Rare Event Simulation for a Static Distribution

F. Cérou[1] P. Del Moral[2] T. Furon[3] A. Guyader[4]

[1]INRIA Rennes

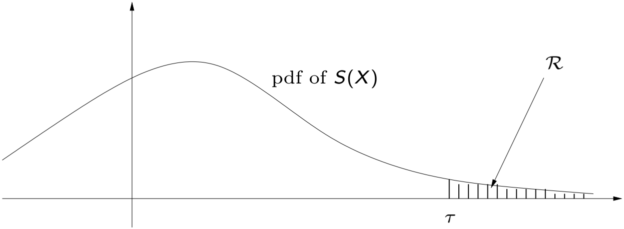
[2]INRIA Bordeaux et Institut de Mathématiques de Bordeaux

[3]Thomson et INRIA Rennes

[4]IRMAR et INRIA Rennes

Groupe de Travail “Rare Events”
CMAP, Ecole Polytechnique, 30 mars 2009

The Model



- Let $X \in E$ be a random vector and $S : E \to \mathbb{R}$ a score.
- **Goal**: Estimate $\alpha = \mathbb{P}(S(X) > \tau) < 10^{-6}$.
- **Framework**: we can only simulate $X \sim \mu$ and compute S at each point, but any analytical study is excluded.

$\Rightarrow$ **Monte-Carlo methods**.

Naive Monte-Carlo

Recall: the aim is to estimate

$$\alpha = \mathbb{P}(\mathcal{R}) = \mathbb{P}(S(X) > \tau).$$

- Simulate $\xi_1, \ldots, \xi_N \sim \mu$ and denote

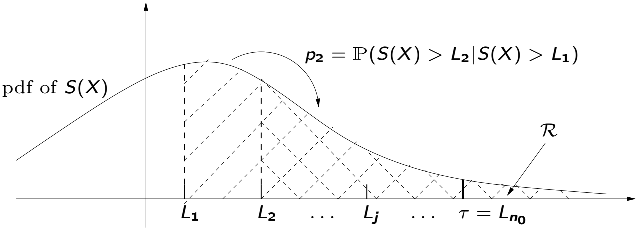
$$N_{\mathcal{R}} = \#\{i \in \{1, \ldots, N\},\ S(\xi_i) > \tau\}$$

- Monte-Carlo Estimate: $\hat{\alpha}_N = N_{\mathcal{R}}/N$, but...
    - About α^{-1} simulations are necessary to make $\mathcal{R}$ occur.
    - The relative standard deviation is a disaster:

$$\frac{\sigma(\hat{\alpha}_N)}{\alpha} = \frac{\sqrt{1-\alpha}}{\sqrt{N\alpha}} \approx \frac{1}{\sqrt{N\alpha}}.$$

$\Rightarrow$ **Idea**: Multilevel Monte-Carlo Method.

Main Idea



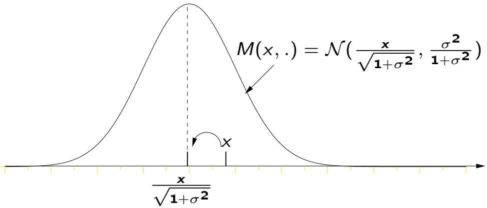
- **Ingredients**: fix n_0 and $L_1 < \cdots < L_{n_0} = \tau$ so that each $p_j = \mathbb{P}(S(X) > L_j | S(X) > L_{j-1})$ is not too small.
- **Bayes decomposition**: $\alpha = p_1 p_2 \dots p_{n_0}$.
- **Unreasonable assumption**: suppose we can estimate each p_j independently with usual Monte-Carlo: $p_j \approx \hat{p}_j = N_j/N$.
- **Multilevel Estimator**: $\hat{\alpha}_N = \hat{p}_1 \hat{p}_2 \dots \hat{p}_{n_0}$.

