum_{a \in x} a \leq n^2$

$$\leq \log n^2 = 2 \log n \text{ bits}$$

and $|x| \leq n \quad \log n \text{ bits}$

Bob sends back total sum and total # elems, say.

Hence $D(AVG) = O(\log n)$.

Ex 5 subsets $x, y \subseteq \{1, 2, \dots, n\}$

$MED(x, y) =$ median of multiset $x \cup y$.

If $t = 2k$ elements, say median is k th smallest

$$(a) \quad D(MED) \leq n + \log n$$

By Prop 3

$$(b) \quad D(MED) = O(\log^2 n)$$

Binary search. Maintain interval $[i, j] \ni$ median

From start $i=1, j=n$.

Look at $k = (i+j)/2$

Alice sends #elems $\leq k$ and #elems $> k$

Now Bob knows if median $\leq k$ or median $> k$

(6)

Bob sends this info to Alice 1 bit.

Both update i, j on their arr.

Then do Next ~~round~~ stage

Interval halved each ~~round~~ ^{stage} $\Rightarrow \log_2 n$ ^{stages} ~~rounds~~

$O(\log n)$ communication per ~~round~~ stage.

Obviously careful look at details needed to get binary search right — ignore for now

(c) Get down to $O(\log n)$

Assume wlog $|x| = |y| = 2^i$ for some i

Alice and Bob exchange $|x|$ and $|y|$ and pad with appropriate number of 1 and 0.

$O(\log n)$ communication.

At each stage Alice has $x' \leq x$ s.t. median ^{can be in} x'
" Bob " $y' \leq y$ " " y'

Initially $x' = x, y' = y$

Halve the sizes in every stage.

Alice sends $a = \text{median of } x'$

Bob sends $b = \text{median of } y'$

If $a < b$: Alice throws away smallest half of x'

Bob " " largest " " y'

$a > b$: Opposite

$a = b$: Then $a = b$ is median.

7

Finally $|x'| = |y'| = 1$

Then smaller number is median

Why? Median must be in $[a, b]$ if $a < b$

In each stage, equally many elements $\leq a$ and $\geq b$ gets thrown away
so median stays the same.

$|x'|$ & $|y'|$ halved in every stage

$\Rightarrow O(\log n)$ stages

$O(\log n)$ communication in each stage.

Total $O(\log^2 n)$.

Now optimize ...

Two observations

1) Alice & Bob can exchange most significant bits of a & b until know which is larger.

2) In next stage, median is in $[a, b]$, so all the bits in which a & b agreed need not be sent again.

So each of $\log_2 n$ coordinates need not be exchanged more than once

$\Rightarrow$ Total comm cplx $O(\log n)$

Maybe skip if we don't have time.

Ex 6 Given (fixed) graph G on n vertices.

(8)

Alice receives clique C in G

Bob receives independent set I

$$\text{cis}(C, I) = |C \cap I|$$

function $2^{\{1, \dots, n\}} \times 2^{\{1, \dots, n\}} \rightarrow ?$

function to $\{0, 1\}$

$$\text{Prove } D(\text{cis}_G) = O(\log^2 n)$$

Protocol in $\log n$ stages

$O(\log n)$ communication / stage.

Invariant Alice & Bob maintain $V' \subseteq V$
s.t. $C \cap I \subseteq V'$.

Initially $V' = V$.

In each stage:

1) Alice looks for $u \in C \cap V'$ s.t.

$$|(\{u\} \cup N(u)) \cap V'| \leq |V'|/2$$

send "0" if doesn't exist

else send "1" + index of u $\log n$ bits

- 2) If Bob gets u , he checks if $u \in I$
 If so, send "1" [$C \cap I \neq \emptyset$]
 Else send "0"

Both players set $V' = V' \cap (\{u\} \cup N(u))$
 without communication and start next ~~send~~ stage

- 3) If Bob instead gets "0", he looks
 for $w \in I \cap V'$ s.t.

$$|(\{w\} \cup N(w)) \cap V'| > |V'|/2$$

If no such w , Bob sends "0"
 to say that $C \cap I = \emptyset$

Else Bob sends "1" plus index of w

- 4) If Alice gets w ,
 she outputs "1" if $w \in C$
 [i.e. $C \cap I \neq \emptyset$]

Else she sends "0"

Both players set

$$V' = V' \setminus N(w)$$

(so that V' contains only w and non-neighbours
 of w in V')

without communication and start next stage.

Claim A V' shrinks by at least half in each stage (so $\log_2 |V|$ stages).

True in (2) and also in (4).

Claim B Always $C \cap I \subseteq V'$. Suppose $C \cap I = \{v\}$
True from beginning ($V' = V$)

If Alice sends $u \neq v$ then $v \in N(u)$
so stays in V'

If Bob sends $w \neq v$ then $v \notin N(w)$
so stays in V'

Claim C If protocol outputs "1" this is correct.
Obvious

Claim D If protocol outputs "0" [done by Bob in (3)] then this is correct.

For all $v \in C \cap V'$ have $|N(v)| < |V'|/2$ by 1)

For all $v \in I \cap V'$

$$|N(v) \cap V'| \geq |V'|/2$$

Hence $C \cap I$ can't possibly be non-empty

— would give v with

$$|V'|/2 \leq |N(v) \cap V'| < |V'|/2$$

Contradiction.

Ex 7

(11)

$x, y \in \{0, 1\}^n$ i.e. n -bit binary strings

$EQ(x, y) = 1$ if $x = y$, 0 otherwise

Trivial protocol uses $n+1$ bits.

Is it possible to do better?

Will return to this question next lecture.