Monte Carlo methods

and Photon Migration

Andrea Farina

andrea.farina@polimi.it

Outlines

- Introduction to Monte Carlo Methods
  - Idea
  - Examples
  - Pseudo-random number generation techniques
- MC for Photon Migration in Tissues
  - Stochastic formulation of the problem
  - CW - approach
  - Time-Resolved approach
- GPU-based MC

What is Monte Carlo?

What are Monte Carlo methods?

Monte Carlo (MC) methods are stochastic techniques, meaning they are based on the *use of random numbers* and probability statistics to investigate problems.

What kind of problems?

- Mathematics: N-dimensional integration and PDE integration.
- Physics sciences: many-body Schrodinger equation, neutron transport, Brownian motion, photon transport, ...
- Engineering: detectors design, illumination problems, ...
- Finance: Black-Scholes equation.

History of Monte Carlo

1733 Buffon's needle problem:

We drop a needle onta a parquet floor. What is the probability that the needle will lie across a line between two strips?

1812 Laplace suggests using Buffon's needle experiment to estimate π.

1930s Enrico Fermi had the idea studying the neutron transport (unpublished).

FERMIAC: a mechanical device for tracing neutron's path.

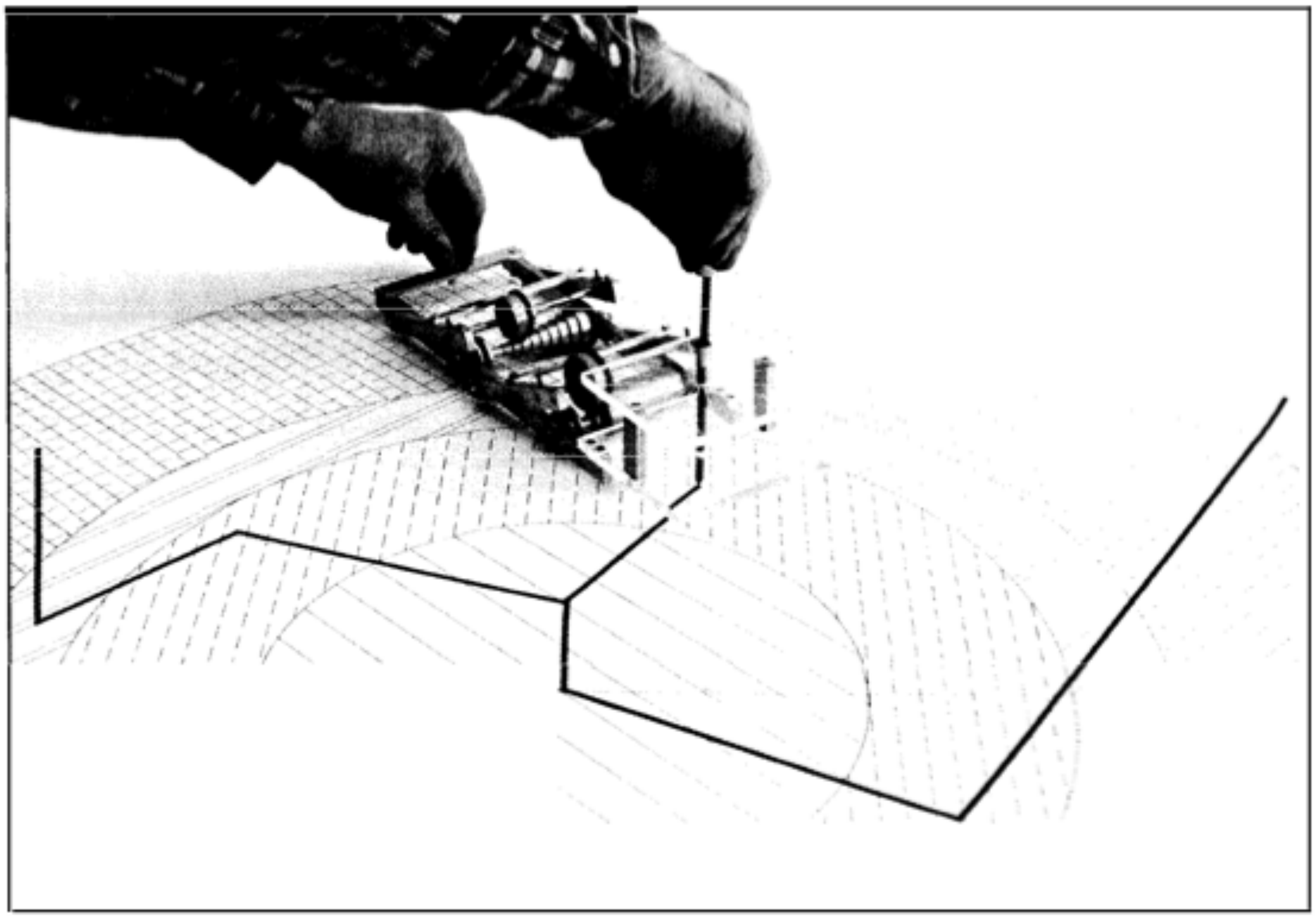
1946 ENIAC (Electronic Numerical Integrator And Computer) built.

1947 John Von Neumann and Stanislaw Ulam propose a computer simulation to solve the problem of neutron diffusion in fissionable material.

1949 Metropolis and Ulam publish their results in the Journal of the American Statistical Association.

"The Beginning of the Monte Carlo Method." Los Alamos Science, No. 15, p. 125, 1987

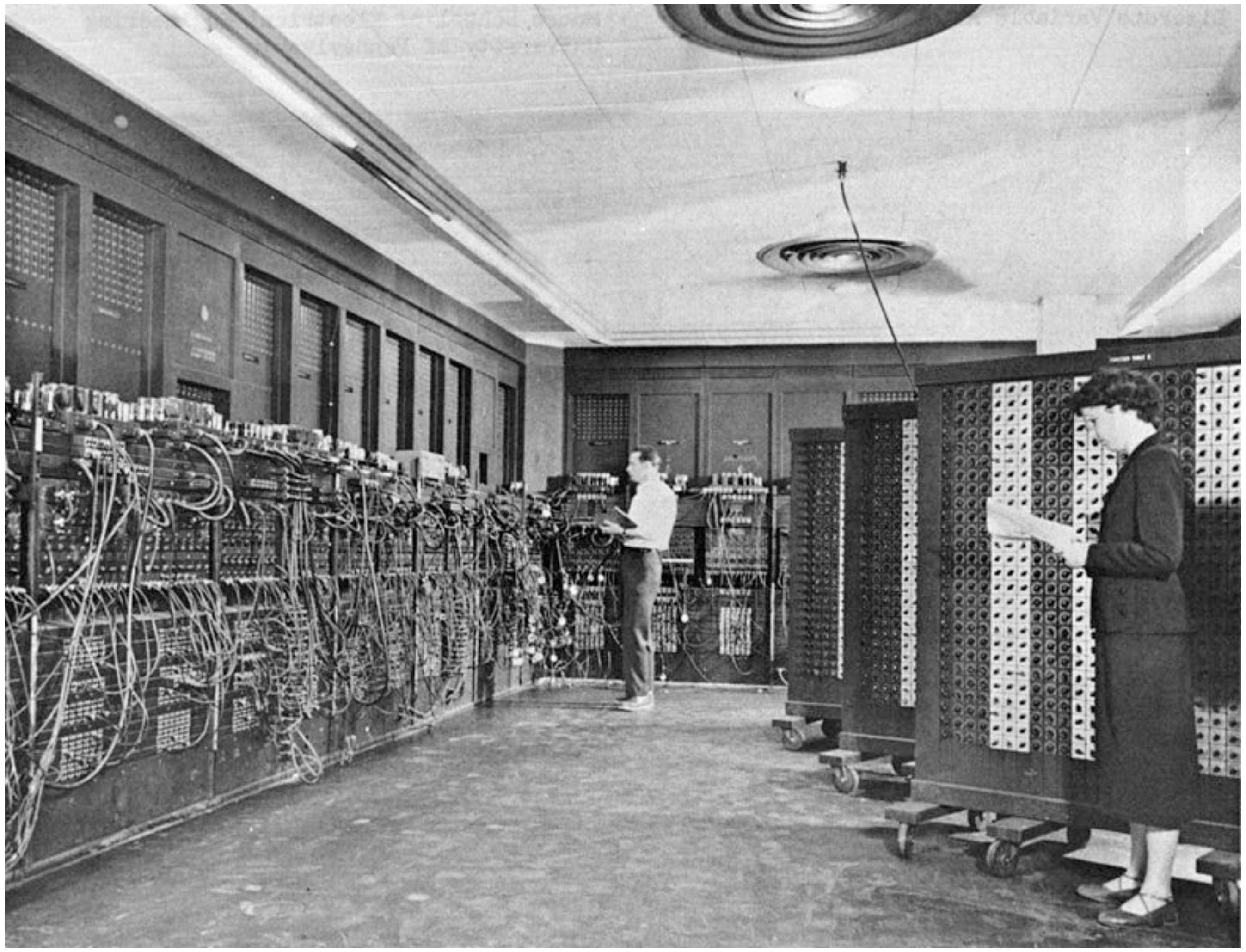
The FERMIAC (1930s)



THE FERMIAC

The Monte Carlo trolley, or FERMIAC, was invented by Enrico Fermi and constructed by Percy King. The drums on the trolley were set according to the material being traversed and a random choice between fast and slow neutrons. Another random digit was used to determine the direction of motion, and a third was selected to give the distance to the next collision. The trolley was then operated by moving it across a two-dimensional scale drawing of the nuclear device or reactor assembly being studied. The trolley drew a path as it rolled, stopping for changes in drum settings whenever a material boundary was crossed. This infant computer was used for about two years to determine, among other things, the change in neutron population with time in numerous types of nuclear systems.

The ENIAC (1946)



ENIAC at the Ballistic Research Laboratory

- 18,000 vacuum tubes
- 7,200 crystal diodes
- 1,500 relays
- 70,000 resistors
- 10,000 capacitors
- 2KW heat dissipation

Memory: 20 numbers of maximum 10 digits!!

Used in the 1947 by Ulam and Von Neumann for solving a problem of neutron transport!

How does it work?

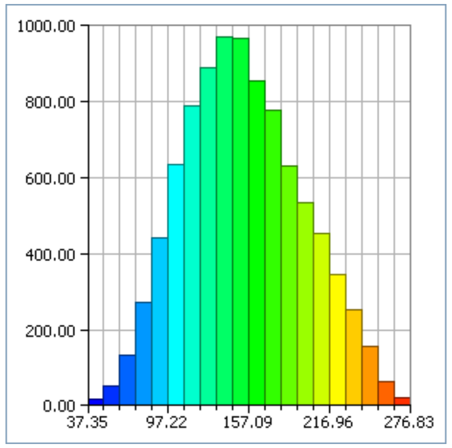
Stochastic formulation of the problem

↓

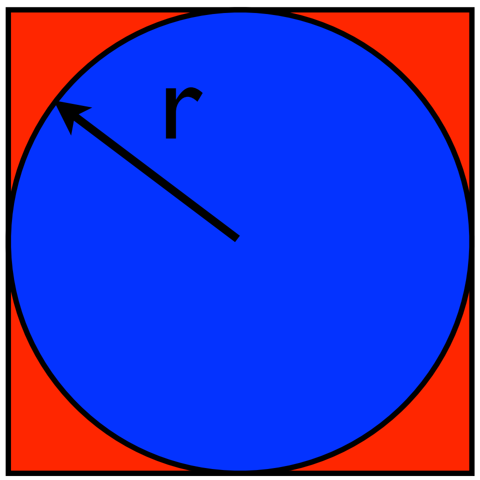
Random number generation to simulate the process

↓

Statistics: data binning, momenta, confidence intervals...

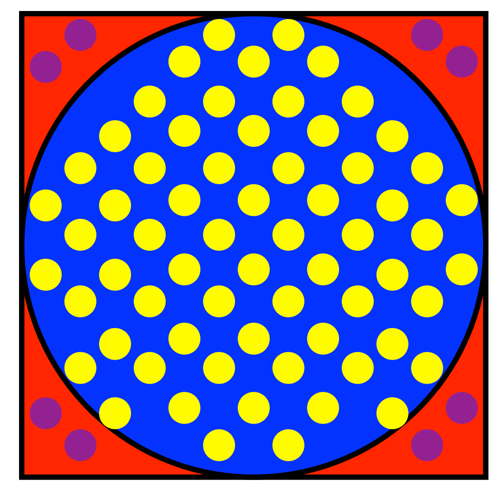


Example 1: π estimation



$$\pi = 4\frac{Area\bigcirc}{Area\square}$$

Uniform probability distribution over the square (raindrops!)



By counting the number of raindrops falling inside the circle...

$$\hat{\pi} = 4\frac{N_{\bigcirc}}{n}$$

$N_{\bigcirc} \sim B(n,p)$ Binomial distribution

$\sigma\left(\hat{\pi}\right) \sim \frac{1}{\sqrt{n}}$ independent of dimension

Binomial distribution

$$X \sim B(n, p)$$

- ***p*** is the success probability.
- ***X*** counts the number of successes over ***n*** trials.

$$P(X = x) = \binom{n}{x} p^x (1-p)^{n-x}$$

$$E[X] = np$$

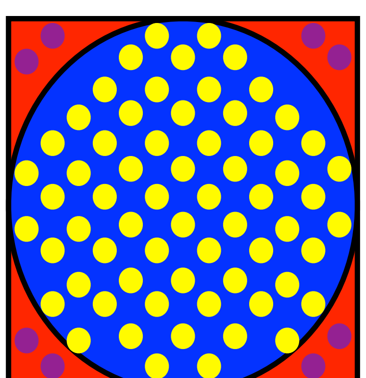
$$Var[X] = np(1-p)$$

x successes over ***n*** trials

$$\hat{p} = \frac{x}{n}$$

$$\sigma^2(\hat{p}) = \frac{1}{n^2}\sigma^2(X) = \frac{\hat{p}}{n}(1-\hat{p})$$

$$N_{\bigcirc} \sim B(n, p)$$



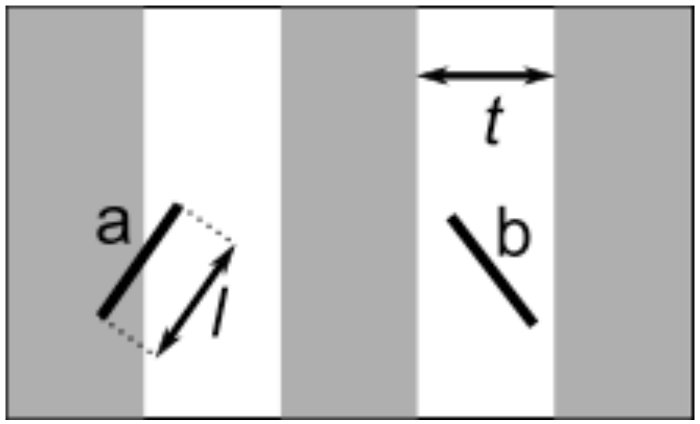
$$\hat{p} = \frac{N_{\bigcirc}}{n}$$

$$\hat{\pi} = 4\frac{N_{\bigcirc}}{n}$$

$$\sigma^2(\hat{\pi}) = \frac{\hat{\pi}(4-\hat{\pi})}{n}$$

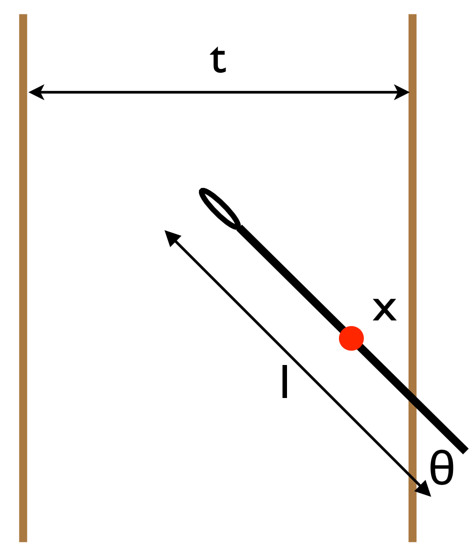
$$\sigma(\hat{\pi}) \sim \frac{1}{\sqrt{n}}$$

Example 2: Buffon’s needle



Shot a needle on a “parquet” floor...
...how much is the probability that
the needle crosses the lines?

