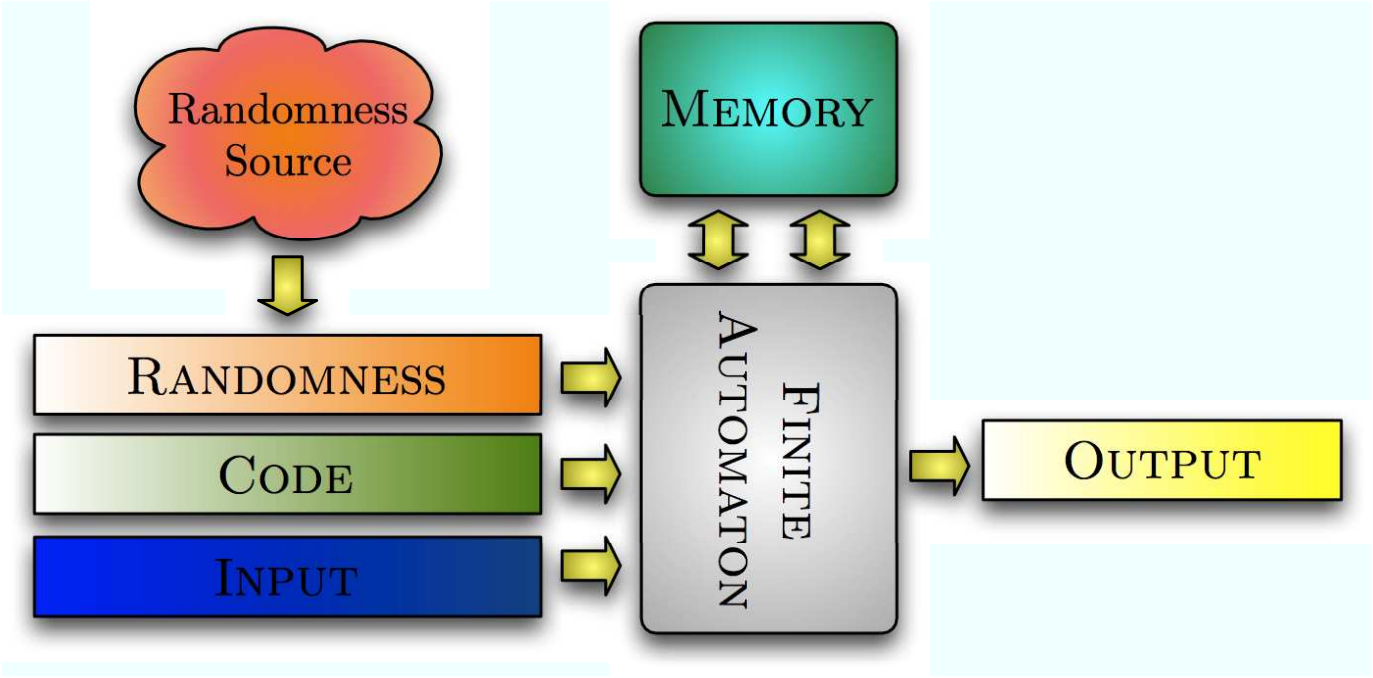
MTAT.07.003 Cryptology II

Analysis of Randomised Algorithms

Sven Laur
University of Tartu

Models of Computation

Conceptual description of a computing device



- ▷ Code is not part of the computing device.
- ▷ Randomness is not part of the computing device.
- ▷ Other details depend on the exact model of computations

Standard models of computation

Universal Turing Machine

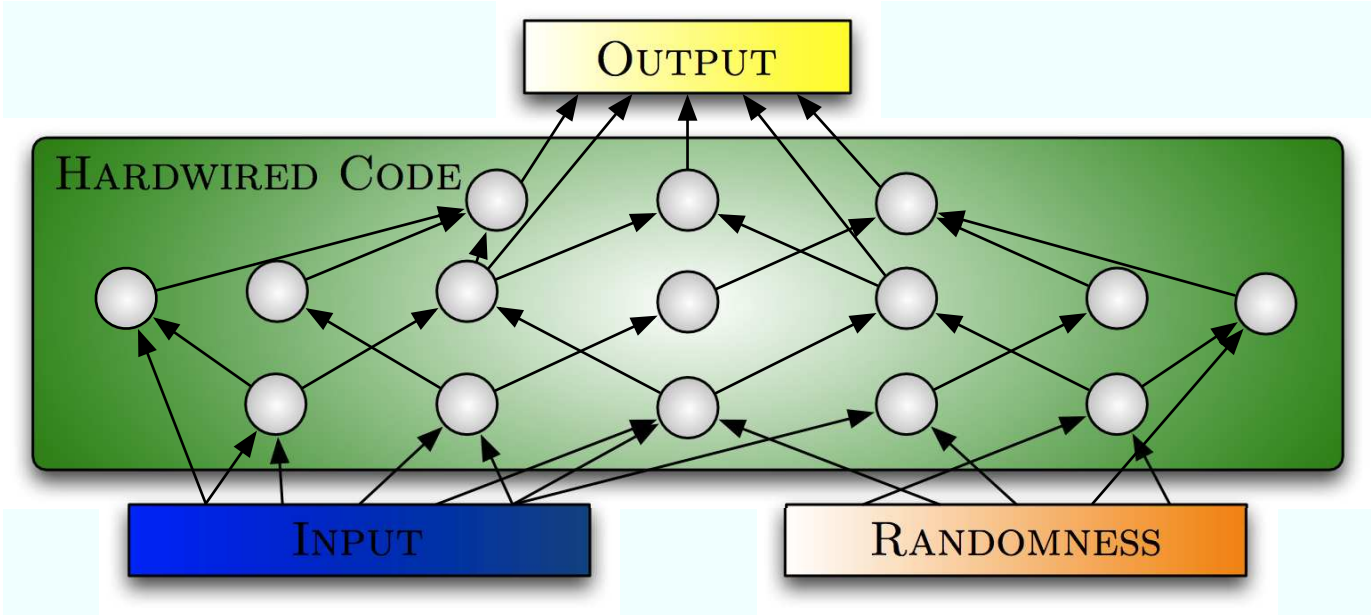
- ◇ Takes in a code ϕ and inputs $x_1, \ldots, x_n$.
- ◇ The random tape $\omega \in \{0,1\}^*$ is filled with fair coin tosses.
- ◇ Jumps in memory and in code costs $\Theta(n)$ where n is address.
- ◇ Programmed by filling the table of configurations and reads.

