INF 554: Using randomness in algorithms **Autumn 2014**

Lecture 1 — September 15th, 2014

Lecturer: Frédéric Magniez *Scribe: Thibault GROUEIX*

1.1 Exercises

1.1.1 Test of commutativity

Problem

Input

- Group G
- Function $\circ$ gives the product of two elements.
- n elements of G h_1 ... h_n and H, the generated group.

Output ACCEPT iff H is a commutative group

Complexity Number of operations involving $\circ$.

Naive solution Checking all possible couples $(i, j) \in G^2$ requires $(O(n^2)$ operations.

Algorithm

- Draw k, l with Sampling at random from H.
- Accept iff. $k \circ \mathrm{l} = \mathrm{l} \circ \mathrm{k}$

Complexity One call to Sampling, two calls to $\circ$

Performance One sided error :

- H commutative group -> ACCEPT always
- H non commutative group -> ACCEPT with probability inferior to 3/4

Property H is commutative iff $\forall i, j; h_i \circ h_j = h_j \circ h_i$ (the generators commute)

Lemma 1.1. *if H is not commutative then*

- $\exists i, j$ *for which* $h_i \circ h_j \neq h_j \circ h_i$
- $\underset{k,\, l \in H}{\mathbb{P}} (k \circ l \neq l \circ k) \geq 1/4$

Proof: the center of H is defined as follow :

$$Z(H) = \{k \in H | \forall l \in H, k \circ l = l \circ k\}$$

Moreover,

$$S(k) = \{l \in H | k \circ l = l \circ k\}$$

So if $S(k) = H$ then $k \in Z(H)$.
Z(H) is a strict sub group of H, so using the Lagrange theorem :

$$|Z(H)| \leq 1/2|H|$$

Therefore,

$$\mathbb{P}_{k \in H}(k \in Z(H)) \leq 1/2$$

Let $k \in H \backslash Z(k)$
Also, $S(k)$ is a strict sub group of H,

$$|S(k)| \leq 1/2|H|$$

Therefore,

$$\mathbb{P}_{l \in H}(l \in S(k)) \leq 1/2$$

Finally,

$$\mathbb{P}_{k,l \in H}(k \circ l \neq l \circ k) = \mathbb{P}_{k,\, l \in H}(k \circ l \neq l \circ k \cap k \notin Z(H))$$

$$\mathbb{P}_{k,\, l \in H}(k \circ l \neq l \circ k) = \mathbb{P}_{k,\, l \in H}(k \notin Z(H) \cap k \notin S(k))$$

$$\mathbb{P}_{k,\, l \in H}(k \circ l \neq l \circ k) = \mathbb{P}_{k,\, l \in H}(k \notin Z(H)) \cdot \mathbb{P}_{k,l \in H}(k \notin S(k) | k \notin Z(H))$$

$$\mathbb{P}_{k,\, l \in H}(k \circ l \neq l \circ k) \geq 1/2 \cdot 1/2$$

$$\mathbb{P}_{k,\, l \in H}(k \circ l \neq l \circ k) \geq 1/4$$

CQFD

$\square$

Note We have demonstrated that the algorithm is a Monte-Carlo algorithm, with a one sided error of 1/4.
We still have to explain the Sampling process, which choose randomly $k, l \in H$

Weak Sampling

- Draw r uniformly from $\{0, 1\}^n$
- Calculate and return $h_1^{r_1} \cdot ... \cdot h_n^{r_n}$

Example

- $n = 4$
- draw 0110
- return $h_2 \cdot h_3$

Complexity n group operation $\circ$

Lemma 1.2. *if K is a strict subgroup of G, and $h_1 ... h_n$ the generators of K then*

$$\mathop{\mathbb{P}}_{h\ with\ Weak\ Sampling}(h \in K) \leq 1/2$$

Proof: Since $K \neq G$, $\exists\ i$ such as $h_i \notin G$
Remember : we draw r uniformly from $\{0,1\}^n$.
$h = h_1^{r_1} \cdot ... \cdot h_n^{r_n}$
Fix r except r_i
Suppose $i = 1$
$\beta = h_2^{r_2} \cdot ... \cdot h_n^{r_n}$
$\mathop{\mathbb{P}}_{r_i \in \{0,1\}}(h \in K) = \mathop{\mathbb{P}}_{r_i \in \{0,1\}}(h_1^{r_1} \cdot \beta \in K)$