The Shaker

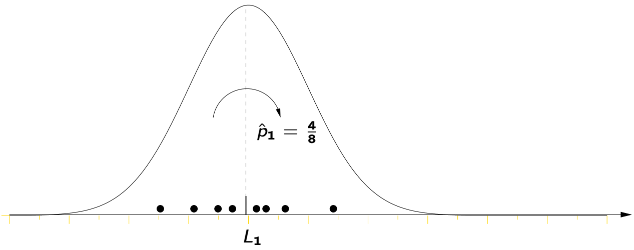
- **Recall**: $X \sim \mu$ on E.
- **Ingredient**: a $\mu-$reversible transition kernel $M(x, dx')$ on E :

$$\forall (x, x') \in E^2 \qquad \mu(dx)M(x, dx') = \mu(dx')M(x', dx).$$

- **Consequence** : $\mu M = \mu$.
- **Example**: if $X \sim \mathcal{N}(0,1)$ then $X' = \frac{X+\sigma W}{\sqrt{1+\sigma^2}} \sim \mathcal{N}(0,1)$, i.e. $M(x, dx') \sim \mathcal{N}(\frac{x}{\sqrt{1+\sigma^2}}, \frac{\sigma^2}{1+\sigma^2})(dx')$ is a "good shaker".


$M(x,.) = \mathcal{N}(\frac{x}{\sqrt{1+\sigma^2}}, \frac{\sigma^2}{1+\sigma^2})$
x
$\frac{x}{\sqrt{1+\sigma^2}}$

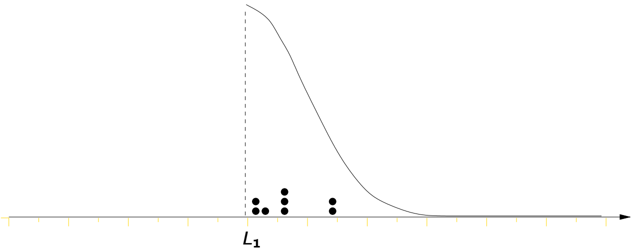
A Selection/Mutation Algorithm



- **Initialization**: Simulate an i.i.d. sample $\xi_0^1, \ldots, \xi_0^N \sim \mu$.
- **Selection**: $\hat{\xi}_0^i = \xi_0^i$ if $S(\xi_0^i) > L_1$, else pick at random among the N_1 selected particles.
- **Mutation**: $\tilde{\xi}_0^i \sim M(\hat{\xi}_0^i, dx')$ and

$$\forall i \in \{1, \ldots, N\} \qquad \xi_1^i = \begin{cases} \tilde{\xi}_1^i & \text{if } S(\tilde{\xi}_1^i) > L_1 \\ \hat{\xi}_1^i & \text{if } S(\tilde{\xi}_1^i) \leq L_1 \end{cases}$$

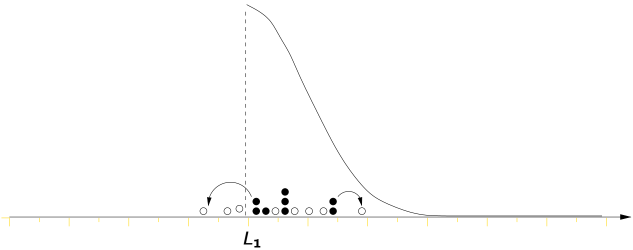
A Selection/Mutation Algorithm



- **Initialization**: Simulate an i.i.d. sample $\xi_0^1, \ldots, \xi_0^N \sim \mu$.
- **Selection**: $\hat{\xi}_0^i = \xi_0^i$ if $S(\xi_0^i) > L_1$, else pick at random among the N_1 selected particles.
- **Mutation**: $\tilde{\xi}_0^i \sim M(\hat{\xi}_0^i, dx')$ and

$$\forall i \in \{1, \ldots, N\} \qquad \xi_1^i = \begin{cases} \tilde{\xi}_1^i & \text{if } S(\tilde{\xi}_1^i) > L_1 \\ \hat{\xi}_1^i & \text{if } S(\tilde{\xi}_1^i) \leq L_1 \end{cases}$$