Universal Random Access Machine

- ◇ Takes in a code ϕ and inputs $x_1, \ldots, x_n$.
- ◇ The random tape $\omega \in \{0,1\}^*$ is filled with fair coin tosses.
- ◇ Jumps in memory and in code costs $\Theta(\log n)$ where n is address.
- ◇ Programmed in modified assembly language (Generalised Intel assembly).

Yet another model of computation

A finite time computations can be represented as Boolean circuits

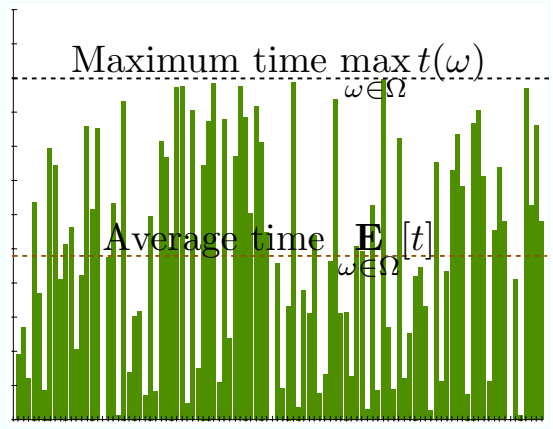


- ▷ No explicit calls to memory. Memory is in-lined to the circuit.
- ▷ No explicit branching. Possible choices must be in-lined to the circuit.

Time-complexity

Let $\mathcal{A}$ be a randomised algorithm and let $t(x, \omega)$ denote the number of elementary steps that are needed to obtain the output $\mathcal{A}(x, \omega)$.

Then for each input we can define average and maximum running time.

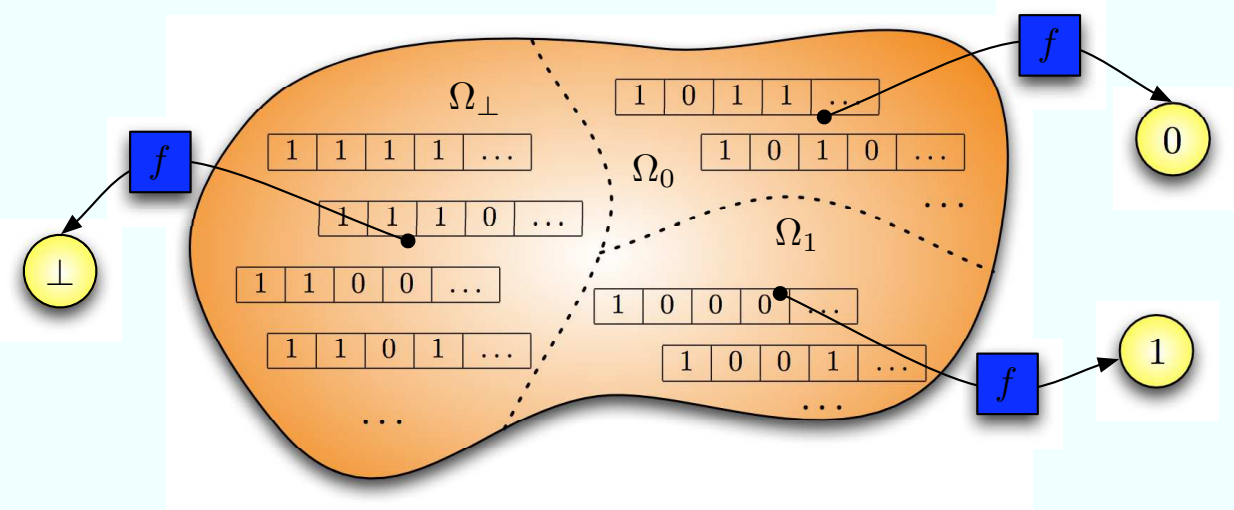


These estimates of running time are defined analogously for sets of inputs.

Finally, we can consider a *t-time algorithm* $\mathcal{A}$ that is halted after t elementary steps. The corresponding invalid output is denoted by $\perp$.

Discrete random variable and its sample space

A *discrete random variable* f is a function $f : \Omega \to \{0,1\}^*$ that maps each *non-deterministic choice* $\omega \in \Omega$ to a concrete output $f(\omega)$.



- ▷ A *sample space* Ω consists of all non-deterministic choices.
- ▷ An *elementary event* $\Omega_y = \{\omega \in \Omega : f(\omega) = y\}$ consists of all non-deterministic choices that lead to the same output y.

Observable events and probability measure

A *probability measure* is determined by the likelihoods for all elementary events. The assignment can be arbitrary as long as their sum is one.