Ip: l<t,
x is the distance of the needle center to the nearest line,
θ is the angle between the needle and the line.



$$x \sim \mathscr{U}(0, t/2)$$

$$\theta \sim \mathscr{U}(0, \pi/2)$$

the needle cross the line if

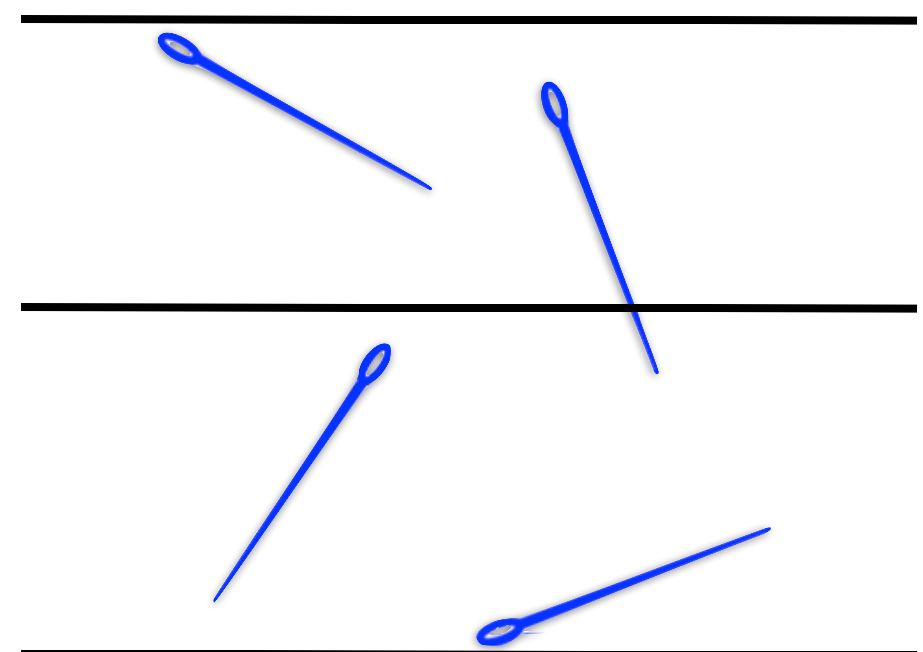
$$x \leq \frac{l}{2}\sin\theta$$

$$P = \int_0^{\frac{\pi}{2}} d\theta \int_0^{l/2\sin\theta} \frac{2}{t}\frac{2}{\pi}dx = \frac{2l}{t\pi}$$

Laplace: “Wow! It’s possible to estimate π by shooting a needle many times on a parquet!!

How to simulate...

How to simulate the "raindrops" falling on the square and the needle falling on the floor?



How to reproduce a random number sequence following a given probability density distribution?

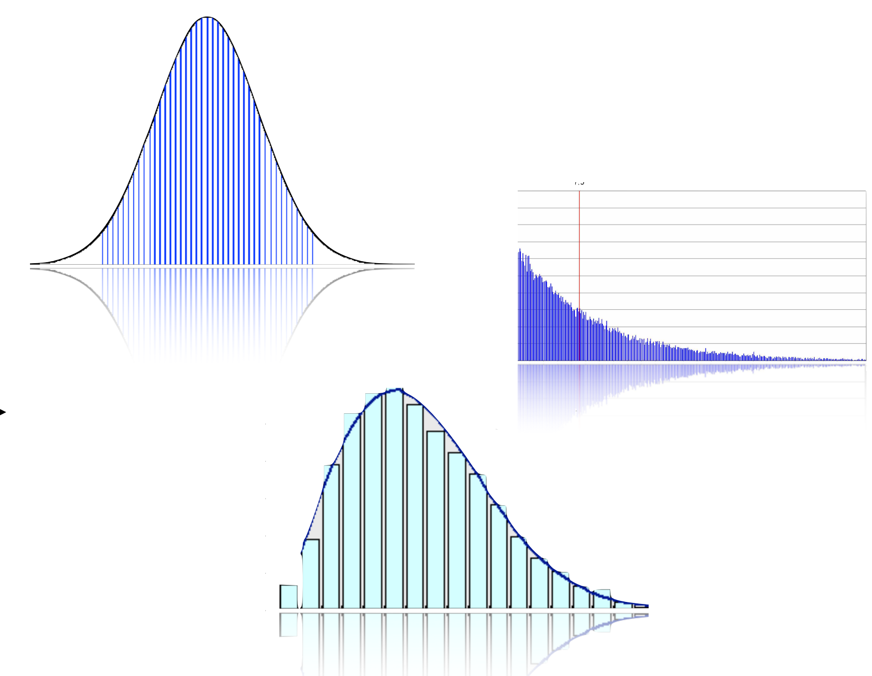
How to simulate...

$$\xi \sim \mathscr{U}(0,1) \xrightarrow{\;\;\;\;\;\;\;\;?\;\;\;\;\;\;\;\;} x(\xi) \sim \mathscr{D}(\lambda)$$

Random number generator (RNG) following a uniform distribution in the interval (0,1).

Generic probability distribution with parameters λ.

0.0012, 0.9212, 0.454, 0.2345, 0.0005, 0.9992, 0.78656, 0.45683, 0.1111, 0.8754, 0.002, 0.0342, 0.0035, ...



Memo

$$F_X(x) = P(X \le x)$$
$$0 \le F_X(x) \le 1$$

Cumulative distribution function (CDF)

$$f(x) = P(x < X < x + dx)$$

Probability density function

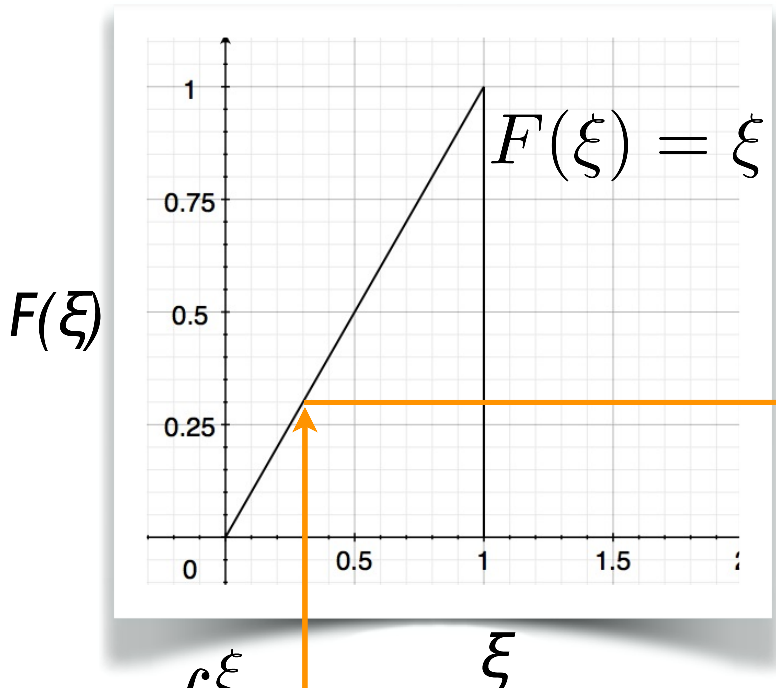
$$F_X(x) = \int_{-\infty}^{x} f(x)\,dx$$

$$f(x) = \frac{dF_X(x)}{dx}$$

$$P(a < x < b) = F(b) - F(a)$$

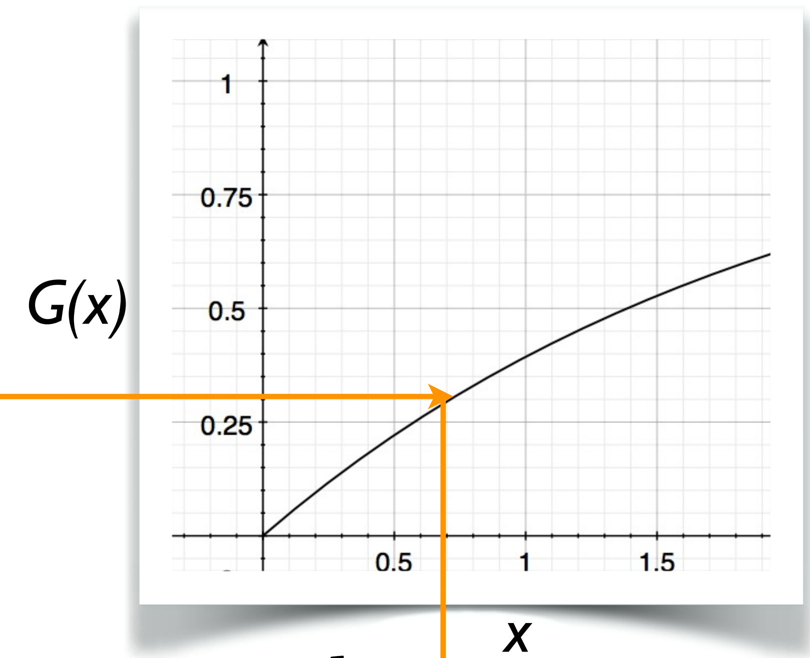
$$P(a < x < b) = \int_{a}^{b} f(x)\,dx$$

How to sample...



CDF

$$F(\xi) = G(x)$$



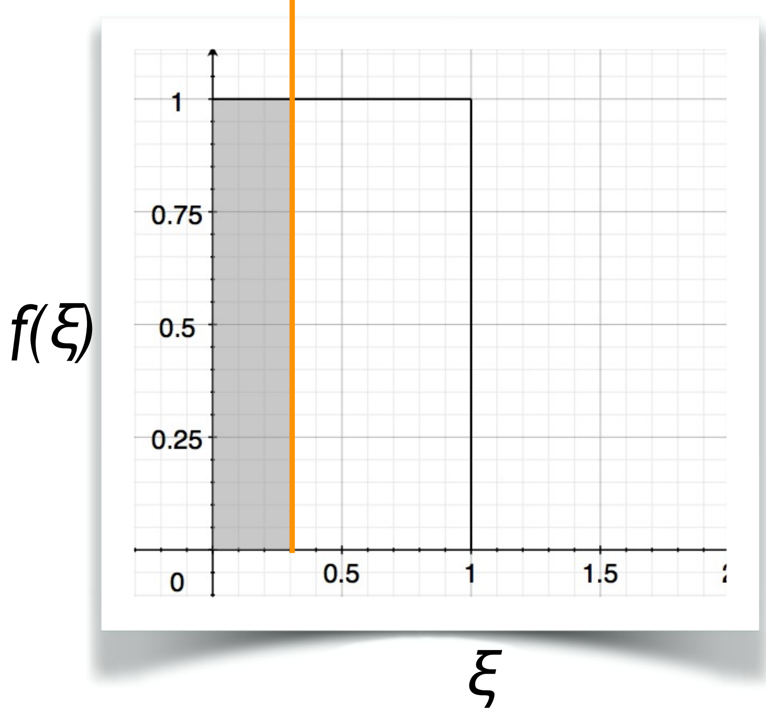
$$\xi = G(x)$$

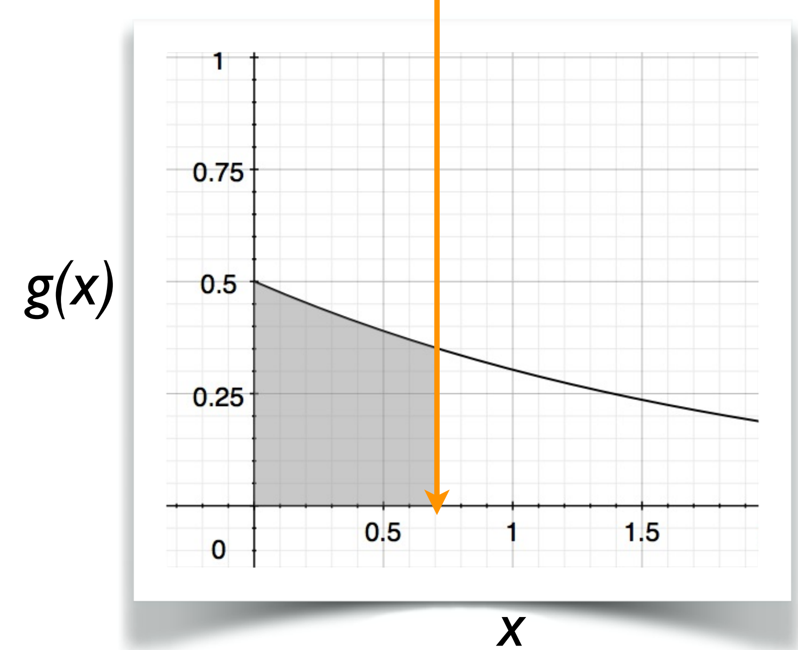
$$\int_{-\infty}^{\xi}$$

$$\frac{d}{dx}$$

$$\boxed{x(\xi) = G^{-1}(\xi)}$$

sampling rule



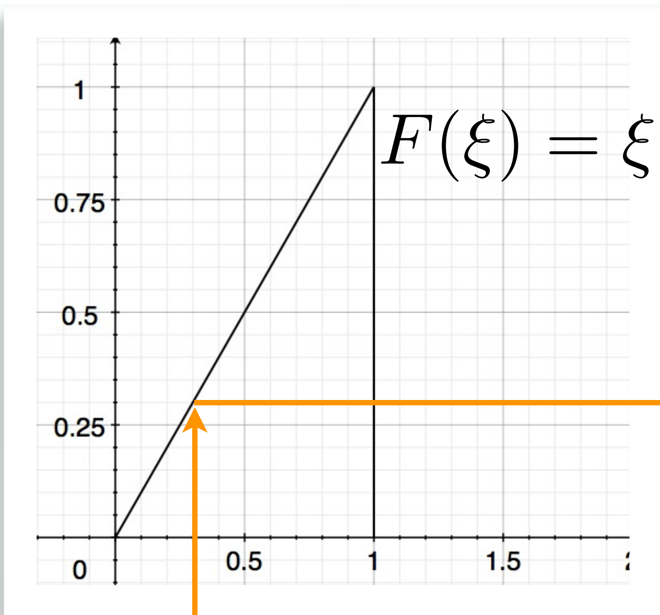


How to sample...