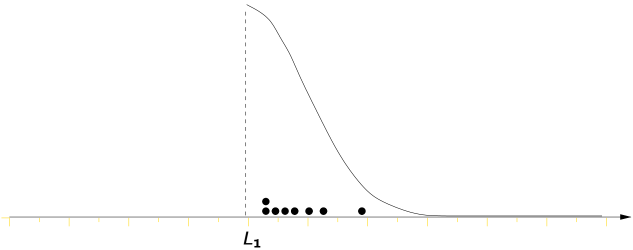
A Selection/Mutation Algorithm



- **Initialization**: Simulate an i.i.d. sample $\xi_0^1, \dots, \xi_0^N \sim \mu$.
- **Selection**: $\hat{\xi}_0^i = \xi_0^i$ if $S(\xi_0^i) > L_1$, else pick at random among the N_1 selected particles.
- **Mutation**: $\tilde{\xi}_0^i \sim M(\hat{\xi}_0^i, dx')$ and

$$\forall i \in \{1, \dots, N\} \qquad \xi_1^i = \begin{cases} \tilde{\xi}_1^i & \text{if } S(\tilde{\xi}_1^i) > L_1 \\ \hat{\xi}_1^i & \text{if } S(\tilde{\xi}_1^i) \leq L_1 \end{cases}$$

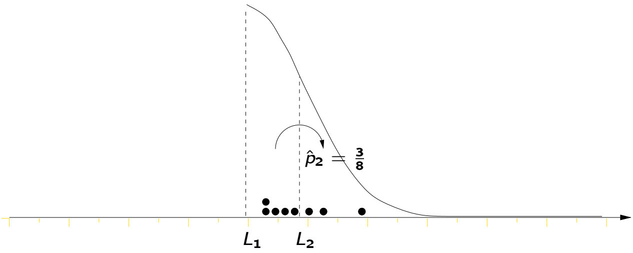
A Selection/Mutation Algorithm



- **Initialization**: Simulate an i.i.d. sample $\xi_0^1, \ldots, \xi_0^N \sim \mu$.
- **Selection**: $\hat{\xi}_0^i = \xi_0^i$ if $S(\xi_0^i) > L_1$, else pick at random among the N_1 selected particles.
- **Mutation**: $\tilde{\xi}_0^i \sim M(\hat{\xi}_0^i, dx')$ and

$$\forall i \in \{1, \ldots, N\} \qquad \xi_1^i = \begin{cases} \tilde{\xi}_1^i & \text{if } S(\tilde{\xi}_1^i) > L_1 \\ \hat{\xi}_1^i & \text{if } S(\tilde{\xi}_1^i) \leq L_1 \end{cases}$$

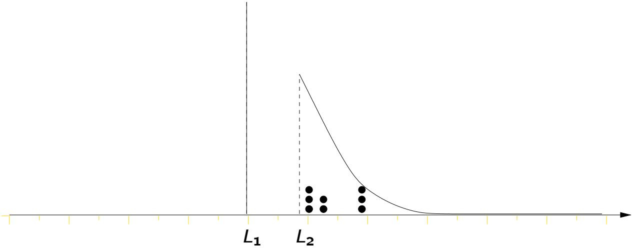
A Selection/Mutation Algorithm



- **Initialization**: Simulate an i.i.d. sample $\xi_0^1, \ldots, \xi_0^N \sim \mu$.
- **Selection**: $\hat{\xi}_0^i = \xi_0^i$ if $S(\xi_0^i) > L_1$, else pick at random among the N_1 selected particles.
- **Mutation**: $\tilde{\xi}_0^i \sim M(\hat{\xi}_0^i, dx')$ and

$$\forall i \in \{1, \ldots, N\} \qquad \xi_1^i = \begin{cases} \tilde{\xi}_1^i & \text{if } S(\tilde{\xi}_1^i) > L_1 \\ \hat{\xi}_1^i & \text{if } S(\tilde{\xi}_1^i) \leq L_1 \end{cases}$$

A Selection/Mutation Algorithm



- **Initialization**: Simulate an i.i.d. sample $\xi_0^1, \dots, \xi_0^N \sim \mu$.
- **Selection**: $\hat{\xi}_0^i = \xi_0^i$ if $S(\xi_0^i) > L_1$, else pick at random among the N_1 selected particles.
- **Mutation**: $\tilde{\xi}_0^i \sim M(\hat{\xi}_0^i, dx')$ and

$$\forall i \in \{1, \dots, N\} \qquad \xi_1^i = \begin{cases} \tilde{\xi}_1^i & \text{if } S(\tilde{\xi}_1^i) > L_1 \\ \hat{\xi}_1^i & \text{if } S(\tilde{\xi}_1^i) \leq L_1 \end{cases}$$

Convergence of the Algorithm

- $A_n = \{x \in E : \; S(x) > L_n\}$ and $\mu_n = \mathcal{L}(X|S(X) > L_n)$.
- Non-homogeneous transition kernel:

$$M_n(x, dx') = M(x, dx')\mathbb{1}_{A_n}(x') + M(x, A_n^c)\delta_x(dx').$$

It is easy to check that μ_n is invariant by M_n.