	$\Omega_\perp$	Ω_0	Ω_1	Ω_{00}	Ω_{01}	$\ldots$	$\sum$
$\Pr$	$p_\perp$	p_0	p_1	p_{00}	p_{01}	$\ldots$	1

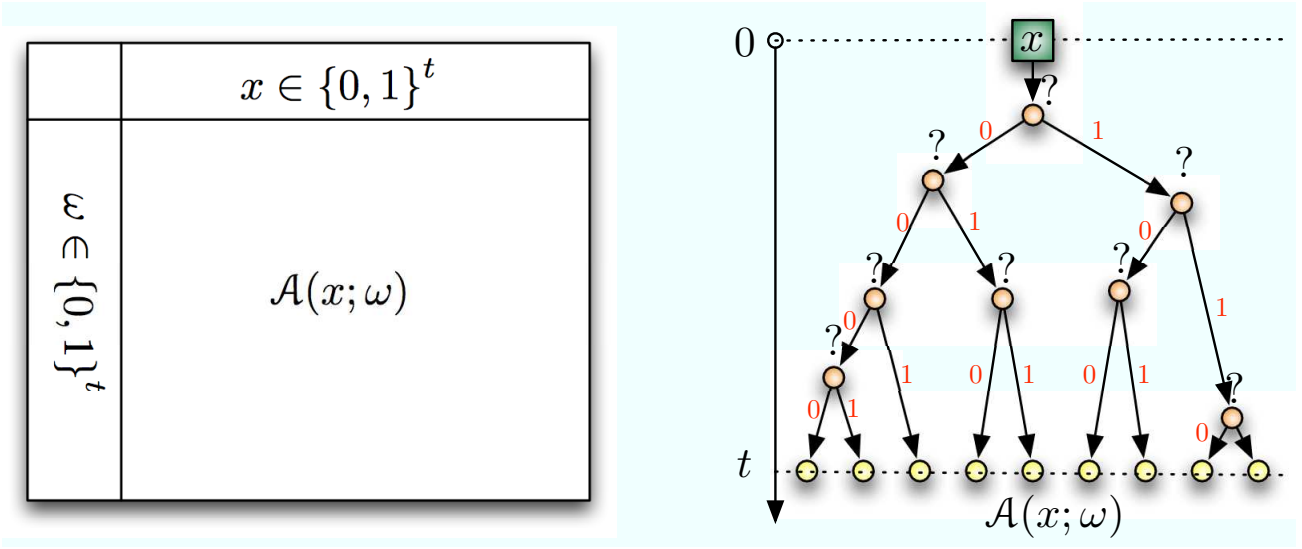
All events that are determined by condition $\{\omega \in \Omega : f(\omega) \in \mathcal{Y}\}$ for some set $\mathcal{Y} \subseteq \{0,1\}^*$ are *observable events* and their probability is defined

$$\Pr\left[\omega \in \Omega : f(\omega) \in \mathcal{Y}\right] \doteq \sum_{y \in \mathcal{Y}} \Pr\left[\omega \in \Omega_y\right] = \sum_{y \in \mathcal{Y}} p_y \ .$$

As a result, the probability measure is both additive and σ-additive as long as we consider mutually exclusive observable events.

Randomised algorithms and strategies

- A *randomised strategy* is a function of type $f : \{0,1\}^* \times \Omega \to \{0,1\}^*$ where the output $f(x) = f(x;\omega)$ depends on *randomness* ω.
- A *randomised algorithm* $\mathcal{A} : \{0,1\}^* \times \Omega \to \{0,1\}^*$ is a randomised strategy that has a finite, precise and complete description.
- A t-time randomised algorithm $\mathcal{A}$ can be represented as a table or a tree.



Analysis by Exhaustive Decomposition

Success of a compound adversary

Let $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3, \mathcal{A}_5$ be algorithms for finding discrete logarithm such that the success probability $\Pr\left[x \leftarrow \mathcal{A}_i(y) : y = g^x\right] \geq 7 \cdot \mathsf{Adv}^{\mathsf{dl}}_{\mathbb{G}}(\mathcal{A}_i)$ if $\pi_i(y) = 1$. Find the advantage $\mathsf{Adv}^{\mathsf{dl}}_{\mathbb{G}}(\mathcal{A})$ of the following adversary $\mathcal{B}$

$\mathcal{B}(y)$

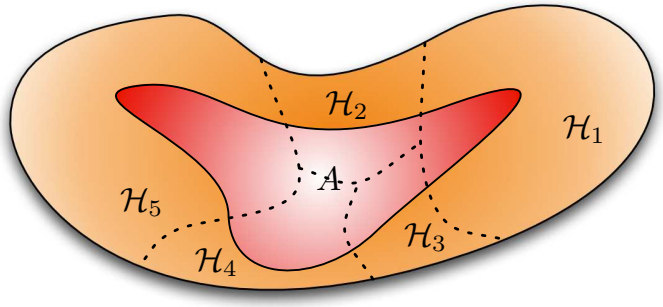
- $i \xleftarrow{u} \{1, 2, 3\}, x \leftarrow \mathcal{A}_i(y)$
- if $\pi_i(y) = 1$ then
  - if $g^x \neq y \wedge \pi_4(y) = 1$ then **return** $\mathcal{A}_4(y)$
  - else **return** x
- else if $\pi_5(y) = 1$ then **return** $\mathcal{A}_5(y)$
- else **return** $\mathcal{A}_1(y)$

provided that $\Pr\left[y \xleftarrow{u} \mathbb{G} : \pi_i(y) = 1\right] = \frac{1}{42+i}$ and $\mathsf{Adv}^{\mathsf{dl}}_{\mathbb{G}}(\mathcal{A}_i) = i^2 \cdot \varepsilon$.