CDF

$$F(\xi) = G(x)$$

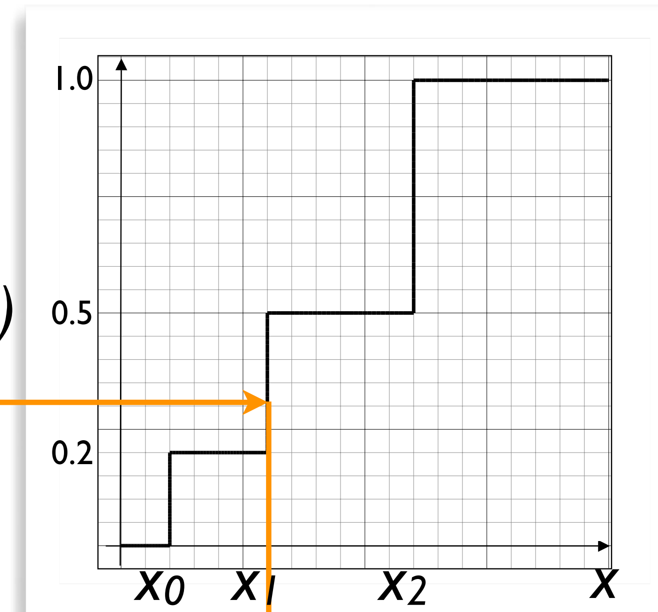
$F(\xi)$



ξ

$$\xi = G(x)$$

$G(x)$



$$\int_{-\infty}^{\xi}$$

$$\frac{d}{dx}$$

$p_0=0.2$
$p_1=0.3$
$p_2=0.5$

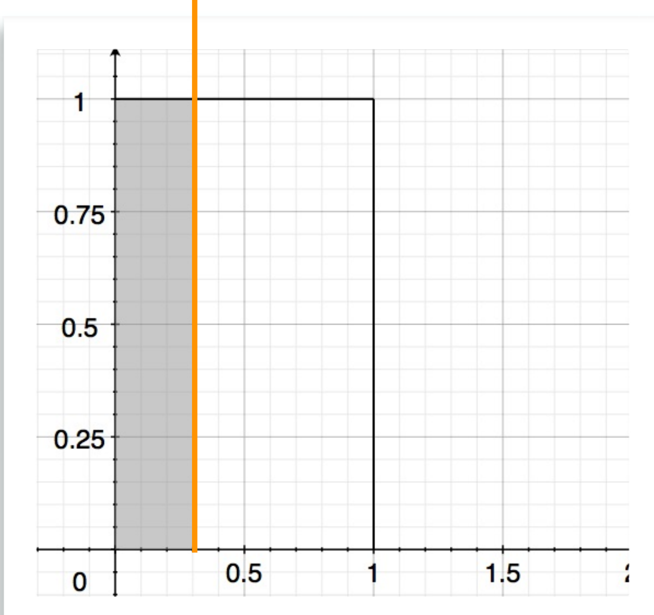
$0 < \xi < 0.2 \rightarrow x = x_1$

$0.2 < \xi < 0.5 \rightarrow x = x_1$

$0.5 < \xi < 1 \rightarrow x = x_2$

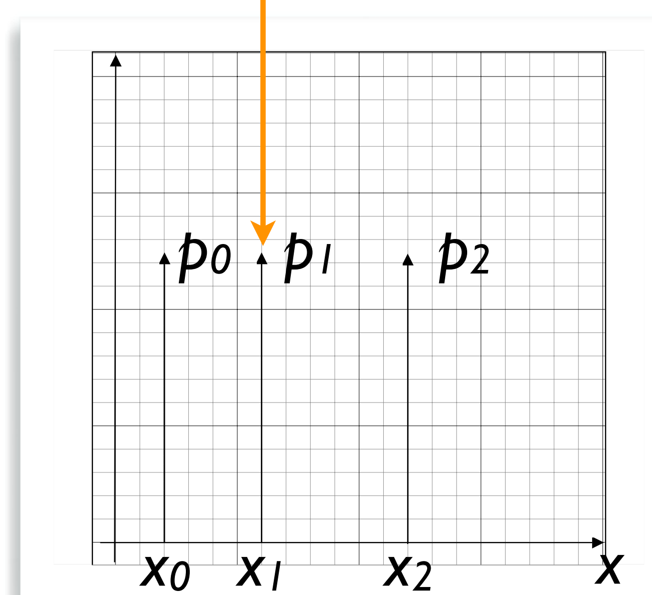
sampling rule

$f(\xi)$



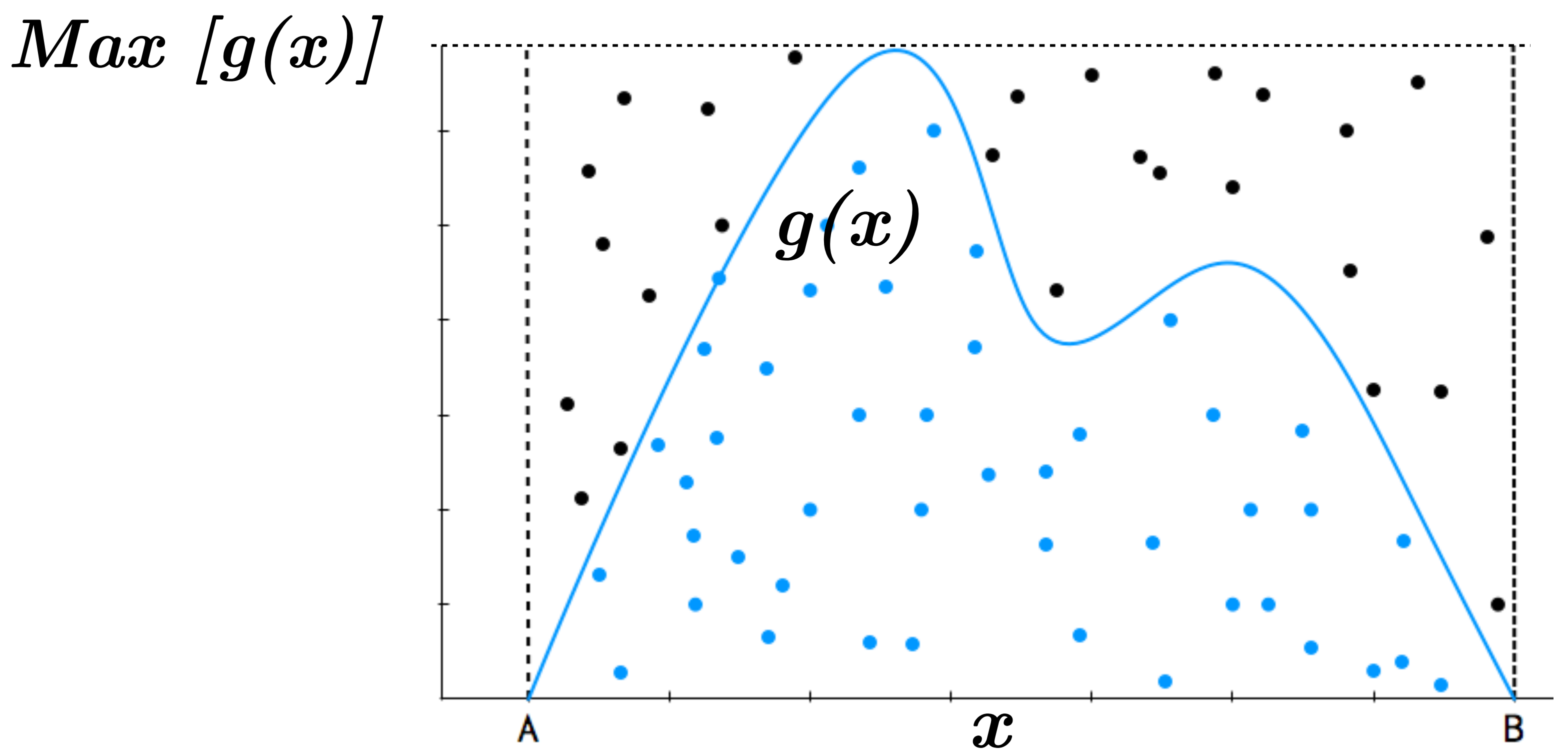
ξ

$g(x)$



Rejection method

The CDF G(x) is not often easy to be inverted…



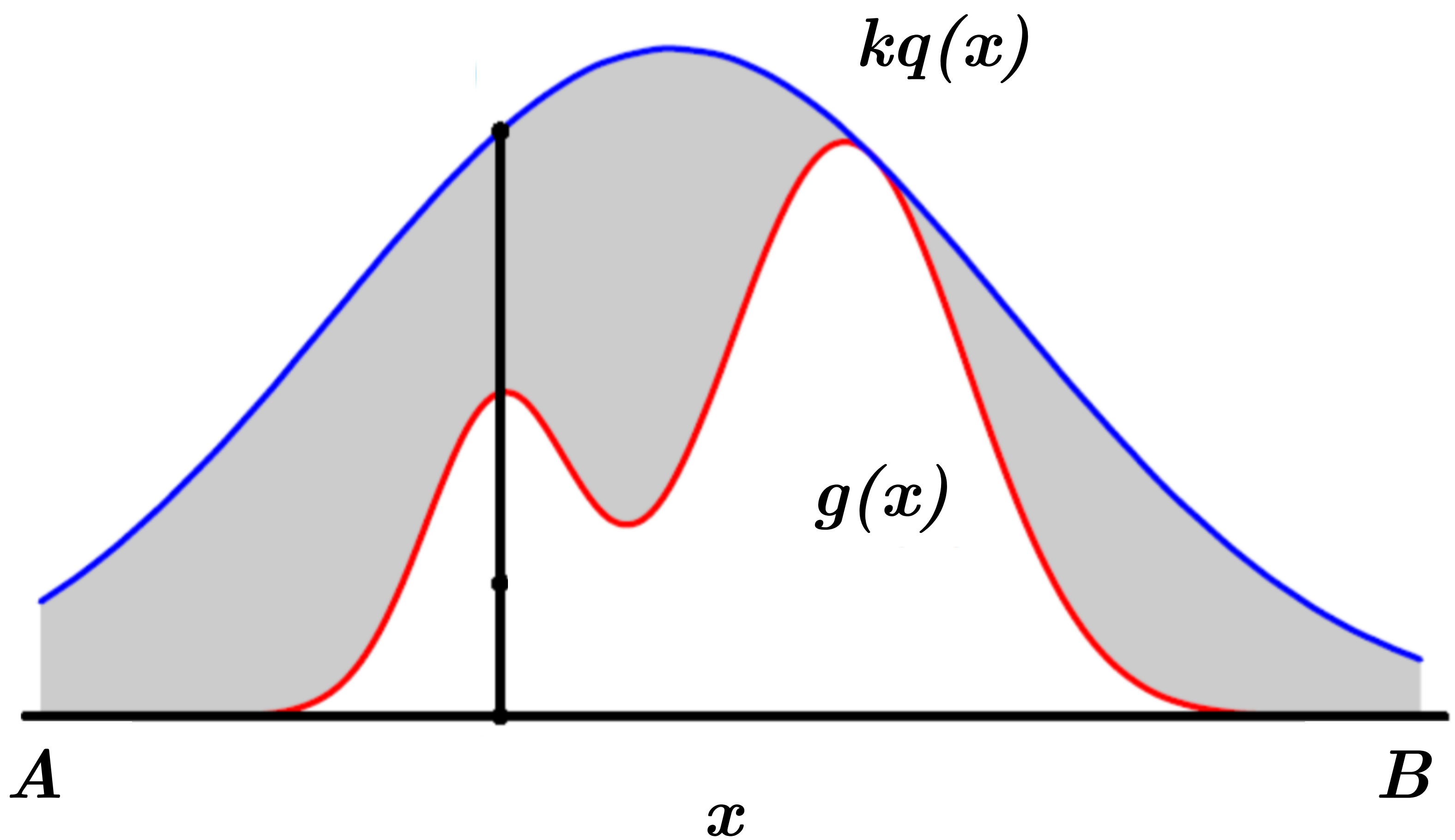
$$x \sim U(A,B)$$

$$y \sim U(0, Max[g(x)])$$

if y is under g(x) accept, otherwise reject

Rejection method

more efficiently…



$$x \sim U(A,B)$$
$$y \sim U(0,kq(x))$$

if y is under g(x) accept, otherwise reject

How to generate random numbers?

- We need to reproduce randomness by a computer algorithm!
- A computer algorithm is *deterministic* in nature.

Pseudo-random number generator (PRNG)

- Pseudo-random number from $U(0, 1)$ will be our only "source of randomness".
- Other distributions can be derived from $U(0, 1)$.

Characterization of a PRNG

The pseudo-random numbers $X_1, X_2, \ldots$ should have the same relevant statistical properties as independent realisations of a $U(0, 1)$ random variable.

- They should reproduce independence ("lack of predictability"): $X_1,\ldots,X_n$ should not contain any discernible information on the next value X_{n+1}.
- The numbers generated should be spread out evenly across (0, 1).

General structure of a PRNG

$$X_n = f(X_{n-1})$$ Number sequence!

$$X_0 \longrightarrow$$ "seed"

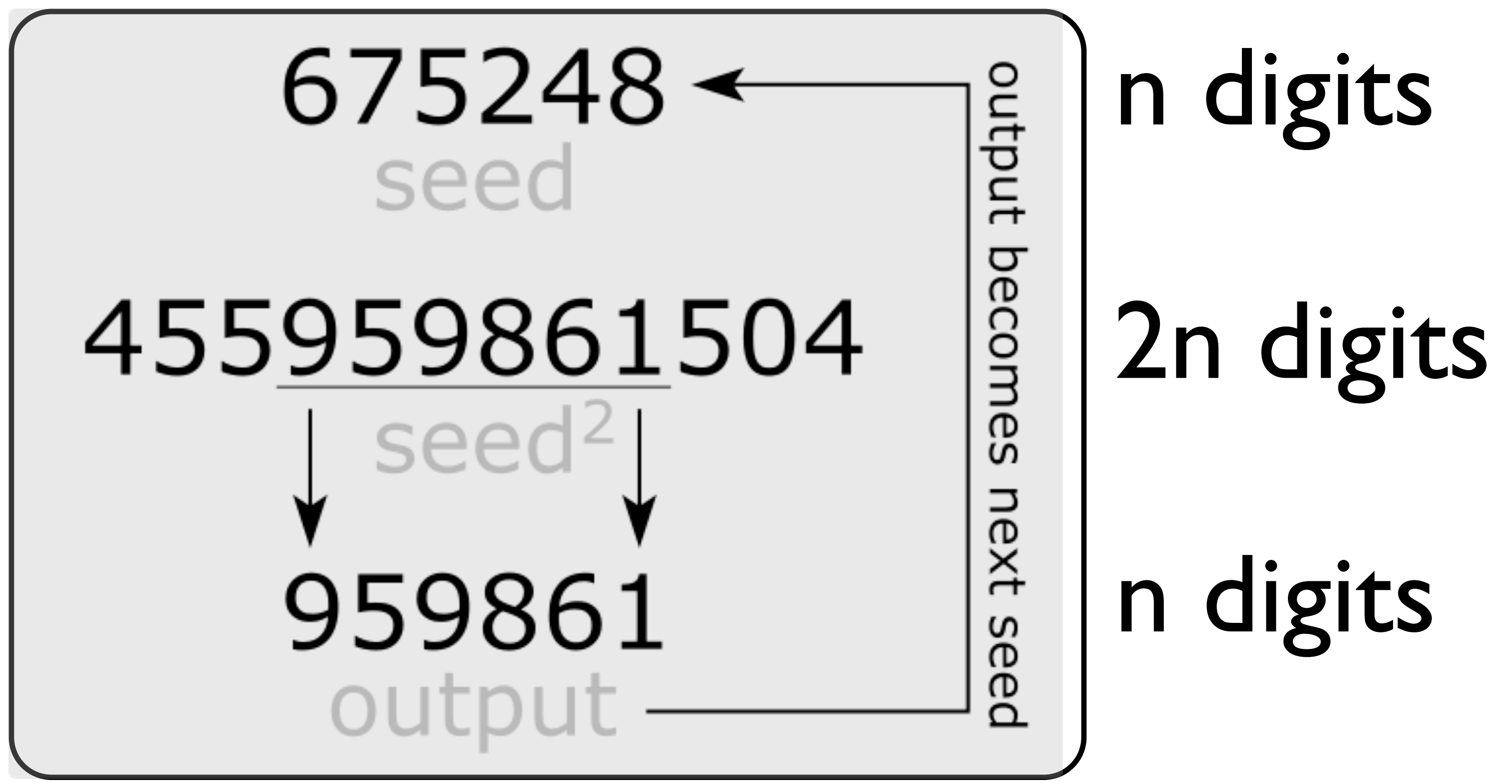
Period

Randomness

Uniformity

PRNG: example 1

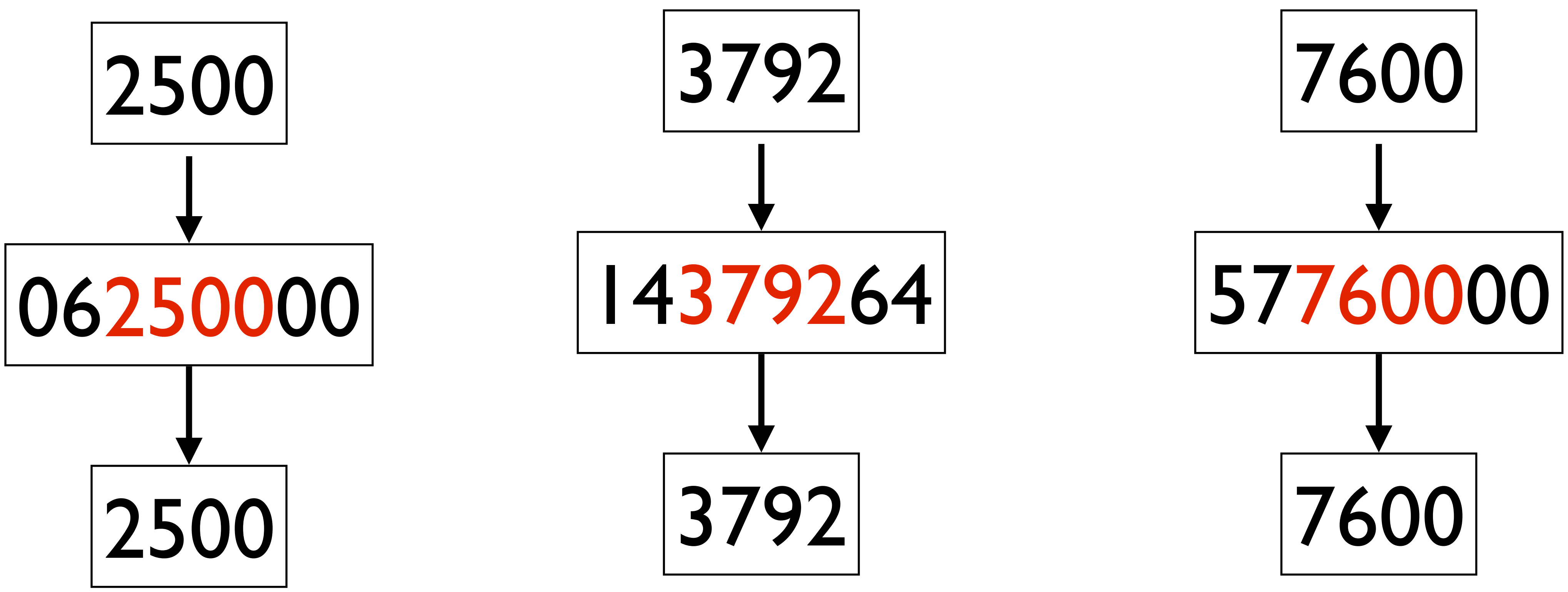
Middle-square method (1949, Von Neumann)



- Used by the ENIAC.
- Short period.
- The method gets stuck with some numbers.

