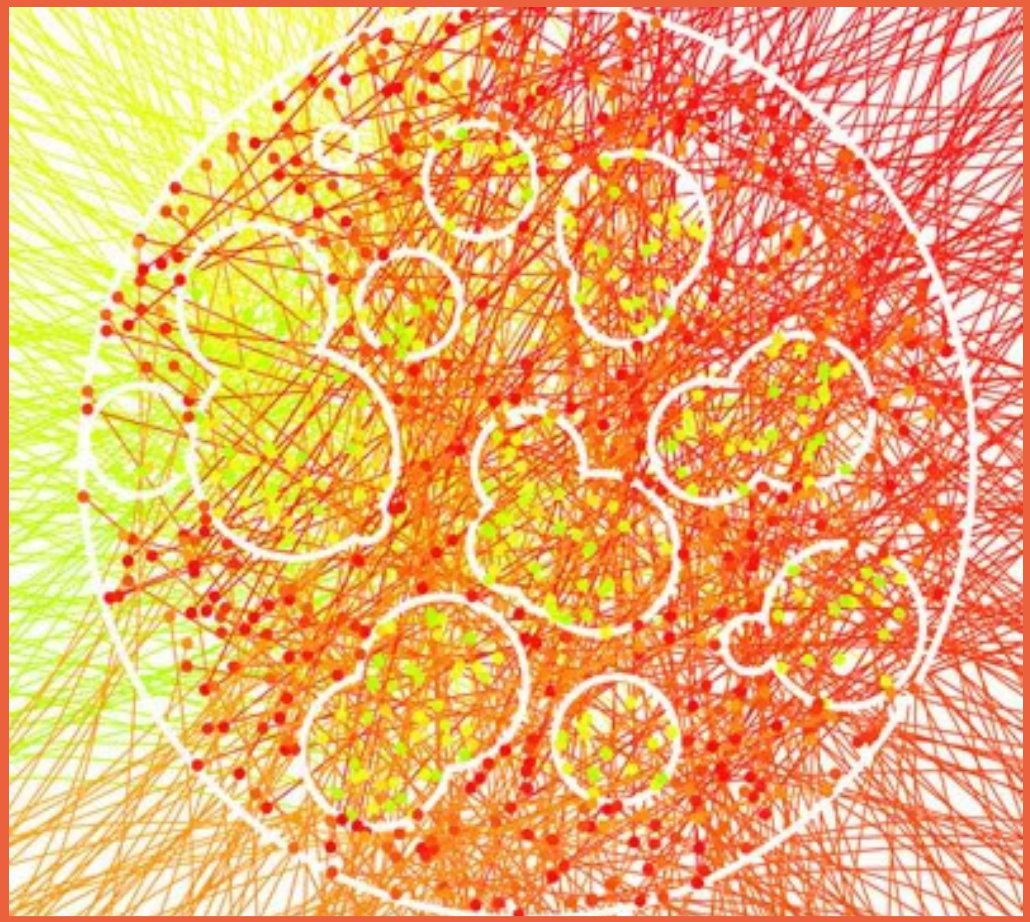
MONTE CARLO METHODS



D. Guirado
Instituto de Astrofísica de Andalucía – CSIC, Granda, Spain
Polarization school, Aussois, 06/06/2013

STOCHASTIC ALGORITHMS

PROBLEM = GET OUTPUT FROM INPUT THROUGH AN ALGORITHM

Deterministic algorithm:

$Input_1$ ➡ $Output_1$

Stochastic algorithm:

$Input_1$ + random decision → $Output_1$, $Output_2$, $Output_3$, ...

MONTE CARLO METHODS

Definition:

Iterative method including random decisions that gives the correct solution of a problem with a certain probability <1.

Named after the district of Monte Carlo (Monaco), European gambling capital.

Monte Carlo

Monte Carlo casino

Origin:

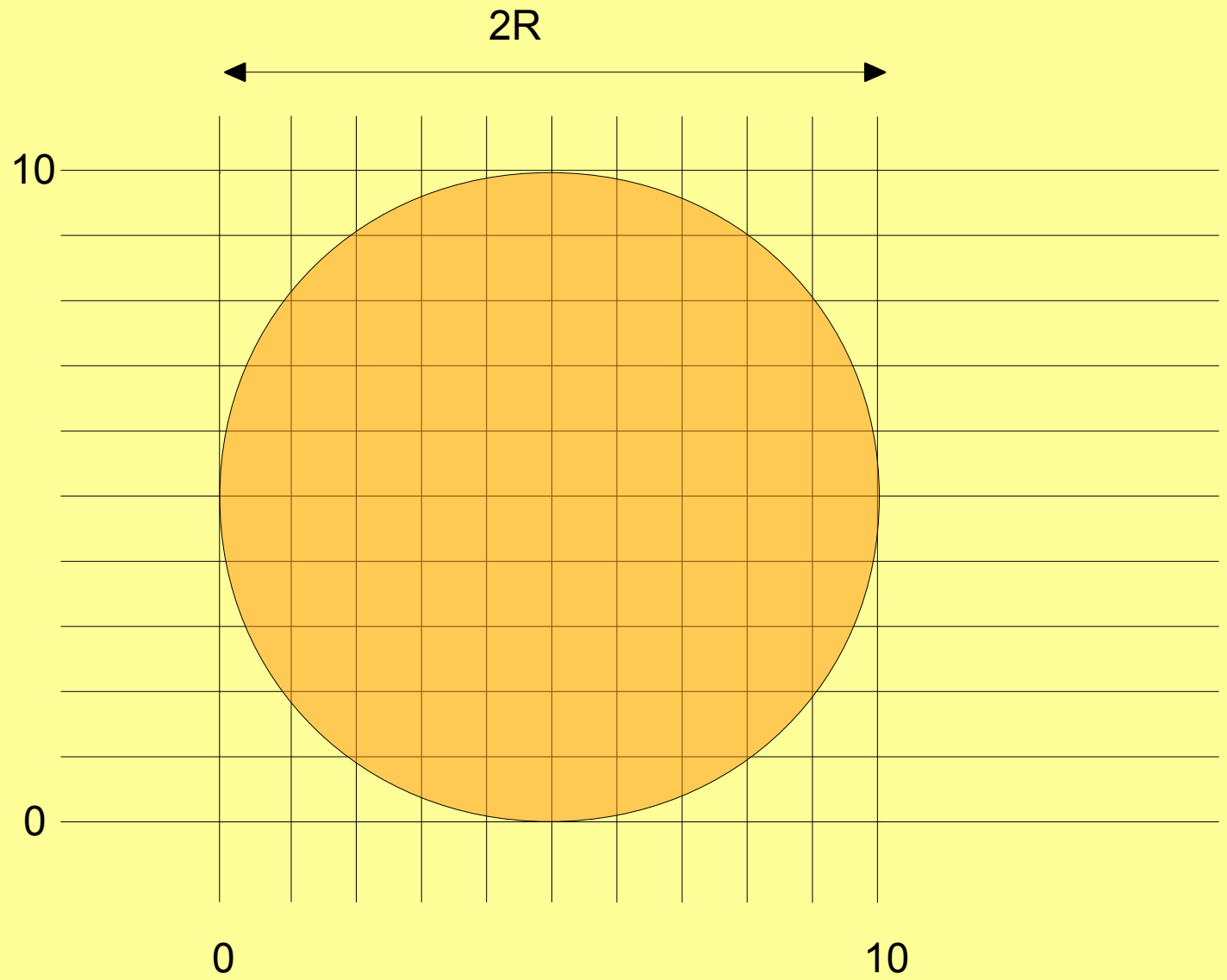
Study of the diffusion of neutrons in a fusion experiment (Los Álamos Laboratory).

TYPICAL MONTE CARLO METHOD

Input parameters + ξ_i → x_i

$$X = \frac{1}{N}\sum_{i=1}^{N} x_i$$

PRACTICE #1: VALUE OF π



$$\left\{\begin{array}{c} Area = \pi R^2 \\ Area = 4R^2 \dfrac{N(darts\ inside)}{N(darts\ thrown)} \end{array}\right\} \Rightarrow \pi = 4\frac{N(darts\ inside)}{N(darts\ thrown)} = 4\frac{1}{N}\sum_{i=1}^{N} x_i$$

$$N = N(darts\ thrown)$$

$$x_i = \left\{\begin{array}{c} 1\ if\ dart\ goes\ inside \\ 0\ if\ dart\ goes\ out \end{array}\right\}$$

STOCHASTIC METHODS WITH COMPUTERS? HOW?

Stochastic method = generating random number + performing operation

$$N \to \infty \Rightarrow t \to \infty$$

Computers make operations much faster.
Computer are deterministic, no random numbers.
Solution: *pseudo-random numbers*.

PSEUDO-RANDOM NUMBERS

Finite (but large) lists of numbers generated by an algorithm.
No correlations beteween the numbers.

Examples:

- *Numerical Recipes*.
- *MKL* (optimized for *iFort*)
- http://www.psychicscience.org/random.aspx

ran2 (Numerical Recipes)

- For any *Fortran* or *C* compiler.
- The authors offer $1000 to the first person who finds a statistical test that proves that *ran2* fails.

CENTRAL LIMIT THEOREM

$\rho(x) =$ Any probability density function with average μ and variance σ^2.
$x_1, x_2, \ldots, x_N =$ collection of N values sampled through ρ.