Total probability formula

Let $\mathcal{H}_1, \ldots, \mathcal{H}_n$ be mutually exclusive events such that

$$\Pr[\mathcal{H}_i \wedge \mathcal{H}_j] = 0 \qquad \text{and} \qquad \Pr[\mathcal{H}_1 \vee \ldots \vee \mathcal{H}_n] = 1 \ .$$



Then for any any event A we can express

$$\Pr[A] = \sum_{i=1}^{n} \Pr[\mathcal{H}_i \wedge A] = \sum_{i=1}^{n} \Pr[\mathcal{H}_i] \cdot \Pr[A|\mathcal{H}_i] \ .$$

Conditional probability

Often, the presence of one event is correlated with some other events. The corresponding influence is formally quantified by *conditional probability*

$$\Pr\left[f(\omega) = y | g(\omega) = x\right] \doteq \frac{\Pr\left[f(\omega) = y \wedge g(\omega) = x\right]}{\Pr\left[g(\omega) = x\right]}$$

Consequently, for any two events A and B:

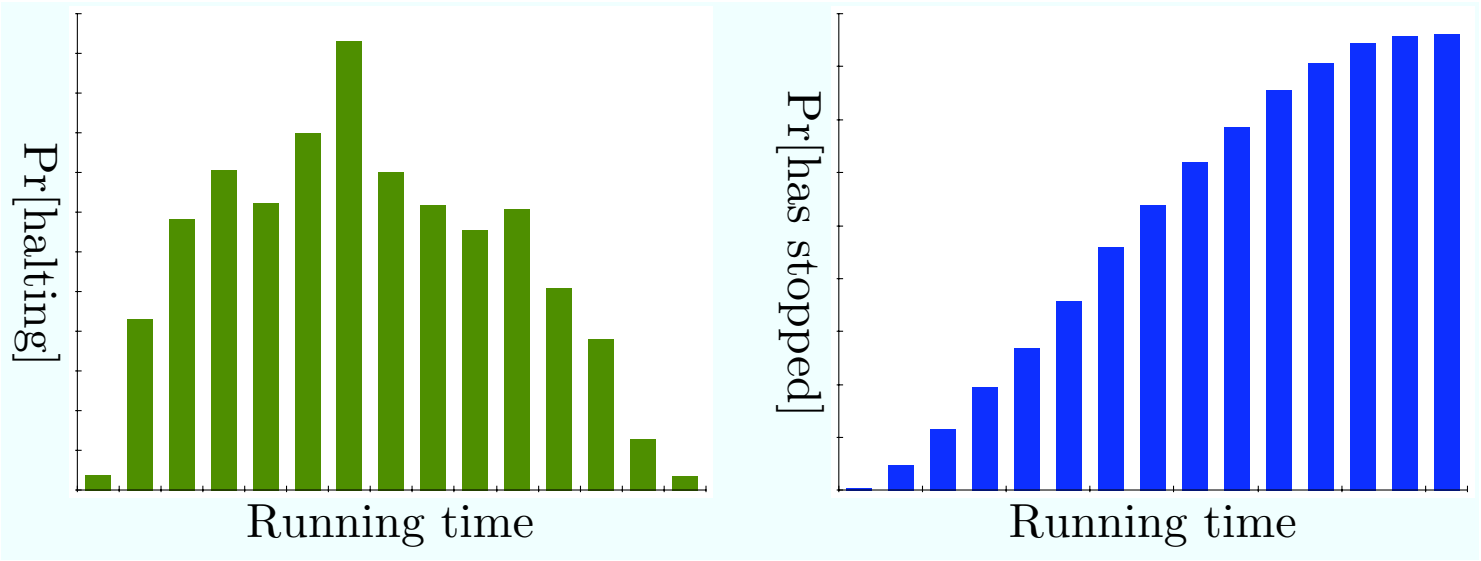
$$\Pr\left[A \wedge B\right] = \Pr\left[A\right] \cdot \Pr\left[B|A\right] = \Pr\left[B\right] \cdot \Pr\left[A|B\right] \quad .$$

Two *events are independent* if $\Pr\left[A \wedge B\right] = \Pr\left[A\right] \cdot \Pr\left[B\right]$.

Premature Halting Problem

The effect of premature halting

Let $\mathcal{A}$ be an algorithm that always succeeds but does not runs in constant time. What happens if we stop the after t time steps?



- ▷ The first graph corresponds to probability pseudo-density function.
- ▷ The second graph corresponds to cumulative distribution function.

Theory. PDF and CDF

Discrete random variables do not have a classical *probability density function*. Instead, we can consider probabilities of the smallest observable events $\Omega_\perp, \Omega_0, \Omega_1, \Omega_{00}, \ldots$. Consider the corresponding pseudo-density function