Middle-squared method

"Pathologic cases"



PRNG: example 2

Linear Congruential Generator (1970s)

1. Choose a, $M \in \mathbb{N}$, $c \in \mathbb{N}_0$, and the initial value ("seed") $Z_0 \in \{1, ...M-1\}$.

2. For $n = 1, 2, ...$
   Set $Z_n = (aZ_{n-1} + c) \bmod M$, and $X_n = Z_n/M$

- Period of the string is < M.
- The sequence is DETERMINISTIC: the same *seed* gives the same sequence!
- Historically, poor choices of *a*, *c* and *M* had led to ineffective implementations of LCGs (RANDU).
- Classical built-in function *rand()* in runtime libraries of various compilers uses the LCG with good choices of *a*, *c* and *M*.
- Typically $M=2^{32}, 2^{64}$ for fast modulus operation (bit shifting, bit truncation, ...)

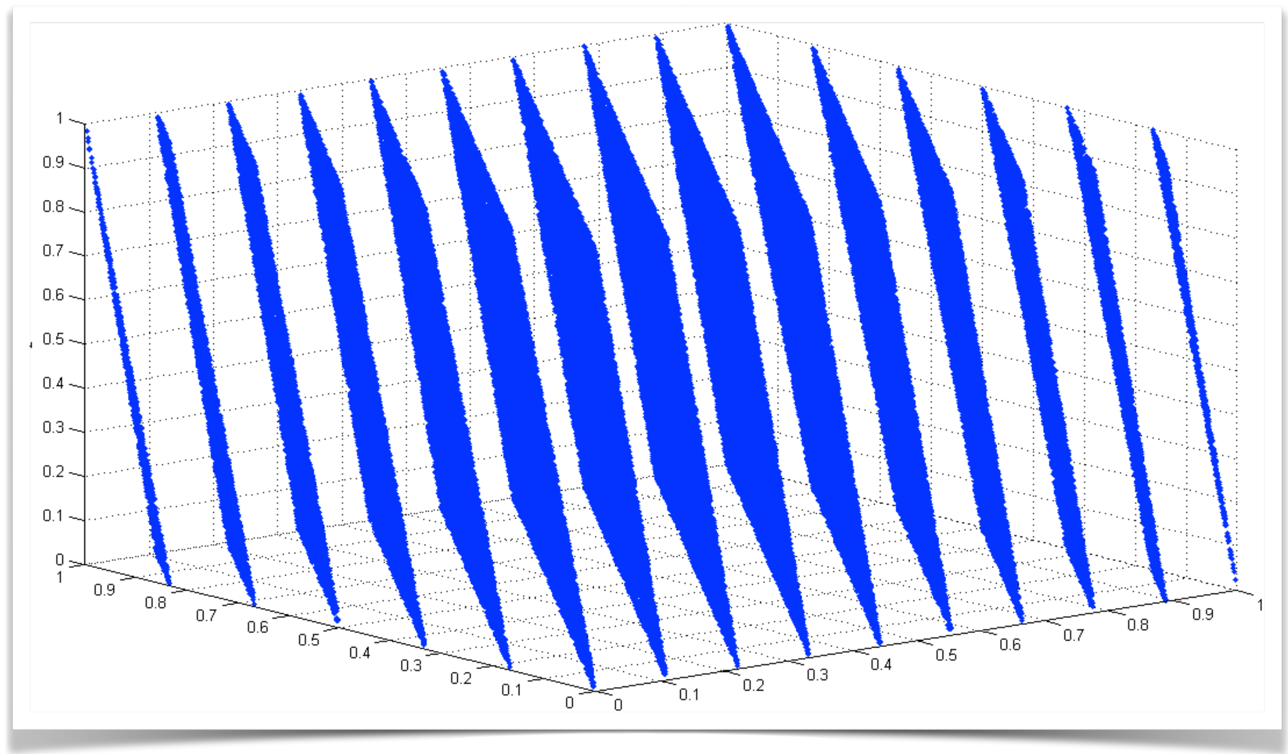
RANDU (1970s)

LCG with:

$$a = 2^{16} + 3$$

$$c = 0$$

$$M = 2^{31}$$



Consecutive triplets lie on 15 hyperplanes in the 3D unitary cube!!

Strong correlation between numbers:
if I know X_k and X_{k+1} I can calculate X_{k+2} !

Linear Congruential Generator

- It’s fast and requires minimal memory (32 or 64 bits) to retain the state.
- There is serial correlation between successive values of the sequence.
- It’s not suitable for Monte Carlo simulations.

PRNG: example 3

Multiply-With-Carry (G. Marsaglia)

1. Choose a, $M, c_0 \in \mathbb{N}$, and the initial value ("seed") $Z_0 \in \{1, \dots M-1\}$. Let $c_0 < a$.

2. For $i = 1, 2, \ldots$
   Set $Z_i = (aZ_{i-1} + c_{n-1}) \bmod M$, where
   $c_n = \lfloor \dfrac{aZ_{n-1} + c_{n-1}}{M} \rfloor$ (quotient)

3. Set $X_i = Z_i / M$

- Typically $M=2^{32}, 2^{64}$ for fast modulus operation (bit shifting, bit truncation, ...)

- The choice of the multiplier *a* and of the base *M* determines the period.

G. Marsaglia. Random number generators. J. Mod. Appl. Stat. Meth., 2(1):2–13, 2003

Multiply-With-Carry

Property

If $(a \cdot M - 1)$ is "safeprime"
then the period of a MWC PRNG is $(a \cdot M - 2)$.

A number q is "safeprime"
when both q and $(q-1)/2$ are prime.

Example: a slightly $<2^{32}$, $M=2^{32} \Rightarrow$ period $\sim 2^{62}$!!!!

Multiply-With-Carry

- It's fast and requires minimal memory (32 or 64 bits) to retain the state.
- It passed statistical tests of uniformity and randomness.
- It's a good candidate for Monte Carlo simulations.

Other PRNGs

- Complimentary Multiply-With-Carry.
- Mersenne-Twister (colossal period: $2^{19937}-1$).
- Lagged Fibonacci.
- Xorshift.

How does it work?

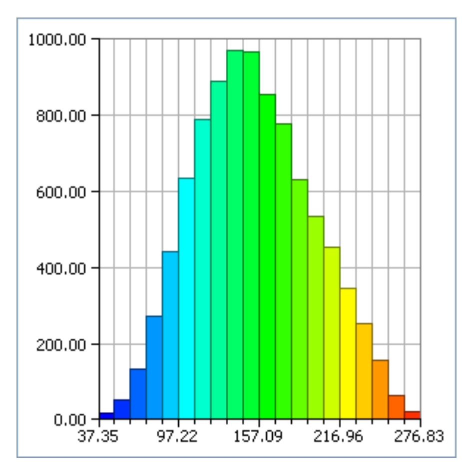
Stochastic formulation of the problem

↓

Random number generation to simulate the process

↓

Statistics: data binning, momenta, confidence intervals...



MC methods for Photon Migration in Diffusive Media

- MC simulations are used to test theoretical models (DE, RTE, PDW, ...)
- MC simulations are suitable for any geometry and can be used to generate forward data.
- Can we used MC for inverse problems?

How does it work?

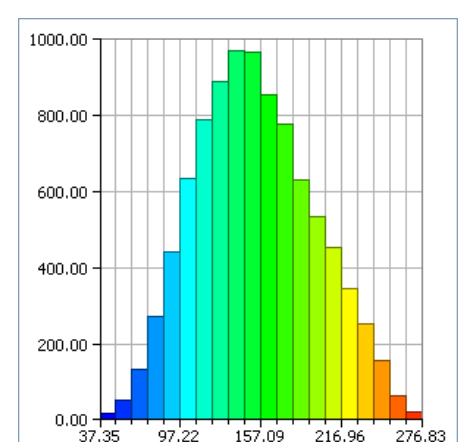
Stochastic formulation of the problem

↓

Random number generation to simulate the process

↓

Statistics: data binning, momenta, confidence intervals...



What do we know?

μ_a — Absorption coefficient: the probability for unit-length that a photon is absorbed.

μ_s — Scattering coefficient: the probability for unit-length that a photon is scattered.

$p(\vartheta, \vartheta')$ — Phase function: the probability that a photon coming along the direction θ is scattered along a direction θ'.

The RTE introduces probabilities to describe a DETERMINISTIC physical process!

Light propagation inside a diffusive medium IS NOT A STOCHASTIC PROCESS!

1: photon's step size

What's the probability distribution for the step size?

$$\mu_t = \mu_s + \mu_a$$

Interaction coefficient: the probability for unit-length that a photon is scattered or absorbed.

$$S$$

Random variable indicating the photon's free path.

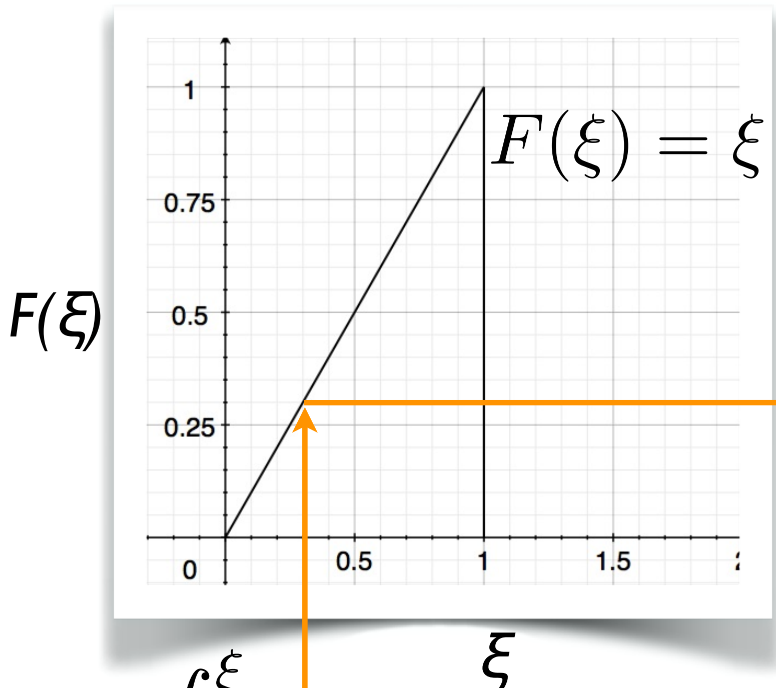
$$P(s < S < s + \mathrm{d}s) = \underbrace{P(S > s)}_{\text{Probability that the free path is >s}}\underbrace{\mu_t \mathrm{d}s}_{\text{Interaction probabilty in ds}} = P(S < s + \mathrm{d}s) - P(S < s) = \mathrm{d}P(S < s) \Rightarrow$$

$$\Longrightarrow P(S > s)\mu_t \mathrm{d}s = -\mathrm{d}P(S > s) \Longrightarrow \mu_t \mathrm{d}s = -\frac{\mathrm{d}P(S > s)}{P(S > s)}$$

$$\Longrightarrow P(S > s) = \exp(-\mu_t s) \Longrightarrow P(S < s) = 1 - \exp(-\mu_t s)$$

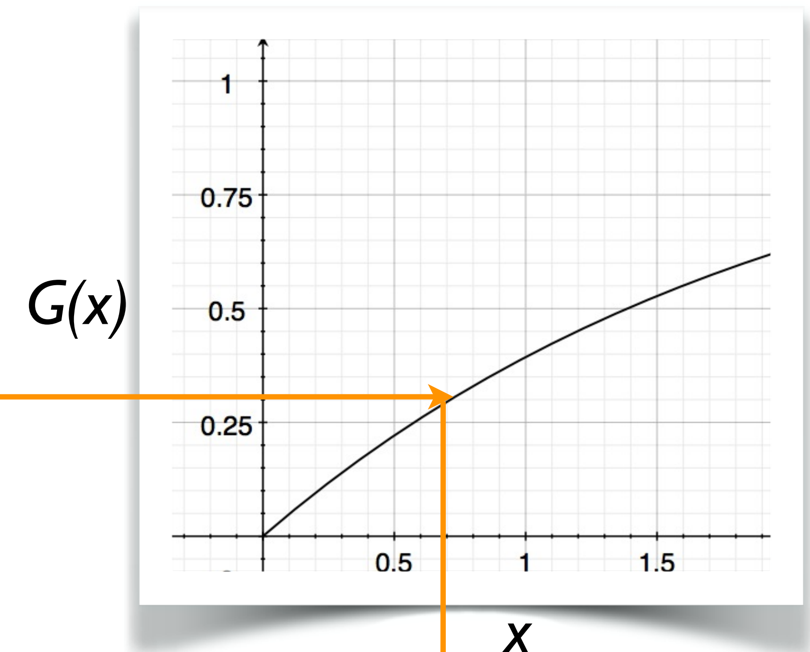
Exponential distribution

1: sampling the step size



CDF

$$F(\xi) = G(x)$$



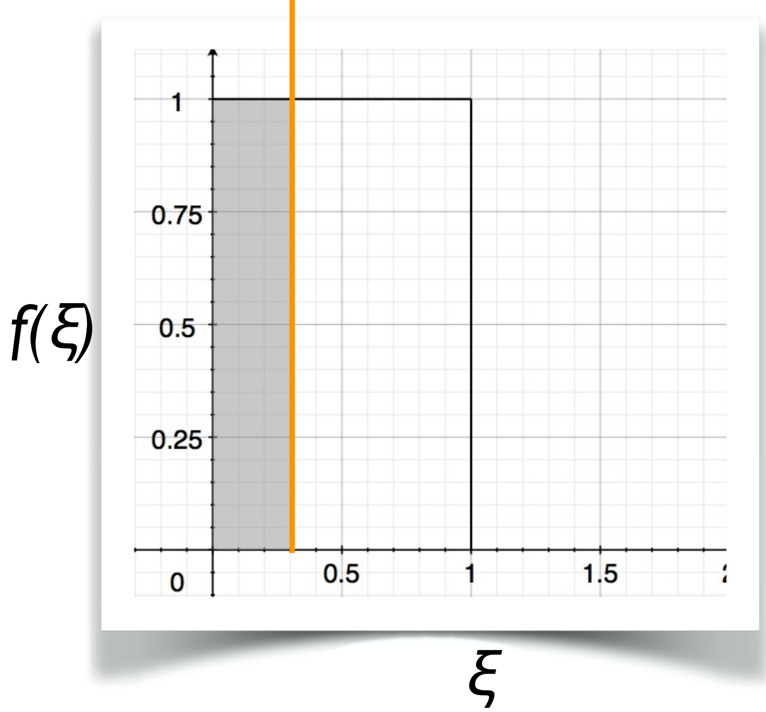
$$\int_{-\infty}^{\xi}$$

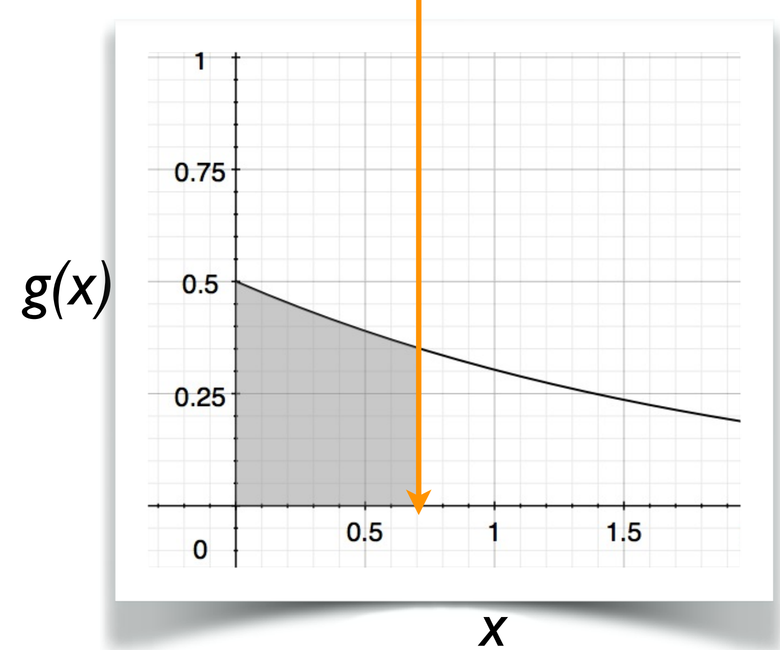
$$\xi = 1 - \exp(-\mu_t s)$$

$$\frac{d}{dx}$$

$$s = -\frac{\log(1-\xi)}{\mu_t}$$

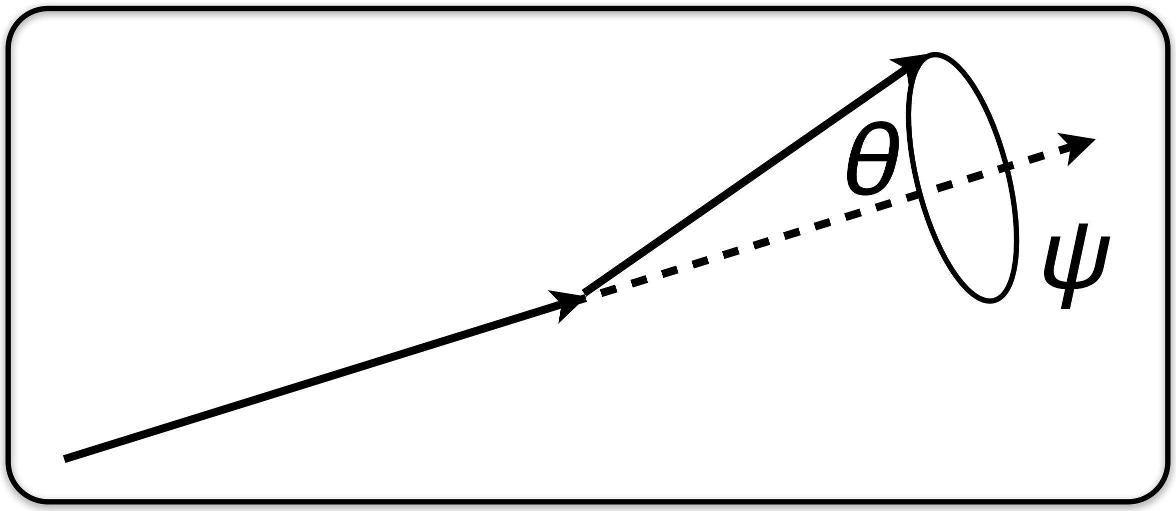
$$\boxed{s = -\frac{\log(\xi)}{\mu_t}}$$