Theorem (Feynman-Kac Formula)

Define a Markov chain (X_n) having the transition kernels (M_n) and initial law μ, then for any test function φ and any n:

$$\mu_n(\varphi) = \frac{\mathbb{E}[\varphi(X_n)\prod_{j=1}^{n}\mathbb{1}_{A_j}(X_{j-1})]}{\mathbb{E}[\prod_{j=1}^{n}\mathbb{1}_{A_j}(X_{j-1})]}.$$

Remark: thus, after n_0 steps, $\mu_{n_0} = \mathcal{L}(X|S(X) > \tau)$.

Variance of the estimator

Theorem (Cérou *et al.*, ALEA (2006))

$$\sqrt{N}\cdot\frac{\hat{\alpha}_N-\alpha}{\alpha}\xrightarrow[N\to\infty]{\mathcal{L}}\mathcal{N}(0,\sigma^2),$$

with

$$\sigma^2=\sum_{j=1}^{n_0}\frac{1-p_j}{p_j}+\sum_{j=1}^{n_0}\frac{\mathbb{V}(\mathbb{P}(S(X_{n_0-1})>L_{n_0}|X_j,S(X_{j-1})>L_j))}{\mathbb{P}^2(S(X_{n_0-1})>L_{n_0}|S(X_{j-1})>L_j)}\frac{1-p_j^2}{p_j}$$

Remark: $\sigma^2\geq\sum_{j=1}^{n_0}\frac{1-p_j}{p_j}$, with equality iff

$$\mathbb{P}(S(X_{n_0-1})>L_{n_0}|X_j,S(X_{j-1})>L_j)\perp X_j.$$

⇒ **Solution**: at each step, iterate the transition kernel.

Iterations of the Kernel

- **Problem**: the choice of M depends on the application, but if μ is a Gibbs measure given by a bounded potential, then...
- **Metropolis Method** $\Rightarrow$ M usually aperiodic and irreducible.
- **Tierney (Annals of Stat, 1994)**: for any initial law λ

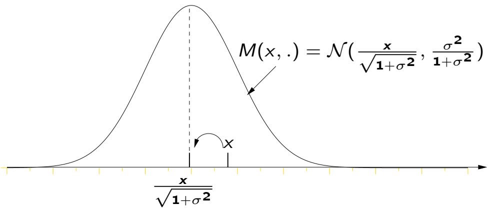
$$\left\| \int \lambda(dx) M_n^m(x,.) - \mu_n \right\|_{tv} \xrightarrow[m\to\infty]{} 0.$$

- **Corollary**: for any cloud of particles $\Xi = (\xi_1, \ldots, \xi_N)$ and any test function ϕ

$$\left| \int \delta_{\Xi}((M_n^{\otimes N})^m)(\phi) - \mu_n^{\otimes N}(\phi) \right| \xrightarrow[m\to\infty]{} 0.$$

- **Rule of thumb**: at each step, iterate the kernel until 90% of the particles have actually moved.

The Impact of the Kernel


$M(x,.) = \mathcal{N}\left(\frac{x}{\sqrt{1+\sigma^2}}, \frac{\sigma^2}{1+\sigma^2}\right)$
x
$\frac{x}{\sqrt{1+\sigma^2}}$

- **The model**: $X' = \frac{X+\sigma W}{\sqrt{1+\sigma^2}} \sim \mathcal{N}(0,1)$.
- **Expected square distance**: $\mathbb{E}[(X'-X)^2] = 2\left(1 - \frac{1}{\sqrt{1+\sigma^2}}\right)$.

Trade-off between two drawbacks:

- σ too large: most proposed mutations are refused.
- σ too small: particles almost don't move.