$$p_x \doteq \Pr\left[\omega \in \Omega : f(\omega) = x\right] \quad .$$

Then we can express a *cumulative distribution function*

$$F(y) = \Pr\left[\omega \in \Omega : f(\omega) \leq y\right]$$

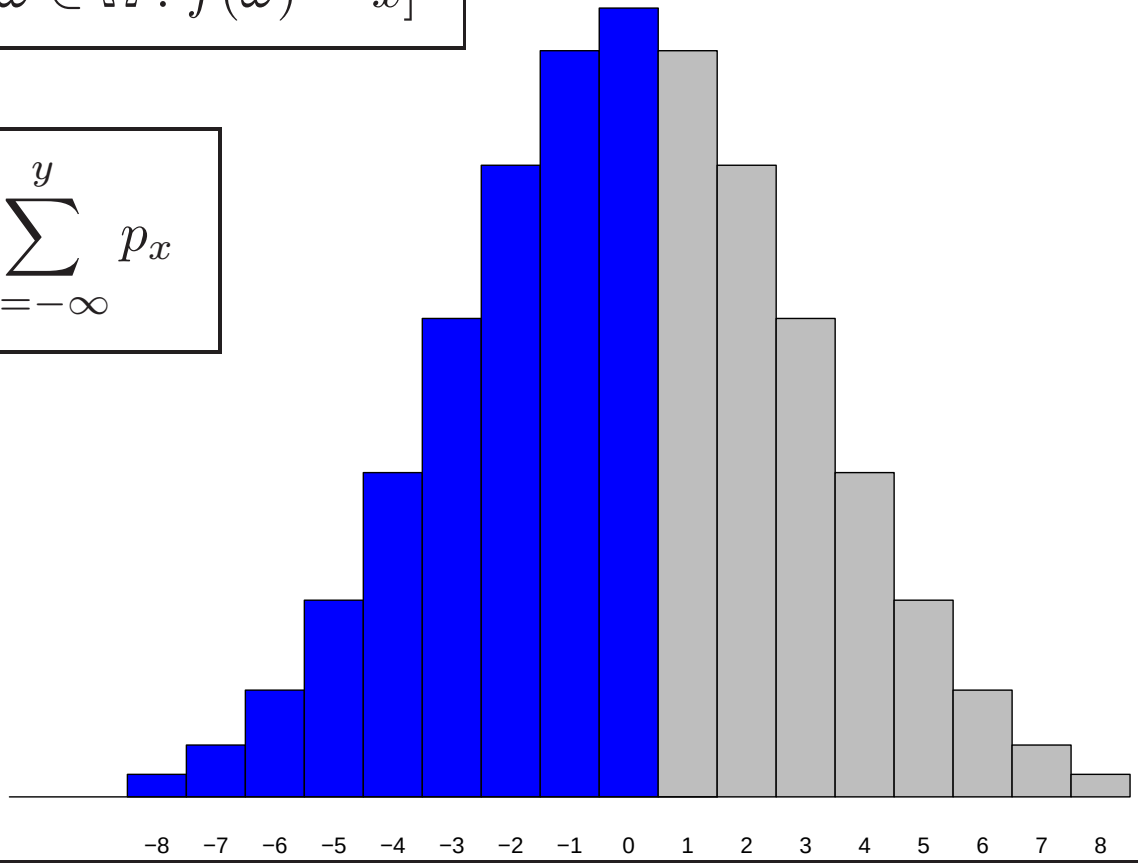
in terms of pseudo-density function

$$F(y) = \sum_{x=-\infty}^{y} \Pr\left[\omega \in \Omega : f(\omega) = x\right] = \sum_{x=-\infty}^{y} p_x \quad .$$

PDF and CDF. Illustration

$$p_x = \Pr\left[\omega \in \Omega : f(\omega) = x\right]$$

$$F(y) = \sum_{x=-\infty}^{y} p_x$$



Bounds on Expected Running-time

Amplification by repetition

Let $\mathcal{A}$ be a discrete logarithm finder with the advantage $\mathsf{Adv}^{\mathsf{dl}}_{\mathbb{G}}(\mathcal{A}) = \varepsilon$. What is the expected running-time of the following algorithm?

$\mathcal{B}^{\mathcal{A}}(y)$

while true **do**

$a \xleftarrow{u} \mathbb{Z}_q$

$x \leftarrow \mathcal{A}(yg^a) - a$

if $g^x = y$ then **return** x

- ▷ The program ends when $\mathcal{A}$ returns a correct answer
- ▷ All runs of $\mathcal{A}$ are independent and succeed with probability ε.

Expected value

The *expected value* of a random variable f is defined as

$$\mathbf{E}[f] = \sum_{x=-\infty}^{\infty} x \cdot \Pr[\omega \in \Omega : f(\omega) = x] = \sum_{x=-\infty}^{\infty} p_x \cdot x \ .$$

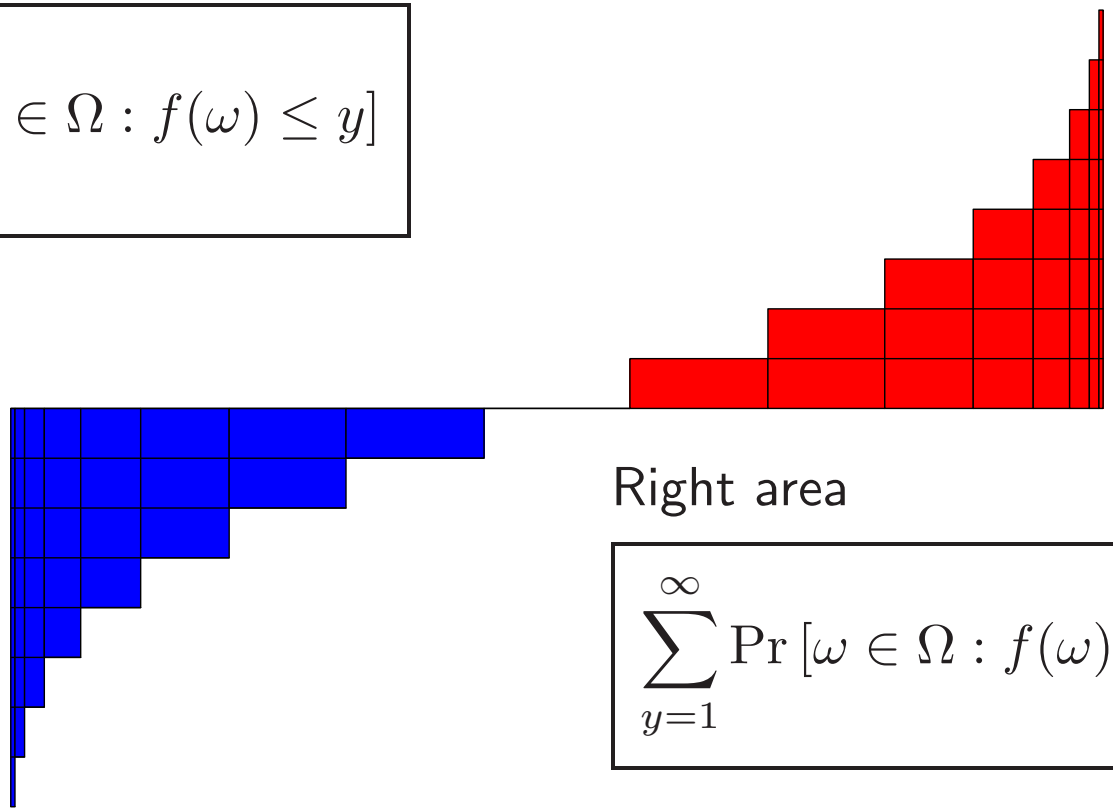
Alternatively, we can compute expected value as