2: photon scattering

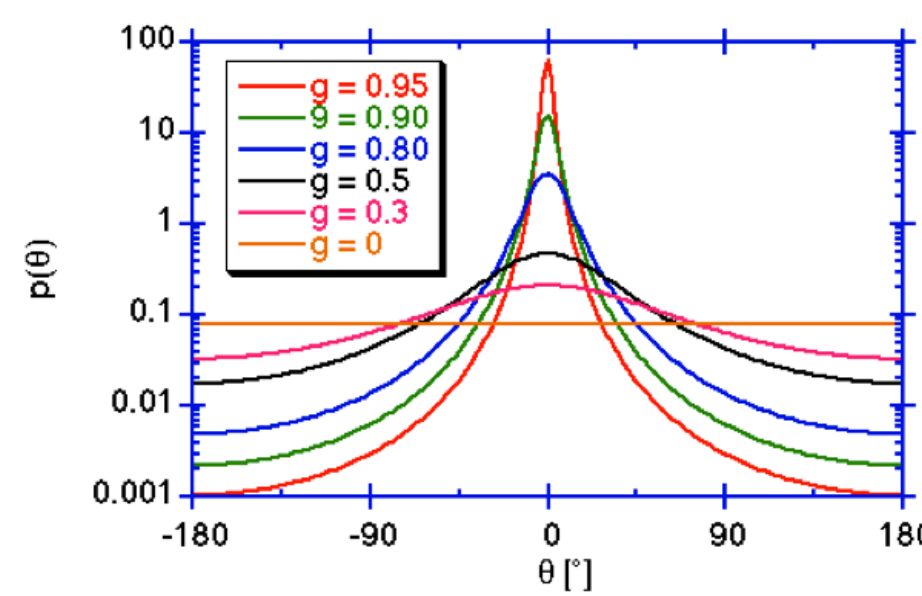
What's the probability distribution for the deflection and azimuthal angles?



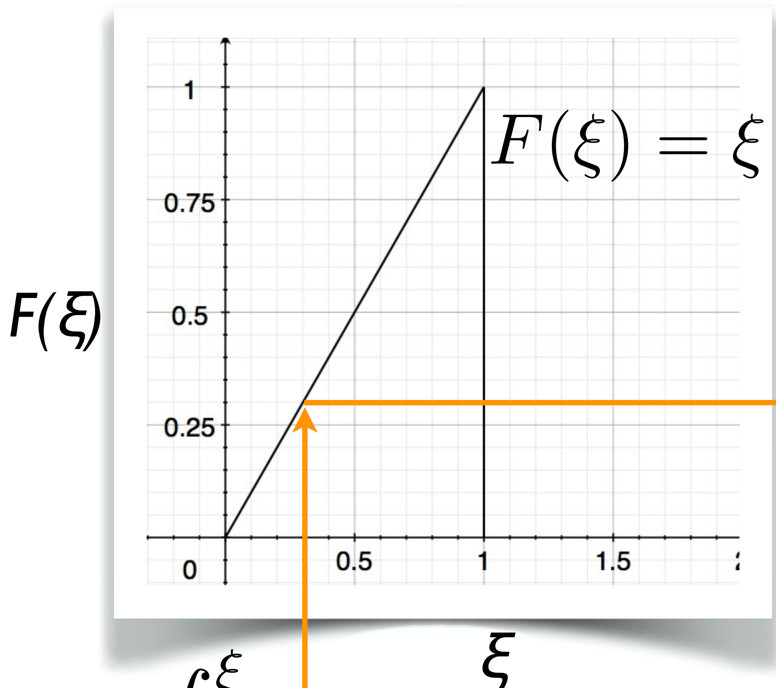
$$p(\vartheta) = \frac{1}{4\pi} \frac{1 - g^2}{(1 + g^2 - 2g\cos\vartheta)^{3/2}}$$

$$g = < \cos\vartheta >$$ Henyey-Greenstein

$$p(\psi) = \frac{1}{2\pi}$$ Uniform distribution



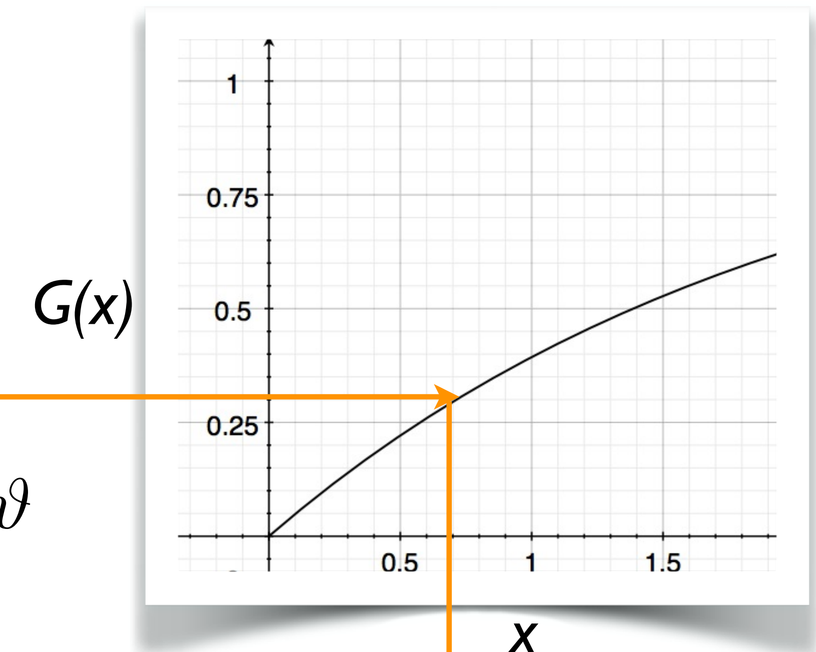
2: sampling θ



CDF

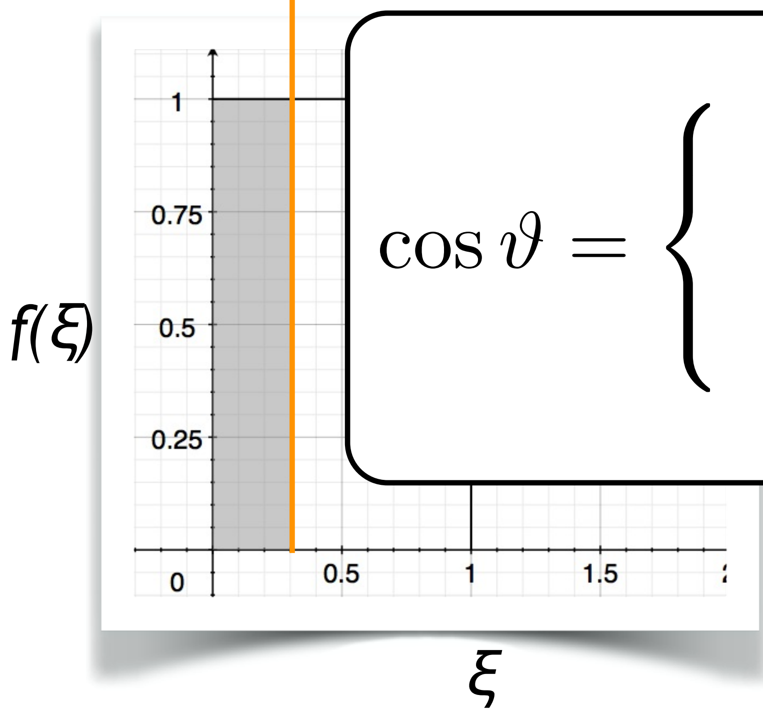
$$F(\xi) = G(x)$$

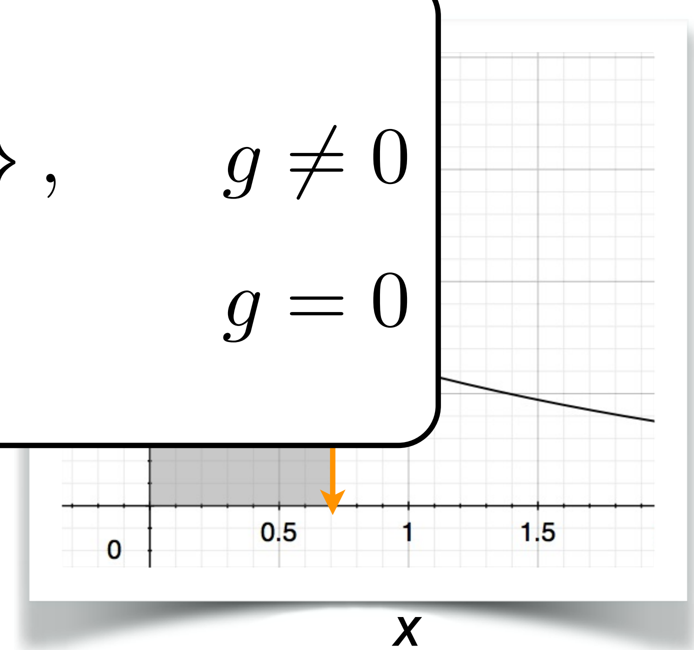
$$\xi = 2\pi \int_0^{\vartheta} p(\vartheta) \sin\theta \mathrm{d}\vartheta$$



$$\int_{-\infty}^{\xi}$$

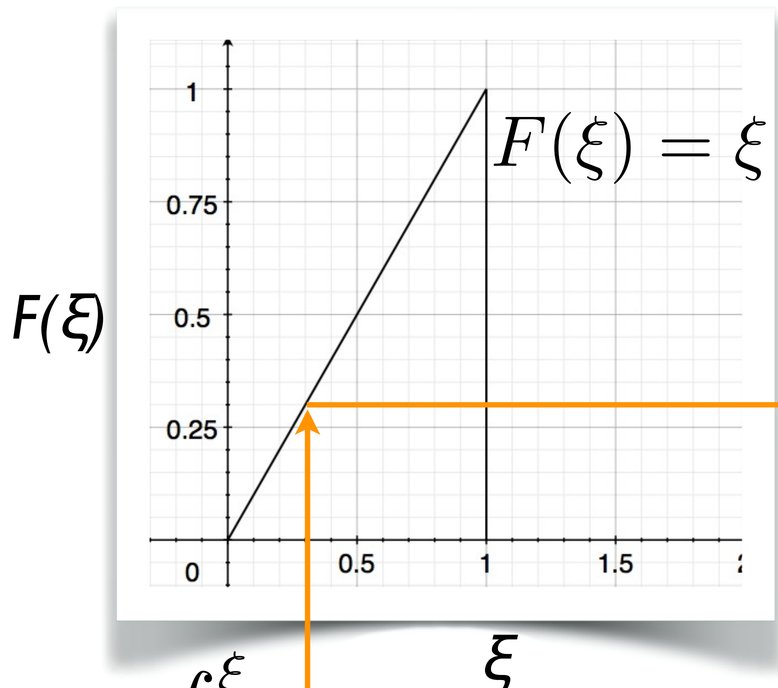
$$\frac{d}{dx}$$



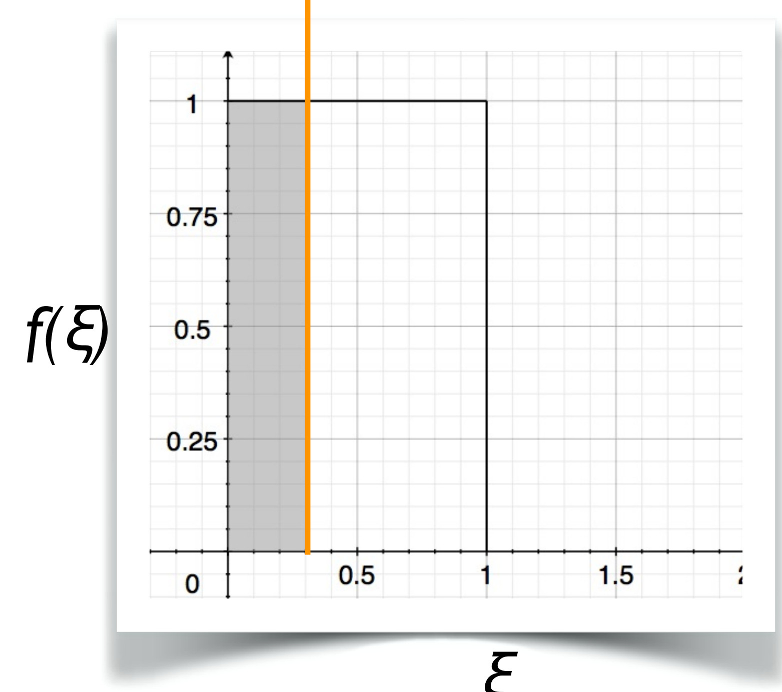


$$\cos\vartheta = \begin{cases} \frac{1}{2g}\left\{1 + g^2 - \left[\frac{1-g^2}{1-g+2g\xi}\right]^2\right\}, & g \neq 0 \\ 2\xi - 1, & g = 0 \end{cases}$$

2: sampling ψ



$$\int_{-\infty}^{\xi}$$



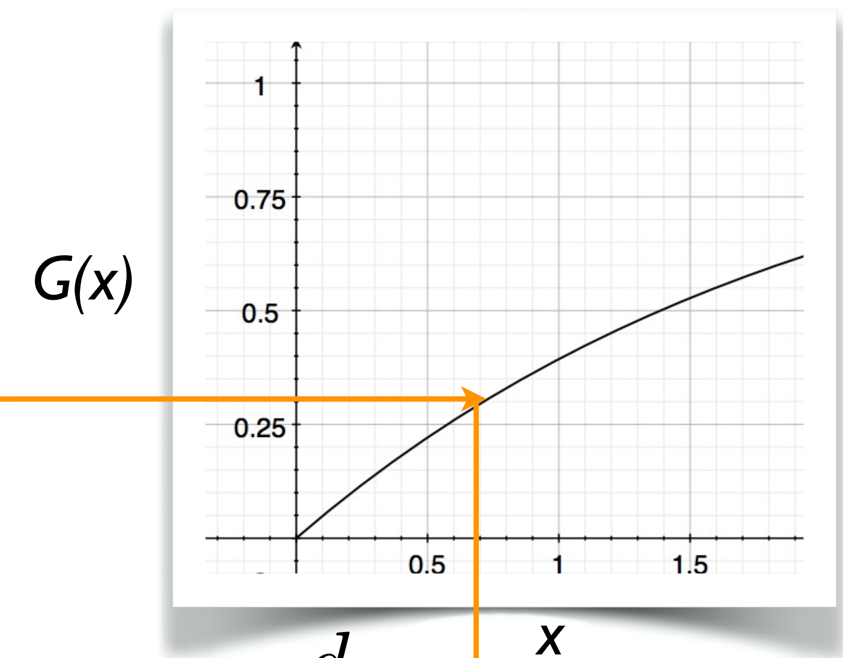
CDF

$$F(\xi) = G(x)$$

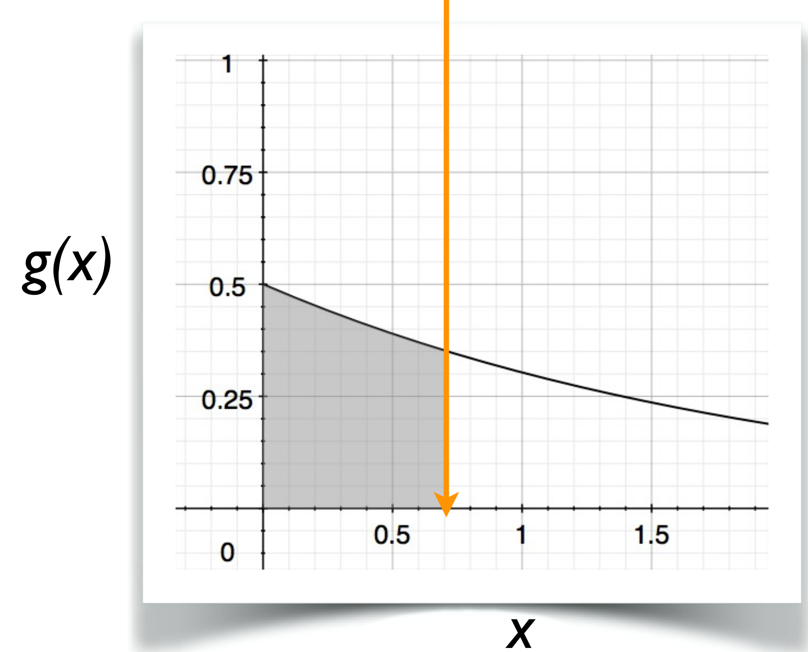
$$\xi = \int_0^{\psi} p(\psi)\mathrm{d}\psi$$