Constrained Optimization

- **Multilevel Estimator**: $\hat{\alpha}_N = \hat{p}_1 \hat{p}_2 \dots \hat{p}_{n_0}$.
- **Fluctuations**: If the $\hat{p}_i$'s are independent, then

$$\sqrt{N} \cdot \frac{\hat{\alpha}_N - \alpha}{\alpha} \xrightarrow[N \to \infty]{\mathcal{L}} \mathcal{N}\left(0, \sum_{j=1}^{n_0} \frac{1 - p_j}{p_j}\right).$$

- **Constrained Minimization**:

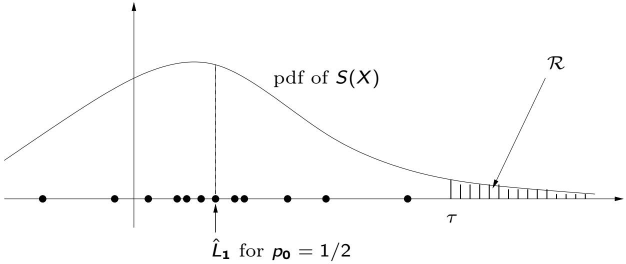
$$\arg \min_{p_1, \dots, p_{n_0}} \sum_{j=1}^{n_0} \frac{1 - p_j}{p_j} \qquad \text{s.t.} \qquad \prod_{j=1}^{n_0} p_j = \alpha.$$

- **Optimum**: $p_1 = \dots = p_{n_0} = \alpha^{1/n_0}$.

⇒ **Solution**: Adaptive levels.

Adaptive Levels

Parameter: fix a proportion p_0 of surviving particles from one step to another rather than n_0 and the levels $L_1, \dots, L_{n_0}$.



$\Rightarrow$ **Adaptive multilevel estimator**:

$$\alpha = r \times p_0^{n_0} \approx \hat{\alpha}_N = \hat{r} \times p_0^{\hat{n}_0},$$

with $n_0 = \left\lfloor \frac{\log \mathbb{P}(S(X)>\tau)}{\log p_0} \right\rfloor$ and $p_0 < r \leq 1$.

Empirical Quantiles

- $\hat{L}_1 \approx L_1$ with $\mathbb{P}(S(X) > L_1) = p_0$.
- Iterate the kernel M an "infinite" number of times, then the particles $\xi_1^1, \dots, \xi_1^N$ are i.i.d. with distribution

$$\mathcal{L}(X|S(X) > \hat{L}_1) \approx \mathcal{L}(X|S(X) > L_1).$$

- $\hat{L}_2 \approx L_2$ with $\mathbb{P}(S(X) > L_2|S(X) > L_1) = p_0$.
- etc.

$\Rightarrow$ if $F(t) \triangleq \mathbb{P}(S(X) \leq t)$, then the $L_j's$ are such that

$$\forall j \geq 0 \qquad \frac{1 - F(L_{j+1})}{1 - F(L_j)} = p_0.$$

Consistency

Theorem (Cérou and Guyader, SAA (2007))

Suppose that F is continuous, then

$$\hat{\alpha}_N \xrightarrow[N\to\infty]{a.s.} \alpha.$$

Sketch of the proof:

- Iterations of $M_j \Rightarrow$ knowing $\hat{L}_j$, the $(\xi_j^i)_{1\leq i\leq N}$ are i.i.d. with distribution $\mathcal{L}(X|S(X) > \hat{L}_j)$.
- $\mathbf{F}(q, q') \triangleq \mathbb{P}(S(X) \leq L' \mid S(X) > L) = \frac{F(L')-F(L)}{1-F(L)}$.
- Convergence of the quantiles : $\forall j$, $\mathbf{F}(\hat{L}_j, \hat{L}_{j+1}) \xrightarrow[N\to\infty]{a.s.} 1 - p_0$.
- Induction on j.

Variance of the Estimator

Theorem (Cérou and Guyader, SAA (2007))

Suppose that F is continuous, then

$$\sqrt{N}\,\frac{\hat{\alpha}_N - \alpha}{\alpha} \xrightarrow[N\to\infty]{\mathcal{D}} \mathcal{N}(0, \sigma^2),$$

with

$$\sigma^2 = n_0 \frac{1 - p_0}{p_0} + \frac{1 - r}{r}.$$

Remark: For fixed levels, we can also obtain non asymptotic variance results and deduce the logarithmic efficiency of the estimate (Cérou, Del Moral and Guyader (2009)).