$$\begin{aligned}\mathbf{E}[f] &= \sum_{y=1}^{\infty} \Pr[\omega \in \Omega : f(\omega) \geq y] - \sum_{y=-\infty}^{-1} \Pr[\omega \in \Omega : f(\omega) \leq y] \\ &= \sum_{y=0}^{\infty} (1 - F(y)) - \sum_{y=-\infty}^{-1} F(y) \ .\end{aligned}$$

Corresponding proof

Left area

$$\sum_{y=-\infty}^{-1} \Pr\left[\omega \in \Omega : f(\omega) \leq y\right]$$



Right area

$$\sum_{y=1}^{\infty} \Pr\left[\omega \in \Omega : f(\omega) \geq y\right]$$

Analysis of the discrete logarithm finder

Let ℓ be the number of iteration made by $\mathcal{B}^{\mathcal{A}}$ before stopping. Then

$$\Pr[\ell \geq y] = (1-\varepsilon)^{y-1}$$

and thus

$$\mathbf{E}[\ell] = \sum_{y=1}^{\infty}(1-\varepsilon)^{y-1} = \frac{1}{1-(1-\varepsilon)} = \frac{1}{\varepsilon} \ .$$

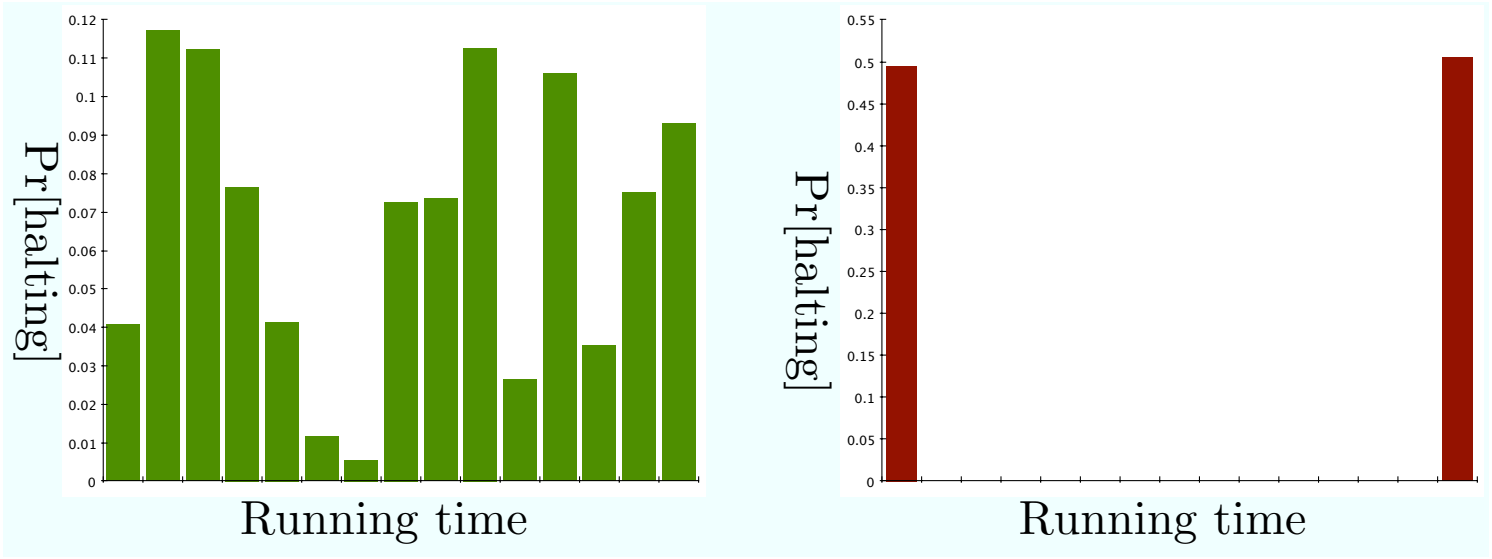
Hence, for a t-time adversary $\mathcal{A}$ the expected running-time of $\mathcal{B}^{\mathcal{A}}$ is