- $\beta \in K$
  - $r_1 = 1 \Rightarrow h_1 \cdot \beta \notin K$ (if $h_1 \cdot \beta \in K$ then $h_1 \in K \Rightarrow$ Contradiction)
  - $r_1 = 0 \Rightarrow h = \beta \in K$

  @ $\mathop{\mathbb{P}}_{r_i \in \{0,1\}}(h \in K | \beta \in K) = 1/2$
- $\beta \notin K$
  - $r_1 = 1 \Rightarrow$ unknown
  - $r_1 = 0 \Rightarrow h = \beta \notin K$

  @ $\mathop{\mathbb{P}}_{r_i \in \{0,1\}}(h \in K | \beta \notin K) \leq 1/2$

Consequently

$$\mathop{\mathbb{P}}_{r_i \in \{0,1\}}(h \in K|) \leq 1/2$$

To generalize to all i, take

$$i = min\{j, h_j \notin K\}$$

Set $\eta = h_1^{r_1} \cdot ... \cdot h_(j-1)^{r_{j-1}}$ and adapt the proof
Finally, using the conditional probability on $r_1, ..., r_n$ except r_j, we reach the conclusion :

$$\mathop{\mathbb{P}}_{r \in \{0,1\}^n}(h \in K) \leq 1/2$$

CQFD

□

Complexity Finally the algorithm requires $O(n)$ against the $O(n^2)$ of the naive method.

1.1.2 Determinist Algorithm to Probabilistic Algorithm

Problem

Input

- Deterministic algorithm P which return the product of two matrix

$$\mathop{\mathbb{P}}_{A,B\ matrix\ modulo\ N}(P(A,B) \neq A \cdot B) \leq 1/9$$

- δ

Output Probabilistic algorithm P_2 such as

$$Err(P) = \mathop{\mathbb{P}}_{A,B\ matrix}(P_2(A,B) \neq A \cdot B) \leq \delta$$

Complexity The algorithm must require $O(n^2)$ additions/multiplications and $(O(log\delta))$ call to P

Questions

1. Prove that $\forall$ R,S matrix modulo N

$$A \cdot B = (A - R) \cdot (B - S) + (A - S) \cdot S + R \cdot (B - S) + R \cdot S (*)$$

2. Deduce the probabilistic algorithm
3. Write an algorithm which requires $O(n^2)$ additions/multiplications and $(O(log\delta))$ call to P such as
   - **If** $Err(P) = 0]$ **then** $\mathbb{P}(Algo\ accept) = 1$
   - **If** $Err(P) \leq 1/11$ **then** $\mathbb{P}(Algo\ accept) \geq 1 - \delta$
   - **If** $Err(P) \geq 1/9$ **then** $\mathbb{P}(Algo\ reject) \geq 1 - \delta$
   - **If** $1/11 \leq Err(P) \leq 1/9$ **then** non specified

Answers

1. Develop
2. - Choose uniformingly at random two matrix R and S modulo N
   - Calculate and return (*) with P

Proof:

$$\begin{aligned}\mathbb{P}\left((*) \neq A \cdot B\right) = \mathbb{P}(&P(A-R, B-S) \neq (A-R)\cdot(B-S)) \\ &\text{or } (P(A-R, S) \neq (A-R)\cdot S) \\ &\text{or } (P(R, B-S) \neq R \cdot (B-S)) \\ &\text{or } (P(R,S) \neq R \cdot S))\end{aligned}$$

$$\mathbb{P}\left((*) \neq A \cdot B\right) \leq 4Err(P)$$

$$\mathbb{P}\left((*) \neq A \cdot B\right) \leq 4/9$$

□

Note Choosing randomly R and S is the same thing as choosing randomly A-R and B-S because we work modulo N

3.
   - Do k times :
     - Choose A,B randomly
     - Verify $P(A, B) = A \cdot B$ with Friedvalds algorithm done multiple times
     - $X_i = 0$ if there is no mistake
       $X_i = 1$ if any mistake
   - **if** $\sum X_i \leq \frac{k}{10}$ **then ACCEPT**
     else REJECT

Proof: Try using the Chernoff Bound □