$$\xi = \int_0^{\psi} \frac{1}{2\pi}\mathrm{d}\psi$$

$$\boxed{\psi = 2\pi\xi}$$



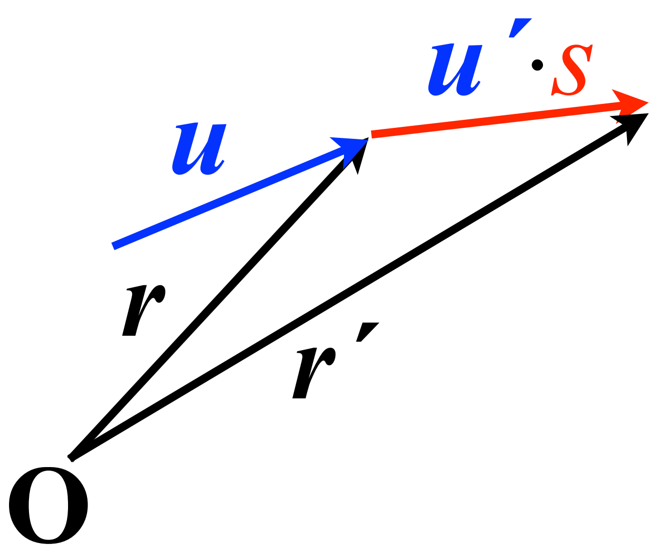
$$\frac{d}{dx}$$

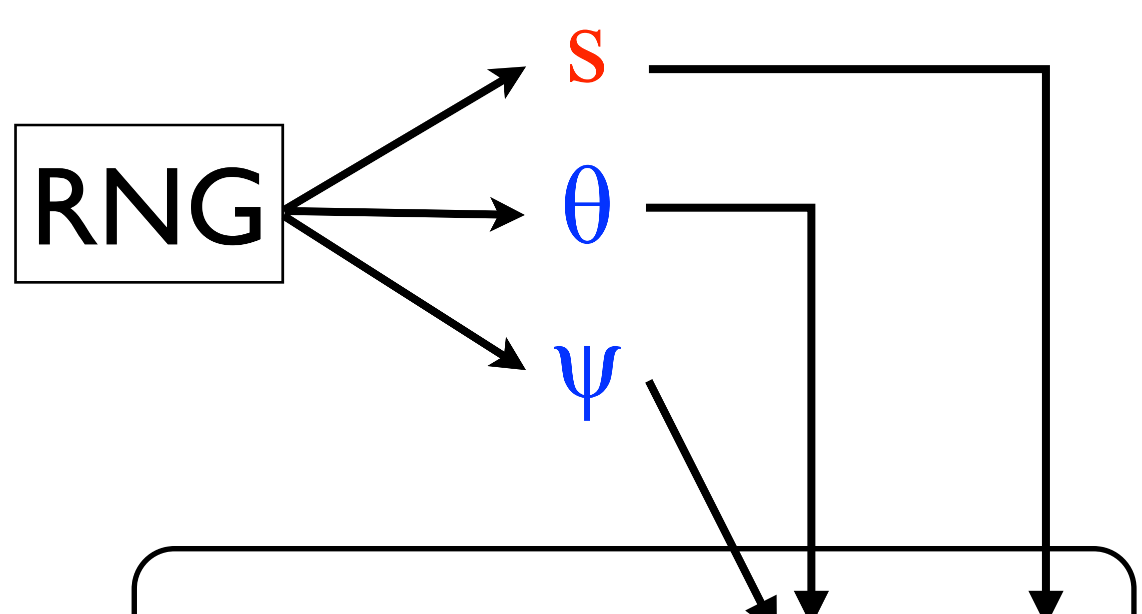


3: Let's move the photon!

Photon → $\mathbf{r}(x, y, z)$ Position vector

Photon → $\mathbf{u}(u_x, u_y, u_z)$ Direction unit vector





$$\boxed{\mathbf{r}' = \mathbf{r} + \mathbf{u}' \cdot s}$$

New photon's position

4: Absorption

- Photons start propagating with a weight W=1.
- After each interaction the weight is reduced by a factor equal to the albedo:

$$W' = W\frac{\mu_s}{\mu_t}$$

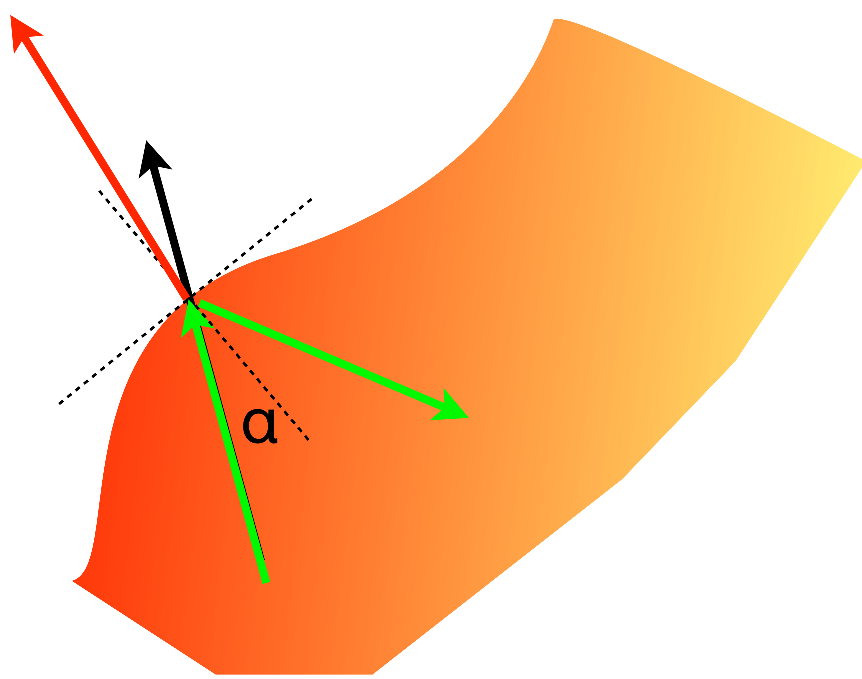
- After N interactions the weight becomes:

$$W = \left(\frac{\mu_s}{\mu_t}\right)^N$$

"Albedo-weight" method

4: Transmission at boundary

When the photon hits the boundary it can be back reflected or transmitted.



Photon is propagated to the boundary by means of a shortened step size.

↓

The incidence angle α and the Fresnel reflection coefficient R(α) are computed.

↓

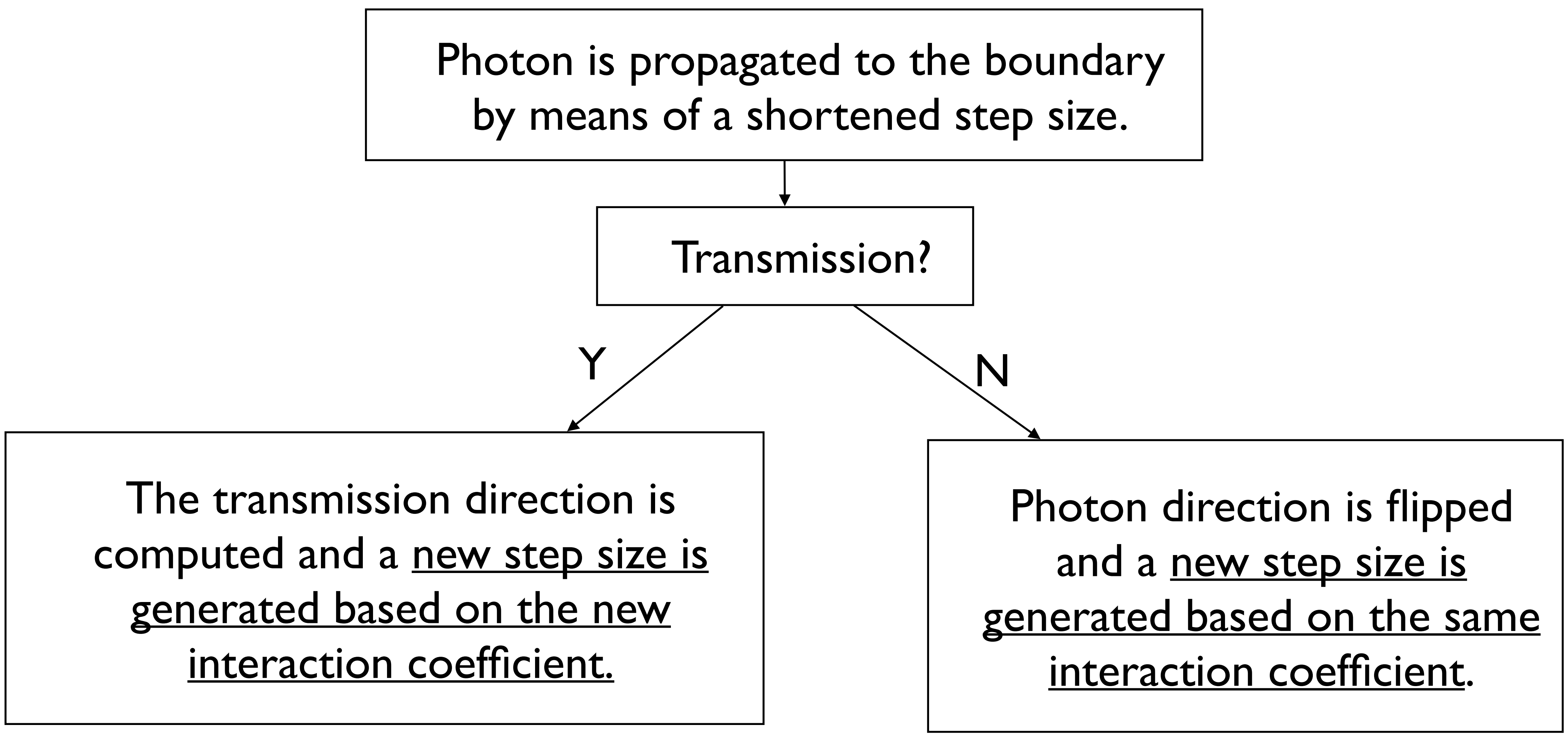
α>TIR angle?

Y → Photon is back reflected.

N → A RND ξ is generated. → <u>If</u> ξ<R(α) the photon is back reflected, <u>else</u> is transmitted.

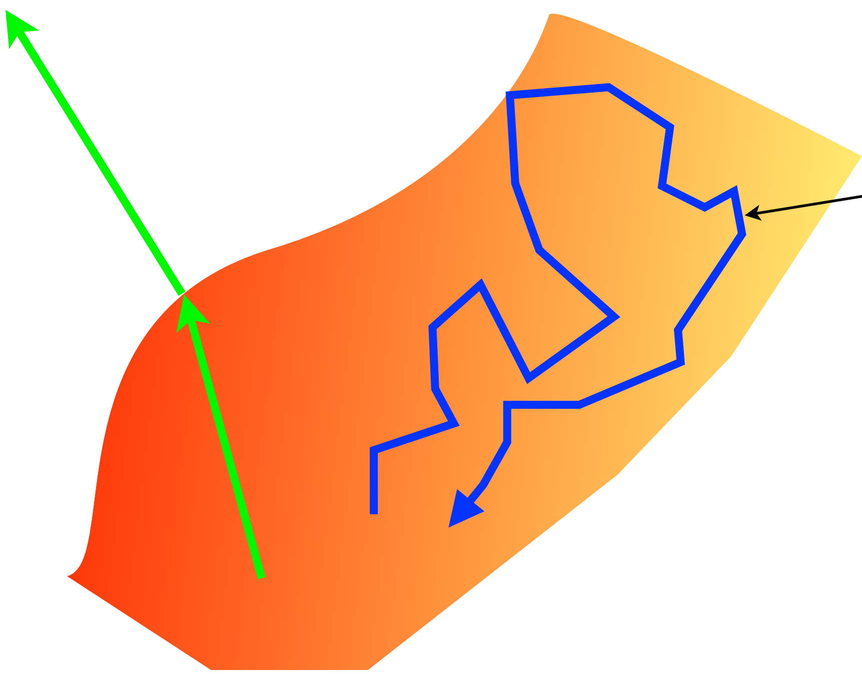
5: Interaction at interface

When the photon hits the boundary between two turbid media...



6: Photon termination

A Photon can be terminated naturally by transmission or it can still propagating inside the tissue...



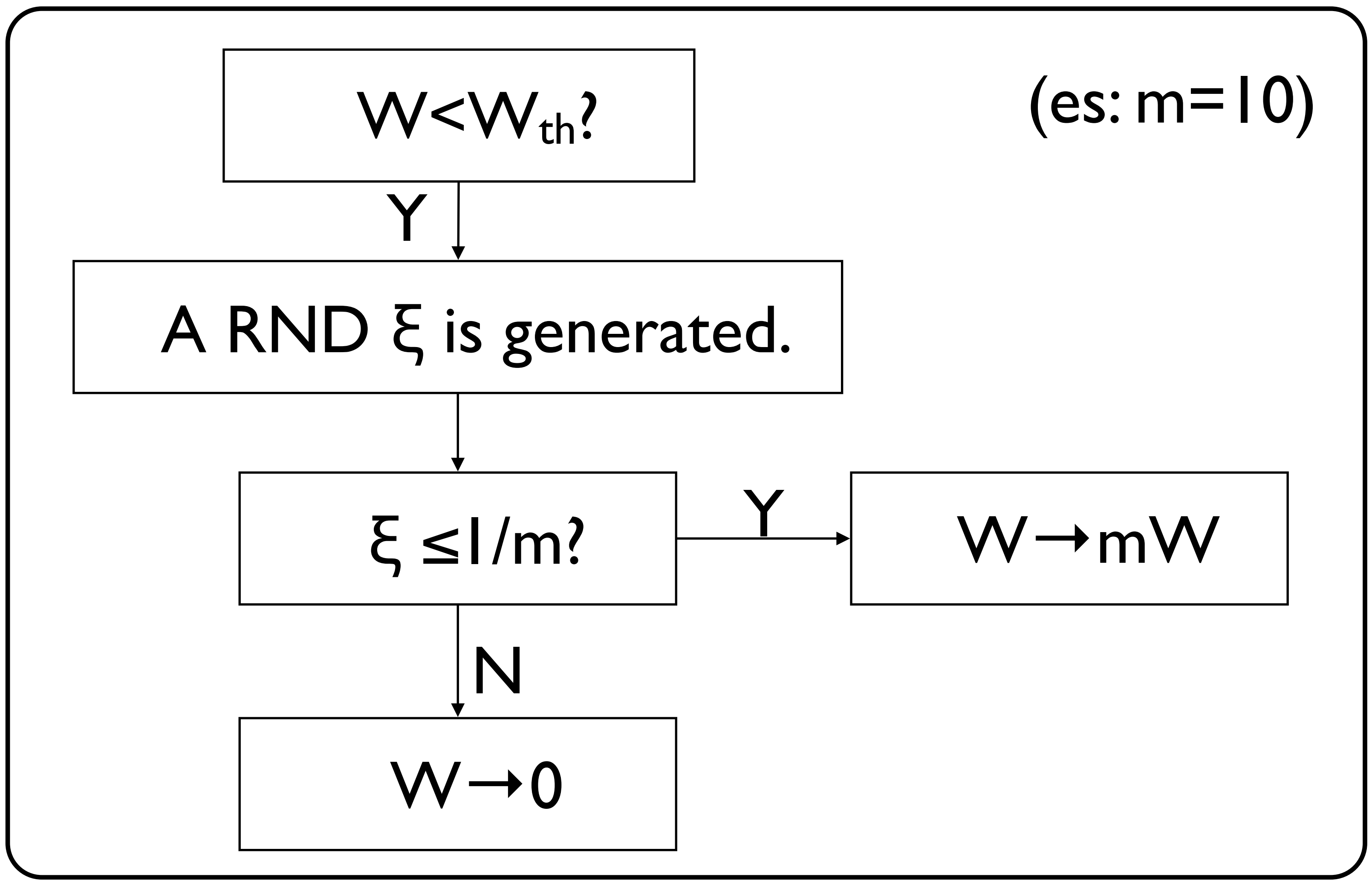
how to terminate this photon?