Then, the probability density distribution $\rho_N(X)$, which elements are

$X_N = \frac{1}{N}\sum_{i=1}^{N} x_i$, has a average of μ and a variance of $\frac{\sigma^2}{N}$, where

$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)^2$.

LAW OF LARGE NUMBERS

$N \to \infty \Rightarrow \rho_N \to$ gaussian.

HOW TO CALCULATE THE ERROR IN A MONTE CARLO METHOD

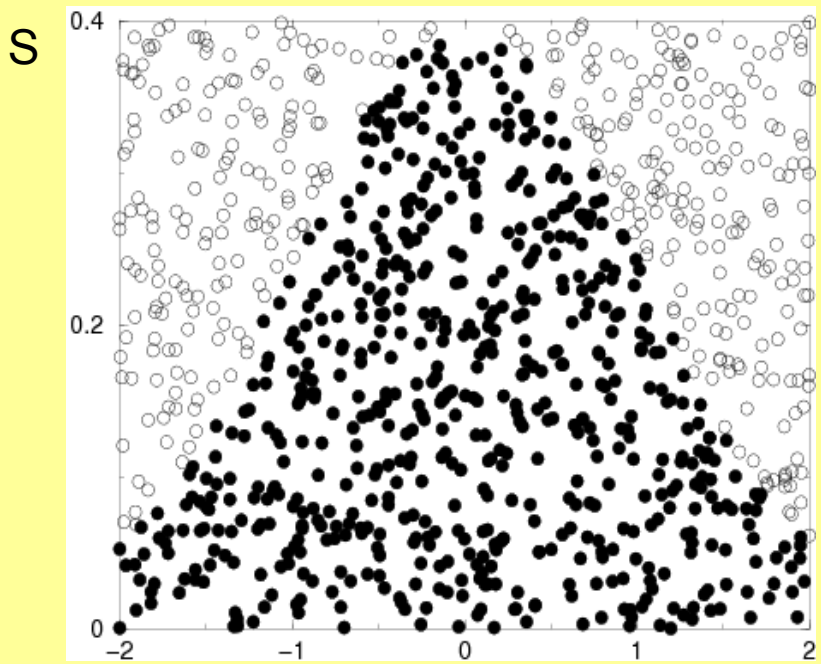
$$(N \to \infty)$$

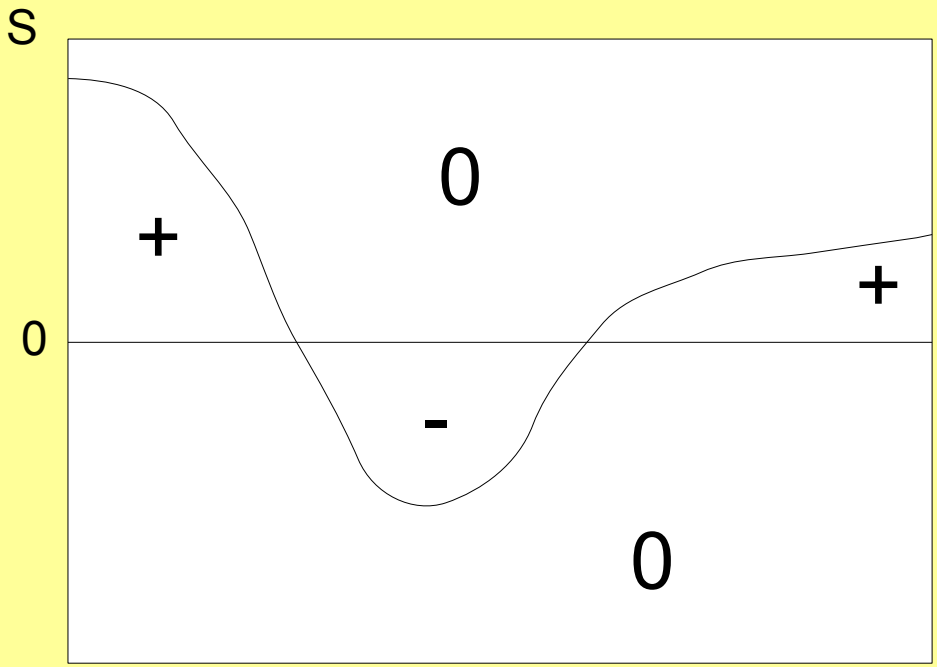
- Fit all calculated x_i to a gaussian distribution and obtain σ. Let us call the correct solution of the problem μ. Then:

- $X \in \left[\mu - \frac{\sigma}{\sqrt{N}}, \mu + \frac{\sigma}{\sqrt{N}} \right]$ with $p = 0.682.$
- $X \in \left[\mu - 2\frac{\sigma}{\sqrt{N}}, \mu + 2\frac{\sigma}{\sqrt{N}} \right]$ with $p = 0.954.$
- $X \in \left[\mu - 3\frac{\sigma}{\sqrt{N}}, \mu + 3\frac{\sigma}{\sqrt{N}} \right]$ with $p = 0.998.$
- If the accuracy is not enough, increase N (the error decreases as $\sim \frac{1}{\sqrt{N}}$).