Proof of the Variance

- $\forall j \geq 0$, we have

$$\mathbb{E}[\varphi(\mathbf{F}(\hat{L}_j, \hat{L}_{j+1}))|\hat{L}_j] = \mathbb{E}[\varphi(U_{(N-\lfloor p_0 N \rfloor)})].$$

- Triangular array of uniform variables:

$$\sqrt{N}(U_{(N-\lfloor p_0 N \rfloor)} - (1-p_0)) \xrightarrow[N\to\infty]{\mathcal{D}} \mathcal{N}(0, p_0(1-p_0)).$$

- Induction on

$$\sqrt{N}\left(\prod_{j=1}^{n}[1-\mathbf{F}(\hat{L}_j, \hat{L}_{j+1})] - {p_0}^n\right).$$

Bias of the Estimator

Theorem (Cérou, Del Moral, Furon and Guyader (2009))

Suppose that F is continuous, then

$$N\,\frac{\mathbb{E}[\hat{\alpha}_N]-\alpha}{\alpha}\xrightarrow[N\to\infty]{} b=n_0\frac{1-p_0}{p_0}.$$

Remarks:

- The bias is of order $1/N$ and is thus negligible compared to the standard deviation.
- The biais is non negative, leading to a slightly overvalued estimate, which is a nice property in concrete situations.

Proof of the Bias

- Suppose $\hat{n}_0 = n_0$, then:

$$\frac{\mathbb{E}[\hat{\alpha}_N] - \alpha}{\alpha} = \frac{\mathbb{E}[\hat{r}] - r}{r} = \mathbb{E}\left[\frac{W_N}{a - W_N}\right],$$

with $a = 1 - F(L_{n_0}) = p_0^{n_0}$ and

$$W_N = F(\hat{L}_{n_0}) - F(L_{n_0}) \xrightarrow[N \to \infty]{a.s.} 0.$$

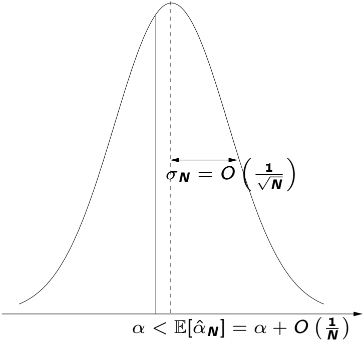
- Make an asymptotic expansion near 0

$$\frac{\mathbb{E}[\hat{\alpha}_N] - \alpha}{\alpha} = \frac{\mathbb{E}[W_N]}{a} + \frac{\mathbb{E}[W_N^2]}{a^2} + \frac{1}{a^2} o(\mathbb{E}[W_N^2]).$$

- Finally, remark that $\mathbb{E}[W_N] = 0$ and

$$\frac{\mathbb{E}[W_N^2]}{a^2} = \frac{n_0}{N} \cdot \frac{1 - p_0}{p_0} + o\left(\frac{1}{N}\right).$$

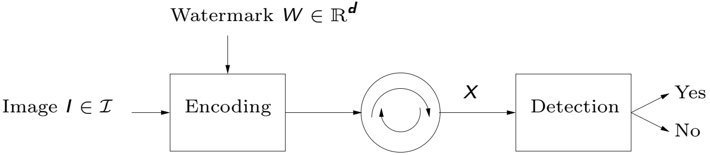
Asymptotic Expansion


$\sigma_N = O\left(\frac{1}{\sqrt{N}}\right)$
$\alpha < \mathbb{E}[\hat{\alpha}_N] = \alpha + O\left(\frac{1}{N}\right)$

Summary: Putting all things together, we have obtained

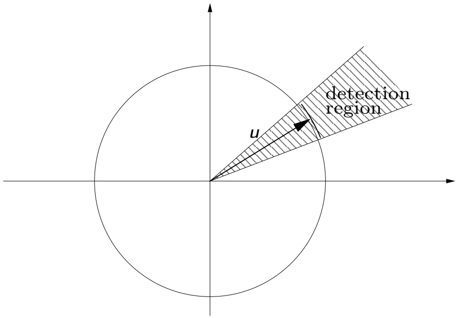
$$\hat{\alpha}_N = \alpha\left(1 + \frac{\sigma}{\sqrt{N}}Y + \frac{b}{N} + o_{\mathbb{P}}\left(\frac{1}{N}\right)\right).$$