We can fix a threshold for the weight W_{th} and kill the photon if $W<W_{th}$ but...

...total energy is not conserved!!

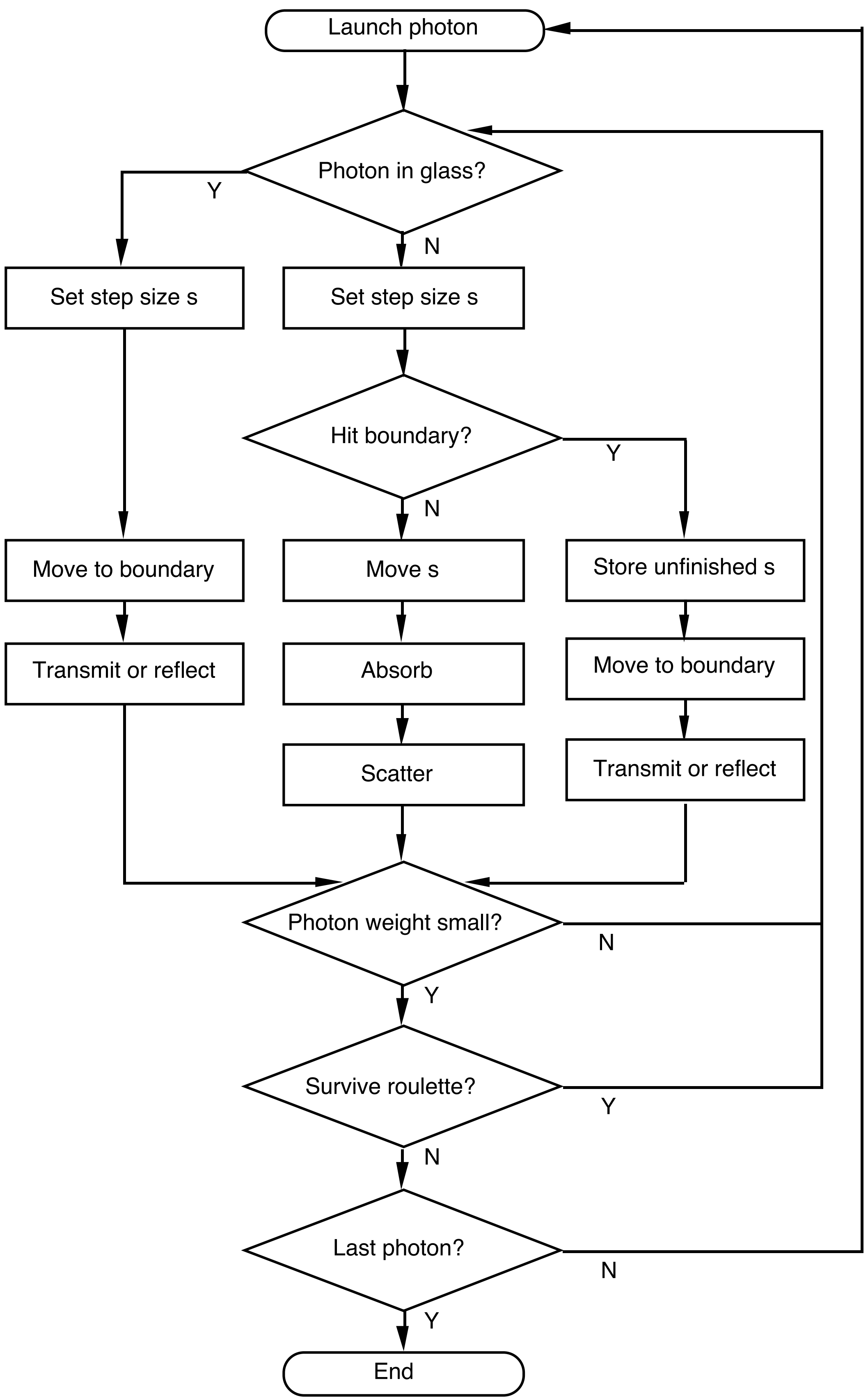
6: Photon termination

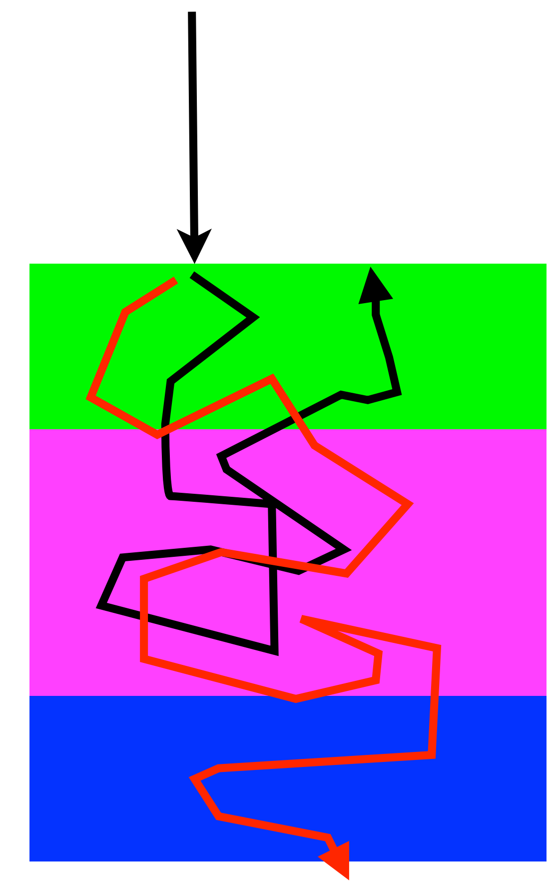
Roulette algorithm: we give the photon a chance to survive and to gain a higher weight.



Total energy is conserved!!

Workflow





Wang. Jacques, "MC Modeling of Light Transport in Multi-Layered Tissues in Standard C"

How does it work?

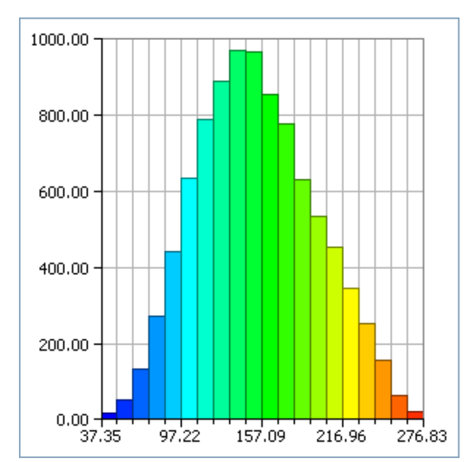
Stochastic formulation of the problem

↓

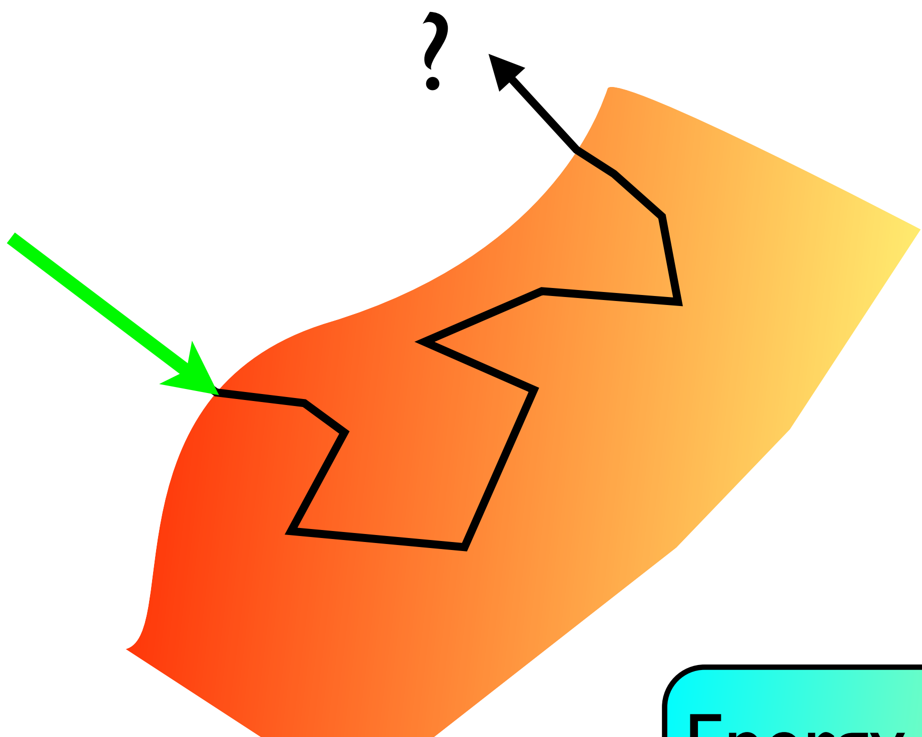
Random number generation to simulate the process

↓

Statistics: data binning, momenta, confidence intervals...



How to score physical quantities?



The calculus depends on what we want to know about the photon distribution.

Transmittance

Radiance

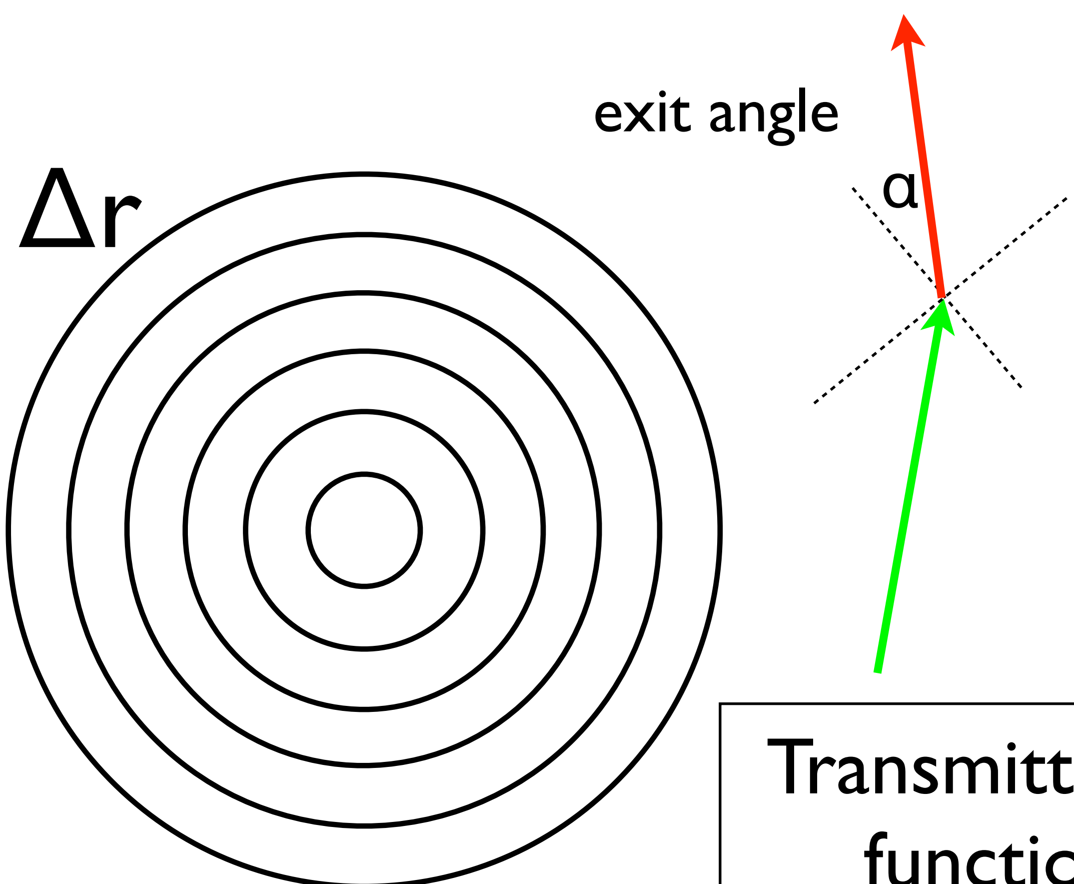
Energy deposition

Flux

Time-resolved

Angularly-resolved

Example: transmittance from a slab



Photon is transmitted inside the anulus r_i with an exit angle α_i

$$T(\alpha_i, r_i) \leftarrow T(\alpha_i, r_i) + W$$

Transmittance as a function of r

$$T(r_i) = \sum_{\alpha_i} T(\alpha_i, r_i)$$

Fluence rate

$$\Phi(r_i) = \frac{T(r_i)}{N 2\pi r \Delta r}$$

What about a time-resolved approach?

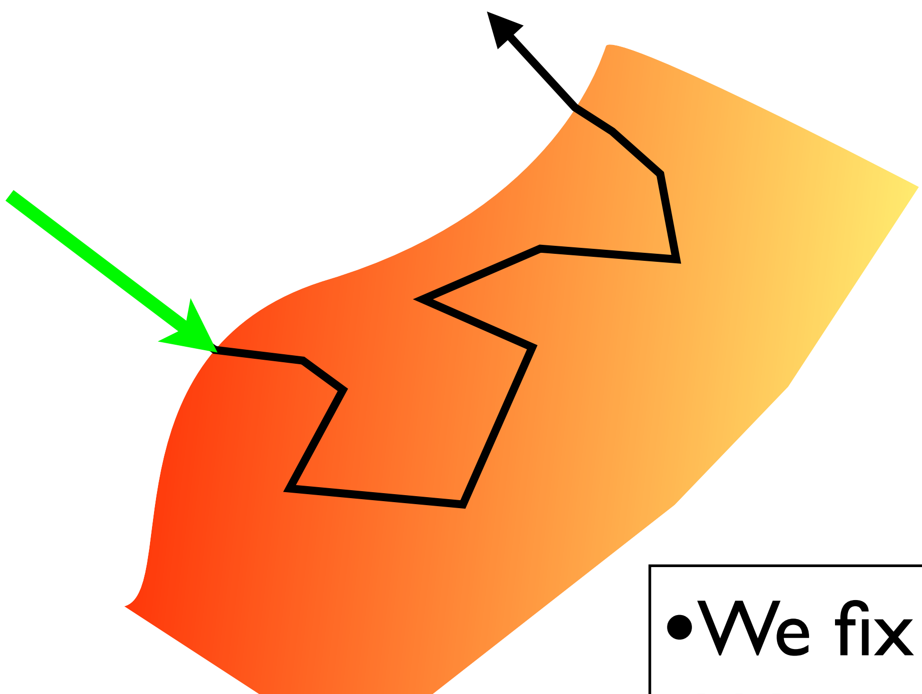
$\boxed{\text{Property of the RTE equation}}$

$$\boxed{I(t, \mu_s, \mu_a) = I_0(t, \mu_s) \cdot \exp(-\mu_a v t)}$$

$$\boxed{\mu_a = 0}$$

We generate a MC simulation in a non-absorbing medium and after we apply the absorption.

Time-resolved MC



$$s = -\frac{\log(\xi)}{\mu_s}$$

Sampling rule
for the step size

- We fix the detectors
- We store the total path of each photon hitting the detectors

"Microscopic Beer-Lambert law" method

TR MC: post-processing

Data.dat

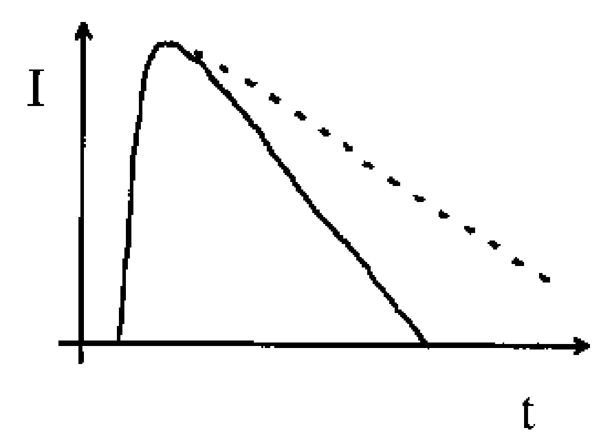
l (cm)
2.1
3.5
10.2
2.4
...
10.2

~ Mbyte

temporal windows

Δt

μ_a



TR MC: post-processing

$$\boxed{t_i - \Delta t/2 < t < t_i + \Delta t/2}$$

$$\boxed{S(t_i, \mu_a) = \frac{N_i}{N_{tot} \Delta t A_{det}} \exp(-\mu_a v t_i)}$$

N_{tot}	Total photon launched.
N_i	Total photon detected in the i^{th} time window.
Δt	Temporal window.
A_{det}	Detector area.

TR MC: analysis

$$\boxed{S(t_i, \mu_a) = \frac{N_i}{N_{tot} \Delta t A_{det}} \exp(-\mu_a v t_i)}$$

What's the distribution of this stochastic variable?