$$\frac{t}{\varepsilon} + \mathrm{O}\Big(\frac{1}{\varepsilon}\Big) \ .$$

Form Average Running-time to Success Probability

Premature halting problem revisited

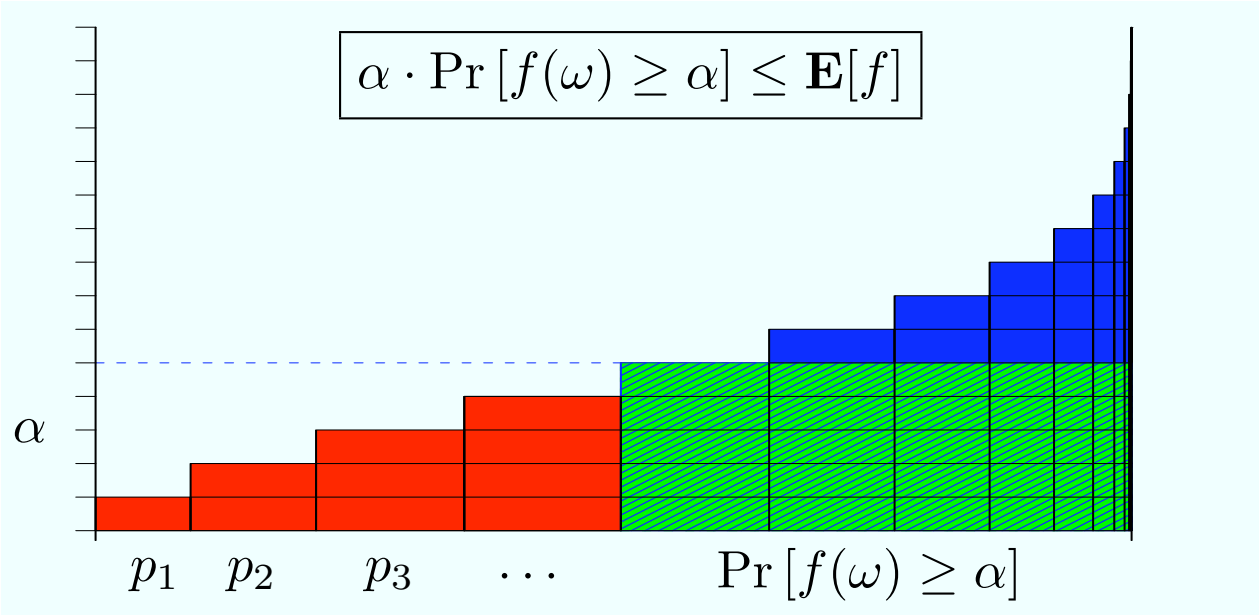
Let $\mathcal{A}$ be an algorithm that always succeeds but runs in expected time τ. What happens if we stop the algorithm after t time steps?



- ▷ Both distributions on the graph have the same expected value.
- ▷ Still, the expected value and variance limit potential distributions.

Markov's inequality

For every non-negative random variable $\Pr[f(\omega) \geq \alpha] \leq \frac{\mathbf{E}[f]}{\alpha}$.



Corollary. Any algorithm $\mathcal{A}$ is stops with probability at least $\frac{1}{2}$ after 2τ time steps where τ is the expected running time.

Success Amplification by Majority Voting

Amplification by majority voting

Let $\mathcal{A}$ be a CDH solver with the advantage $\mathsf{Adv}^{\mathsf{cdh}}_{\mathbb{G}}(\mathcal{A}) = \varepsilon > \frac{1}{2}$. Find a lower bound of the advantage of the following algorithm

$\mathcal{B}^{\mathcal{A}}(x, y)$

- For $i \in \{1, \ldots, n\}$ do
  - $a \xleftarrow{u} \mathbb{Z}_q, b \xleftarrow{u} \mathbb{Z}_q$
  - $z_i \leftarrow \mathcal{A}(xg^a, yg^b) \cdot x^{-b} y^{-a} g^{-ab}$
- Output the most frequent value among $z_1, \ldots z_n$.

- ▷ All runs of $\mathcal{A}$ are independent and succeed with probability ε.
- ▷ The program succeeds if more than half of the answers are correct.

Variance

Variance $\mathbf{D}[f]$ characterises how scattered are possible values $f(\omega)$:

$$\mathbf{D}[f] = \mathbf{E}\left[(f - \mathbf{E}[f])^2\right] = \mathbf{E}\left[f^2\right] - \mathbf{E}[f]^2 \quad .$$

Usually, one also needs standard deviation $\boldsymbol{\sigma}[f] = \sqrt{\mathbf{D}[f]}$.

Important properties

- ▷ If random variables $X_1, \ldots, X_n$ are pairwise independent then

$$\mathbf{D}[X_1 + \cdots + X_n] = \mathbf{D}[X_1] + \cdots + \mathbf{D}[X_n] \quad .$$

- ▷ For binary random variables $\mathbf{D}[X] = \Pr[X = 1] \Pr[X = 0]$.

Chebyshev's inequality

For any random variable $\Pr\left[|f(\omega) - \mathbf{E}\left[f\right]| \geq \alpha\right] \leq \frac{\mathbf{D}[f]}{\alpha^2}$.

Proof

- ▷ Let $g = (f - \mathbf{E}\left[f\right])^2$. Then by definition $\mathbf{D}\left[f\right] = \mathbf{E}\left[g\right]$.
- ▷ As g is non-negative, Markov's inequality assures that

$$\Pr\left[(f - \mathbf{E}\left[f\right])^2 > \alpha^2\right] \leq \frac{\mathbf{E}\left[g\right]}{\alpha^2}$$

$$\Updownarrow$$

$$\Pr\left[|f - \mathbf{E}\left[f\right]| > \alpha\right] \leq \frac{\mathbf{D}\left[f\right]}{\alpha^2}$$

Analysis of the CDH solver

Let X_i denote whether $\mathcal{A}$ succeeded in computing z_i and let X be the number of correct answers. Then the following claims hold:

- ▷ The advantage can be expressed as $\mathsf{Adv}^{\mathsf{cdh}}_{\mathbb{G}}(\mathcal{B}) = \Pr\left[X > \frac{n}{2}\right]$.
- ▷ The variance can be computed as $\mathbf{D}\left[X\right] = n\varepsilon(1-\varepsilon)$.
- ▷ Chebyshev's inequality gives

$$\begin{aligned}\Pr\left[X \leq \tfrac{n}{2}\right] &= \Pr\left[|X - \varepsilon n| \geq n\left(\varepsilon - \tfrac{1}{2}\right)\right] \\ &\leq \frac{4n\varepsilon(1-\varepsilon)}{n^2(2\varepsilon-1)^2} = \frac{4\varepsilon(1-\varepsilon)}{n(2\varepsilon-1)^2}\end{aligned}$$

- ▷ The upper bound on the failure probability is inversely proportional to n.