Zero-Bit Watermarking



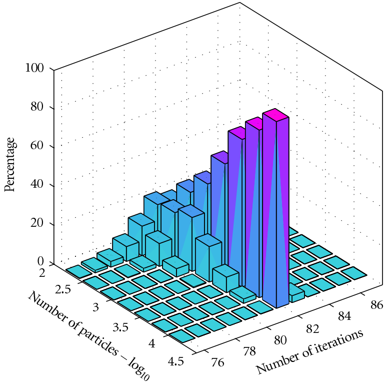
- **Principle**: The watermark must be both invisible and robust.
- **False Detection**: An unwatermarked content detected as watermarked.
- **Constraint**: Copy Protection Working Group $\Rightarrow P_{fd} < 10^{-5}$.

Zero-Bit Watermarking



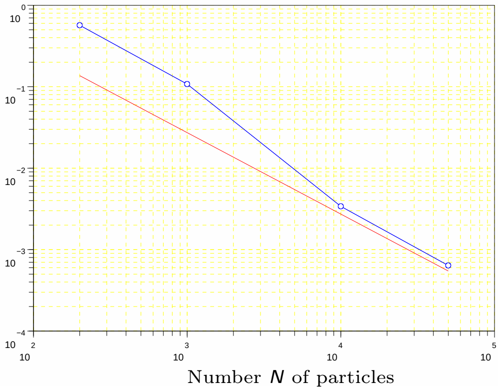
- $u \in \mathbb{R}^d$ is a fixed and normalized secret vector.
- A content X is deemed watermarked if $S(X) = \frac{\langle X, u \rangle}{\|X\|} > \tau$.
- **Usual assumption**: An unwatermarked content X has a radially symmetric pdf.
- **False detection**: $P_{fd} = \mathbb{P}(S(X) > \tau | X \text{ unwatermarked})$.

Number of Iterations



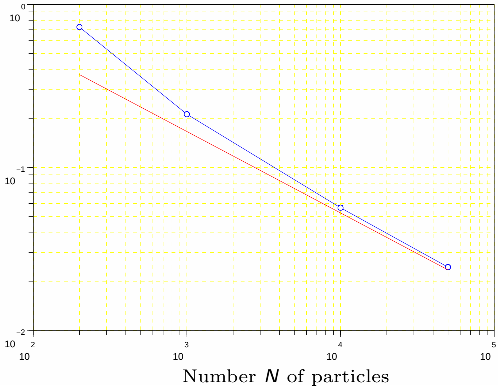
- **The model**: $X \sim \mathcal{N}(0, I_{20})$.
- **Rare event**: $\alpha = \mathbb{P}\left(\frac{\langle X,u\rangle}{\|X\|} > 0.95\right)$.
- **Numerical computation**: $\alpha = 4.704 \cdot 10^{-11}$.
- **Parameter**: $p_0 = 3/4 \rightsquigarrow \alpha = r \times p_0^{n_0} = 0.83 \times (3/4)^{82}$.

Bias



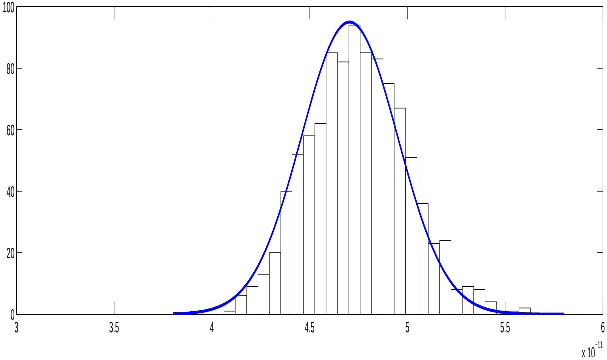
$$\frac{\mathbb{E}[\hat{\alpha}_N] - \alpha}{\alpha} \approx \frac{b}{N} = \frac{1}{N} \cdot n_0 \frac{1 - p_0}{p_0}.$$

Standard Deviation



$$\hat{\sigma}_N \approx \frac{\sigma}{\sqrt{N}} = \frac{1}{\sqrt{N}} \cdot \sqrt{n_0 \cdot \frac{1-p_0}{p_0} + \frac{1-r}{r}}.$$