It's a Binomial distribution!!

$$\boxed{\sigma(t_i, \mu_a) \simeq \frac{S(t_i, \mu_a)}{\sqrt{N_i}}}$$

Other MC schemes

$$\xi \sim \mathscr{U}(0,1) \xrightarrow[s = -\frac{\log(\xi)}{\mu_t}]{} \text{step-length } s$$

$$\omega \sim \mathscr{U}(0,1) \begin{cases} \omega \leq \dfrac{\mu_s}{\mu_t} \longrightarrow \text{propagate photon} \\[2ex] \omega > \dfrac{\mu_s}{\mu_t} \longrightarrow \text{stop propagating} \end{cases}$$

*I.A. Sassaroli and F. Martelli, "Equivalence of four Monte Carlo methods for photon migration in turbid media.," J. Opt. Soc. Am. A.. **29**, 2110–7 (2012).*

"Albedo-rejection" method

Other MC schemes

$$\xi_1 \sim \mathscr{U}(0,1) \xrightarrow[s_s = -\frac{\log(\xi_1)}{\mu_s}]{} \text{scattering step-length}$$

$$\xi_2 \sim \mathscr{U}(0,1) \xrightarrow[s_a = -\frac{\log(\xi_2)}{\mu_a}]{} \text{absorption step-length}$$

$$s_s \leq s_a \longrightarrow \text{propagate photon}$$

$$s_s > s_a \longrightarrow \text{stop propagating}$$

I. A. Sassaroli and F. Martelli, "Equivalence of four Monte Carlo methods for photon migration in turbid media.," J. Opt. Soc. Am. A. ***29****, 2110–7 (2012).*

"Absorption-Scattering path length rejection" method

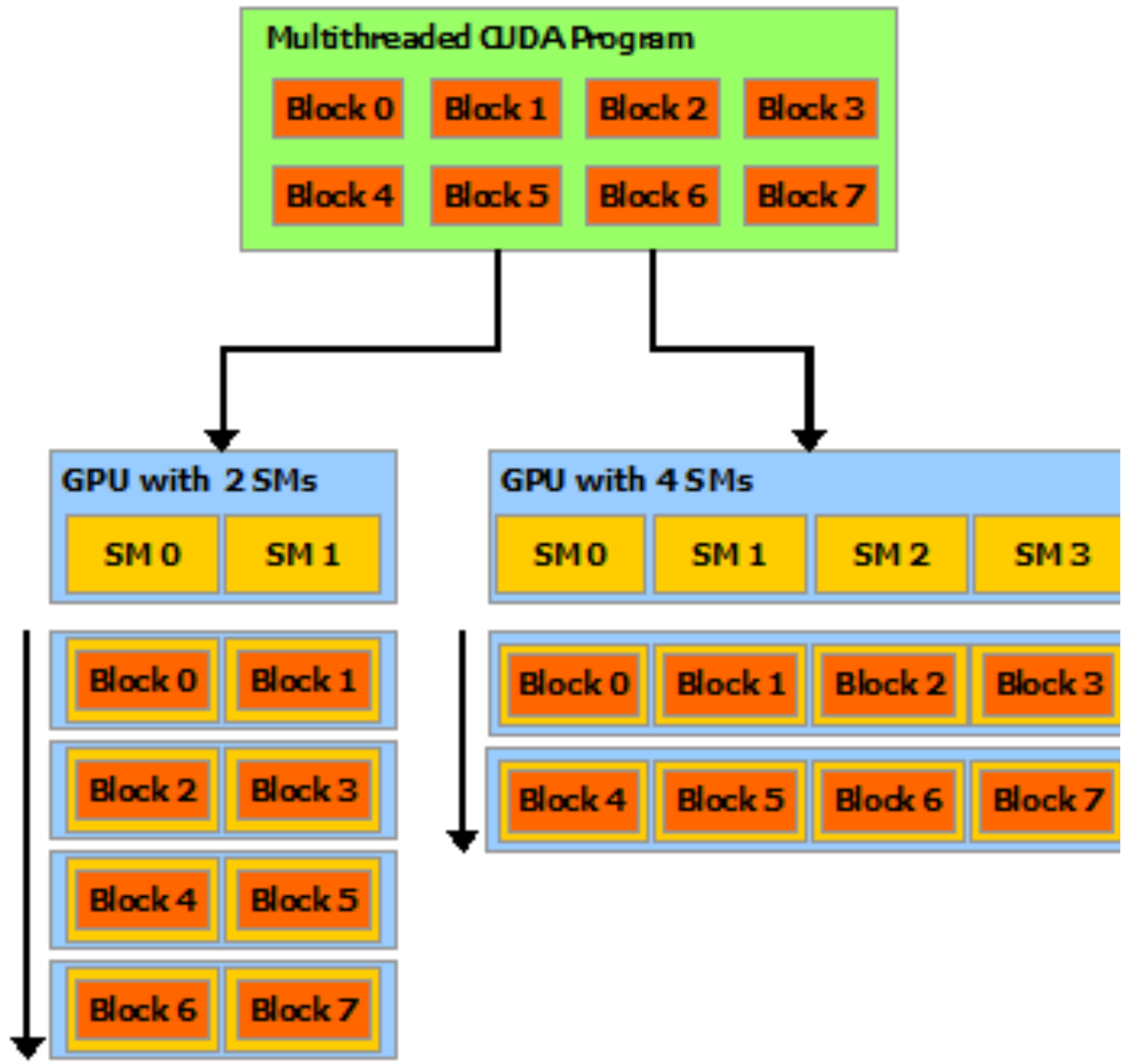
Considerations

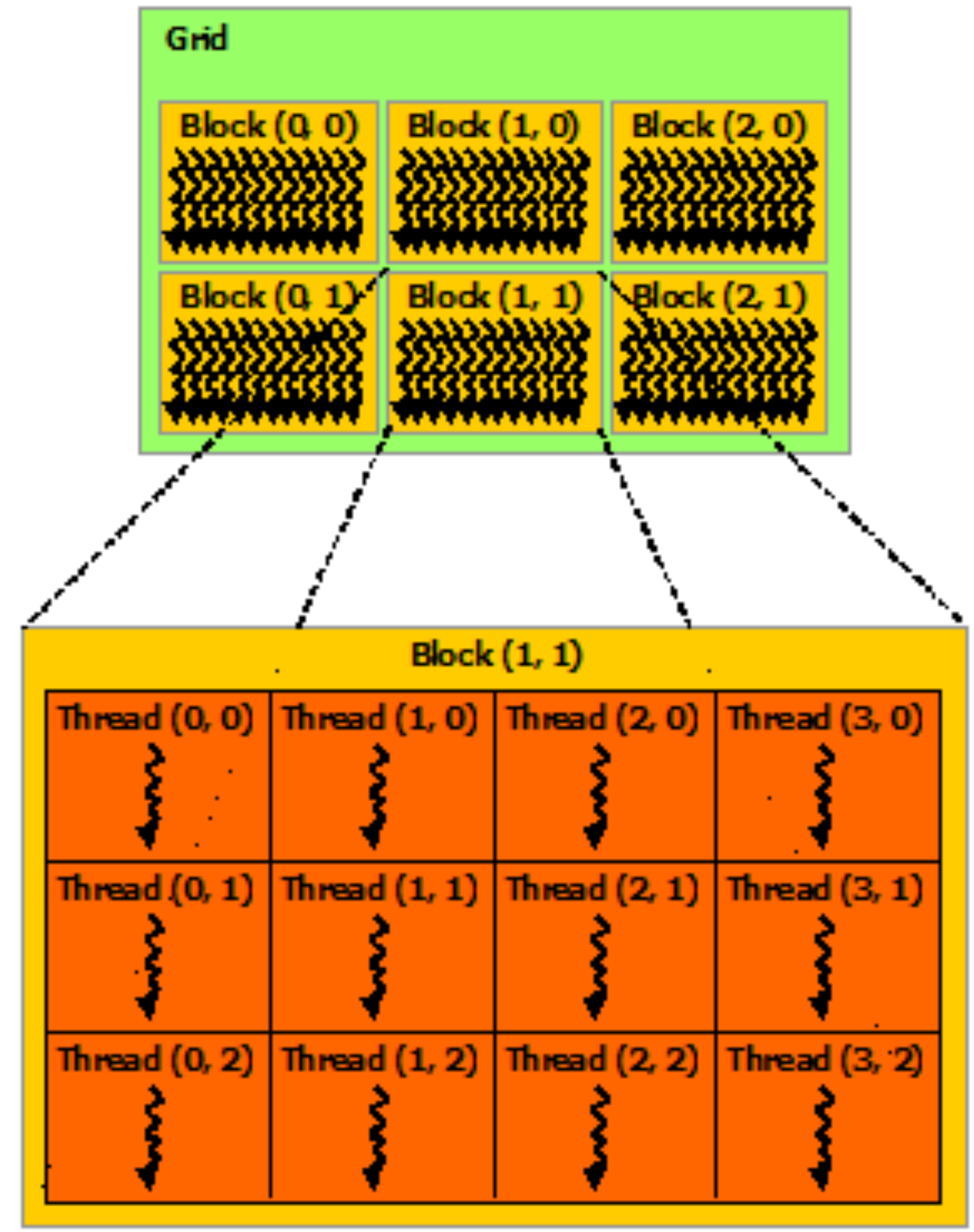
- The main bottle-neck of MC simulations in turbid media is the intersection check with boundaries and regions.
- Someone proposed intersection algorithms taken from the rendering.
- MC can be speeded up by parallel-computing (GPU, cluster, ..)

Parallel-Computing approach

- MC methods, based on the stochastic independence of events, are well suitable for parallel computing (GPU, cluster,..)
- ATTENTION: you should generate good RND numbers sequences for each CPU, but these sequences should be uncorrelated and must not overlap!!

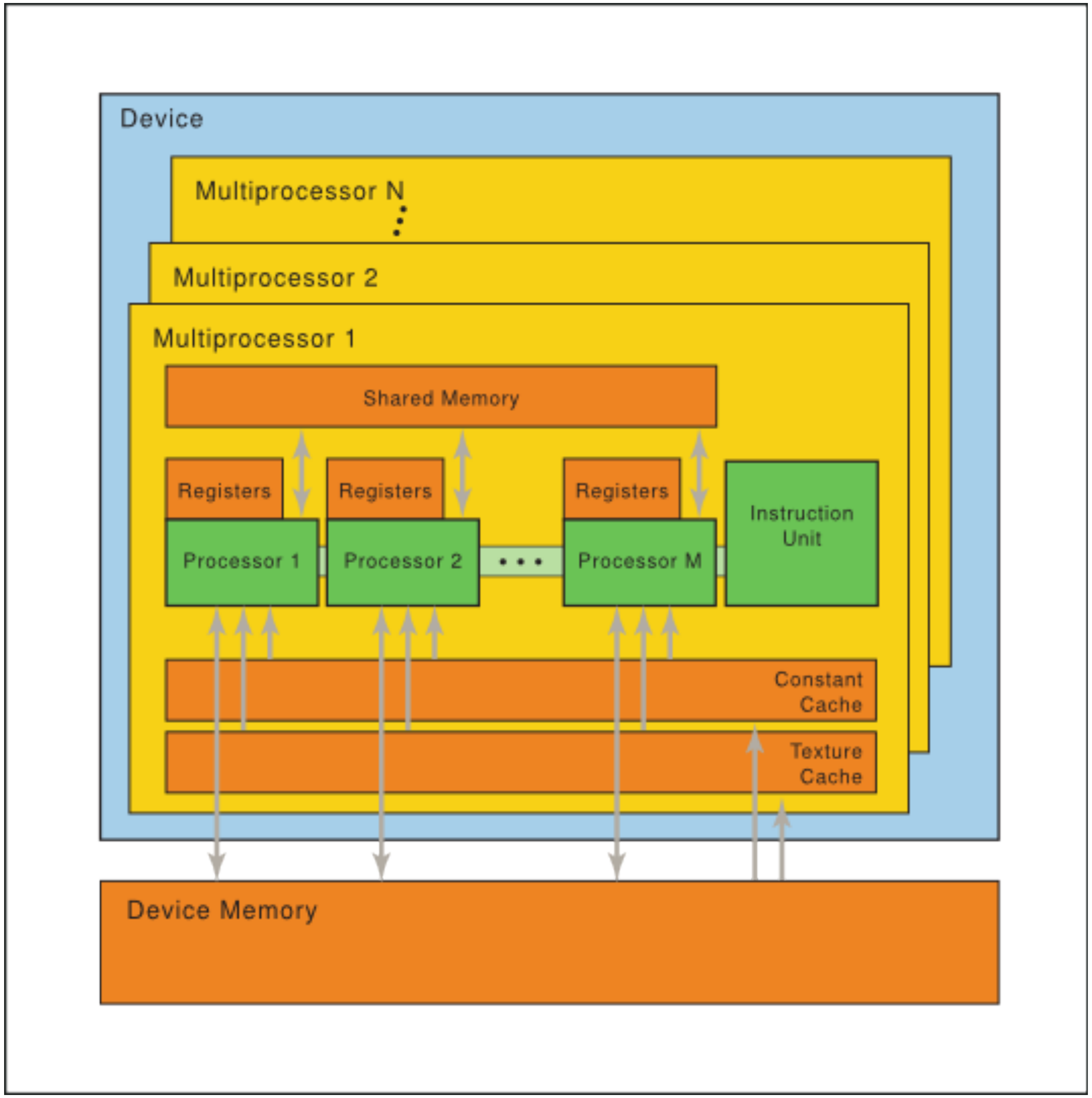
CUDA® architecture





CUDA C Programming guide, NVIDIA

CUDA® architecture



CUDA C Programming guide, NVIDIA

CUDA-based MC for photon transport in biological tissues

1. Q. Fang and D. A. Boas, "Monte Carlo simulation of photon migration in 3D turbid media accelerated by graphics processing units.," ***Opt. Express*** 17, 20178–90 (2009).

1. E. Alerstam, W. Chun, Y. Lo, T. D. Han, J. Rose, S. Andersson-Engels, and L. Lilge, "Next-generation acceleration and code optimization for light transport in turbid media using GPUs," ***Opt. Express 1***, 658–675 (2010).

1. A. Doronin and I. Meglinski, "Online object oriented Monte Carlo computational tool for the needs of biomedical optics," ***Biomed. Opt. Express 2***, 2461–2469 (2011).

Parallel implementation of PRNG

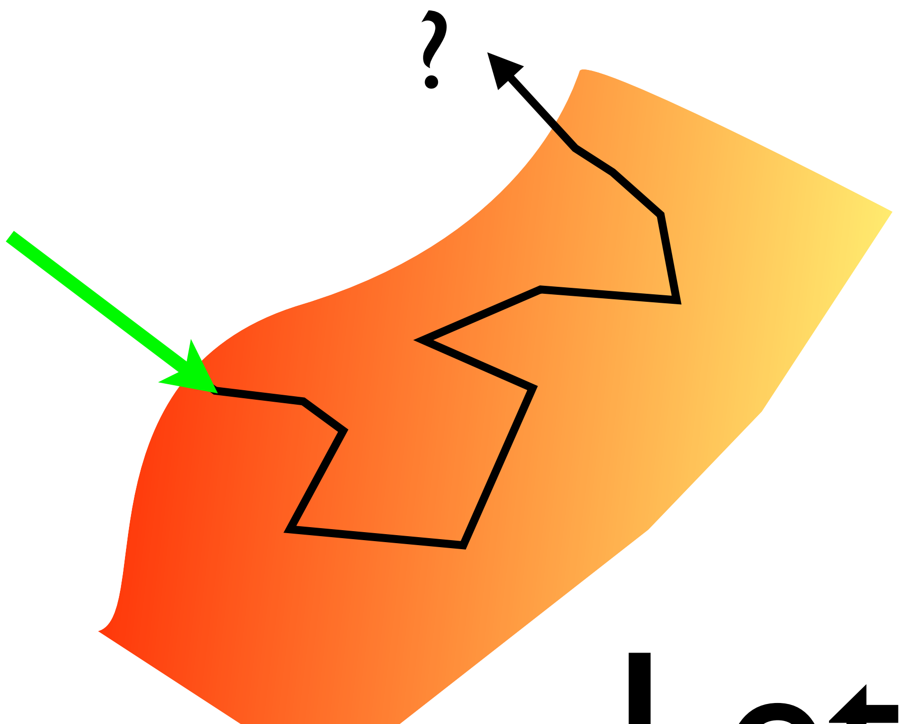
- The PRNG should satisfy the following before discussed properties:
  - Long enough <u>period.</u>
  - <u>Uniformness</u> between 0 and 1.
  - <u>Randomness.</u>
  - The sequences of each process should not overlap each other and should also be uncorrelated each other.!!

Parallel Multiply-With-Carry (MWC)

- Low amount of memory required.
- Every process is started with a <u>different seed</u> and a <u>different multiplier</u>.

This ensure uncorrelated and non-overlapping sequences.

Alerstam et al, ***Opt. Express 1***, 658–675 (2010)



Let's see some simulations now!!!