Remark. Hoeffding and Chernoff bounds provide sharper estimates.

Basic Properties of Entropy

Jensen's inequality

Let x be a random variable. Then for every convex-cup function f

$$\mathbf{E}\left[f(x)\right] \leq f(\mathbf{E}\left[x\right])$$

and for every convex-cap function g

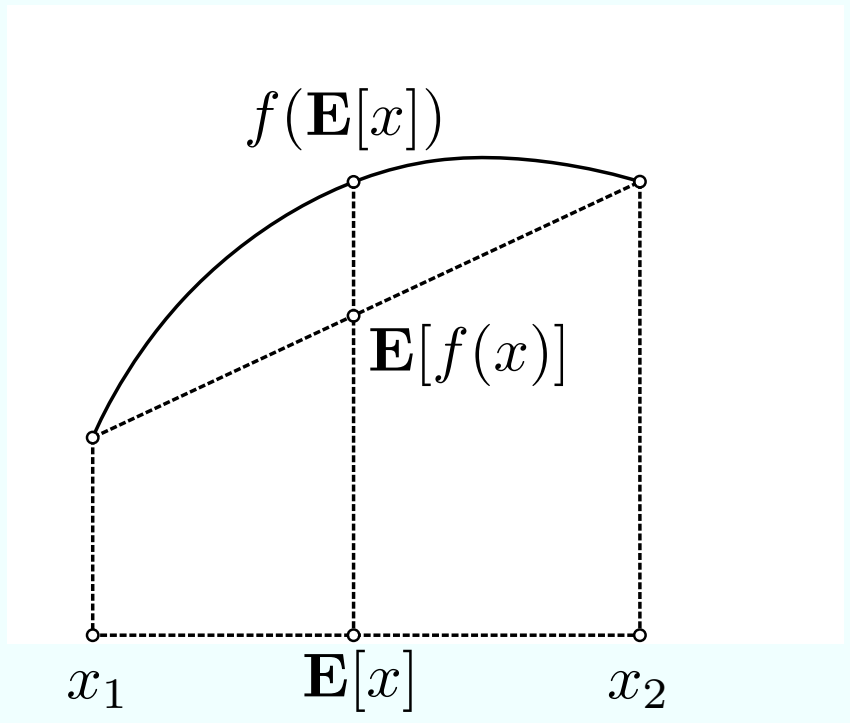
$$\mathbf{E}\left[g(x)\right] \geq g(\mathbf{E}\left[x\right]) \ .$$

These inequalities are often used to get lower and upper bounds:

- ▷ for success probabilities,
- ▷ for complex expressions involving probabilities.

Corresponding proof

Note that it is sufficient to give a proof for sums with two terms.



For any weight p_1 and p_2, the expected values align as shown in the figure.

Shannon entropy

Entropy is another measure of uncertainty for random variables. Intuitively, it captures the minimal amount of bits that are needed on average to describe a value of a random variable X.

Shannon entropy is defined as follows

$$H(X) = -\sum_{x \in \{0,1\}^*} p_x \cdot \log_2 p_x = \mathbf{E}\left[\log_2 \frac{1}{\Pr[X]}\right]$$

Jensen's inequality assures that

$$0 \leq H(X) \leq \log_2 |\mathrm{supp}(X)|$$

where the *support* of X is defined as $\mathrm{supp}(X) = \left\{x \in \{0,1\}^* : p_x > 0\right\}$.

Conditional of entropy

Conditional entropy is defined as follows

$$H(Y|X) = -\mathbf{E}_{X,Y}\left[\log_2 \Pr\left[Y|X\right]\right]$$

Now observe that

$$\begin{aligned} H(X,Y) &= -\mathbf{E}_{X,Y}\left[\log_2 \Pr\left[X \wedge Y\right]\right] \\ &= -\mathbf{E}_{X,Y}\left[\log_2 \Pr\left[X\right] + \log_2 \Pr\left[Y|X\right]\right] \\ &= -\mathbf{E}_{X}\left[\log_2 \Pr\left[X\right]\right] - \mathbf{E}_{X,Y}\left[\log_2 \Pr\left[Y|X\right]\right] \\ &= H(X) + H(Y|X) \ . \end{aligned}$$

Mutual information

Recall that entropy characterises the average length of minimal description. Now if we consider two random variables. Then we can describe them jointly or separately. *Mutual information* captures the corresponding gain

$$I(Y : X) = H(X) + H(Y) - H(X, Y)$$

Evidently, mutual information between independent variables is zero:

$$I(Y : X) = H(X) + H(Y) - H(X) - \underbrace{H(Y|X)}_{H(Y)} = 0 \ .$$

Similarly, if X and Y coincide then

$$I(Y : X) = H(X) + H(Y) - H(X) - \underbrace{H(Y|X)}_{0} = H(X) \ .$$

Min-entropy. Rényi entropy

Shannon entropy is not always descriptive enough for measuring uncertainty. For example, consider security of passwords.

- ▷ Obviously, we can just try the most probable password. The corresponding uncertainty measure is known as *min-entropy*

$$H_\infty(X) = -\log_2 \max_{x \in \{0,1\}^*} \Pr[X = x]$$

- ▷ Often, we do not want that two persons have coinciding passwords. The corresponding uncertainty measure is known as *Rényi entropy*

$$H_2(X) = -\log_2 \Pr[x_1 \leftarrow X, x_2 \leftarrow X : x_1 = x_2]$$

where x_1 and x_2 are independent draws from the distribution X.