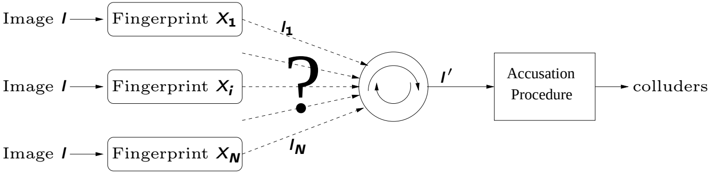
Histogram



Asymptotic expansion

$$\hat{\alpha}_N = \alpha \left(1 + \frac{\sigma}{\sqrt{N}}\, \mathcal{N}(0,1) + \frac{b}{N} + \dots \right)$$

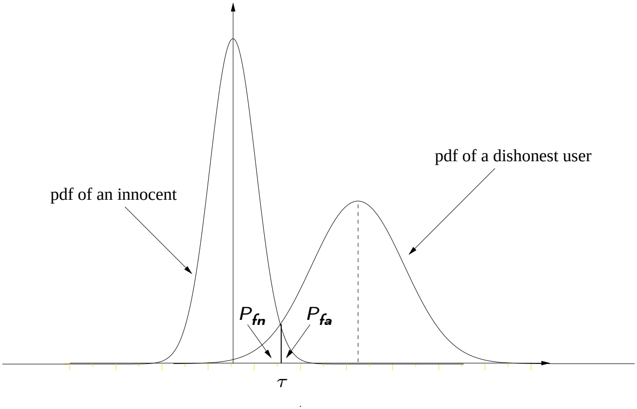
Fingerprinting



- **Principle**: $X_i \in \{0,1\}^m$ is hidden in the copy of each user.
- **Benefit**: Find a dishonest user via his fingerprint.
- **Question**: What if several dishonest users collude ?
- **False Detections**: Accusing an innocent (false alarm) or accusing none of the colluders (false negative).

$\Rightarrow$ **Answer**: Tardos probabilistic codes.

Probabilistic Fingerprinting



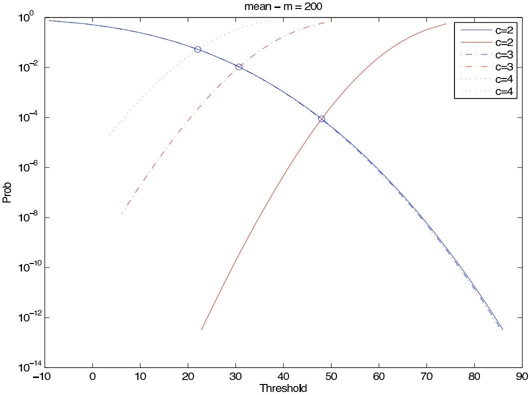
- **Fingerprint**: $X = [X_1, \dots, X_m]$, $X_\ell \sim \mathcal{B}(p_\ell)$ and $p_\ell \sim f(p)$.
- **Pirated Copy**: $y = [y_1, \dots, y_m] \in \{0,1\}^m$.
- **Accusation procedure**: $S(X) = \sum_{\ell=1}^{m} y_\ell g_\ell(X_\ell) \gtrless \tau$.

Estimation of P_{fa}

- **Parameters**: Fix m, N, r, c, p_0 and the threshold τ.
- **Colluders**: c fingerprints $\rightsquigarrow y = [y_1, \ldots, y_m]$.
- **Initialization**: N fingerprints $\xi_1, \ldots, \xi_N$.
- **Scores**: $\forall i$, compute $S(\xi_i) = \sum_{\ell=1}^{m} y_\ell g_\ell(\xi_{i,\ell})$.
- **First level**: $\hat{L}_1$ is the $\lfloor p_0 N \rfloor$-th greatest score.
- **Selection**: branch the killed particles on the selected ones.
- **Mutation**: pick r indices $\{\ell_1, \ldots, \ell_r\}$ at random among $\{1, \ldots, m\}$, then for each particle ξ_i

$$\forall \ell_k \in \{\ell_1, \ldots, \ell_r\}, \text{ draw a new } \xi'_{i,\ell_k} \sim \mathcal{B}(p_{\ell_k})$$

Estimation of P_{fa} and P_{fn}


mean – m = 200
Prob
Threshold
10^{0}
10^{-2}
10^{-4}
10^{-6}
10^{-8}
10^{-10}
10^{-12}
10^{-14}
c=2
c=2
c=3
c=3
c=4
c=4