RESULTS FROM MONTE CARLO SIMULATIONS HAVE ERRORS!

PRACTICE #2: INTEGRATION





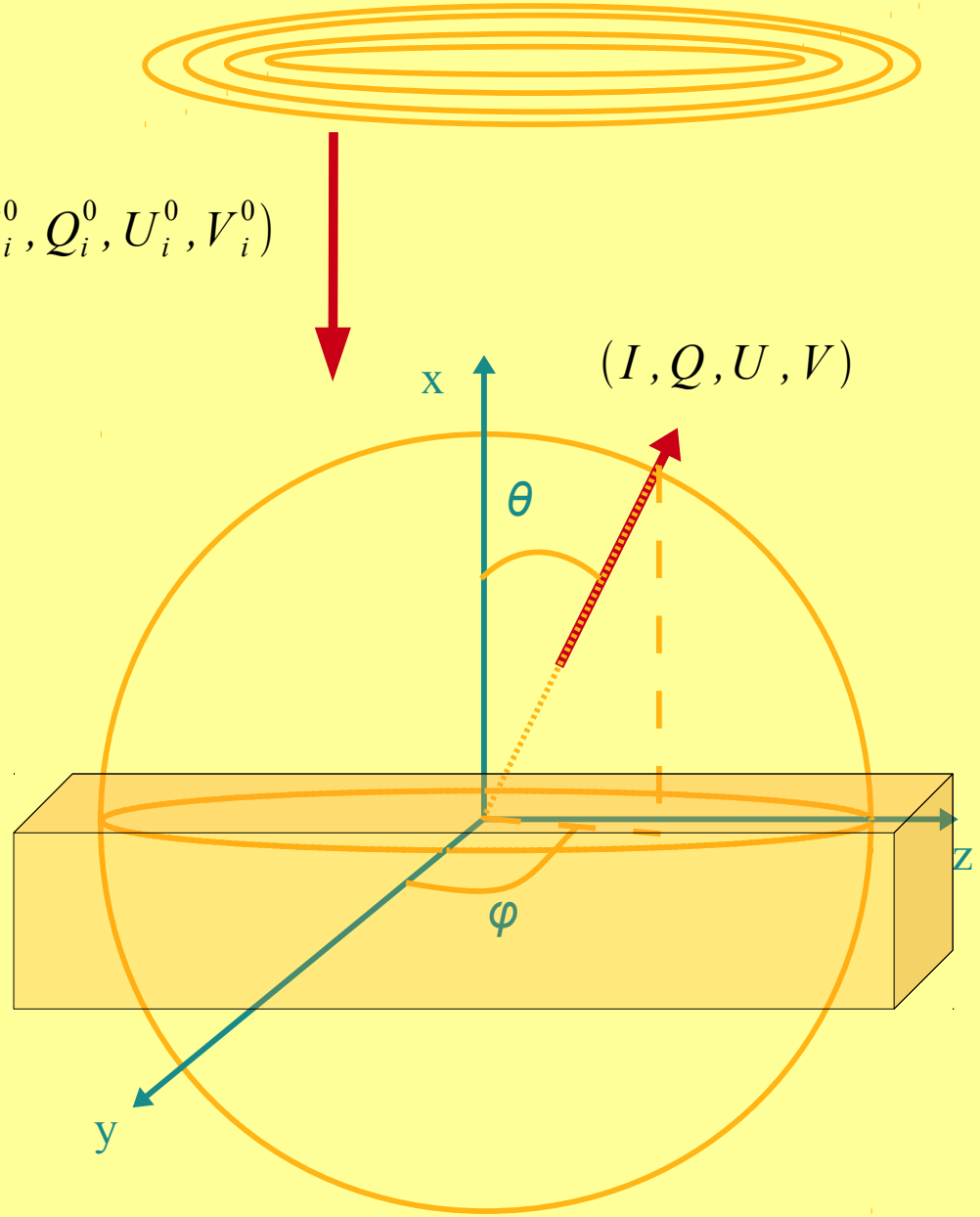
$$Integral = S\frac{N(darts\ inside\ positive) - N(darts\ inside\ negative)}{N(darts\ thrown)} = S\frac{1}{N}\sum_{i=1}^{N} x_i$$

$$N = N(darts\ thrown)$$

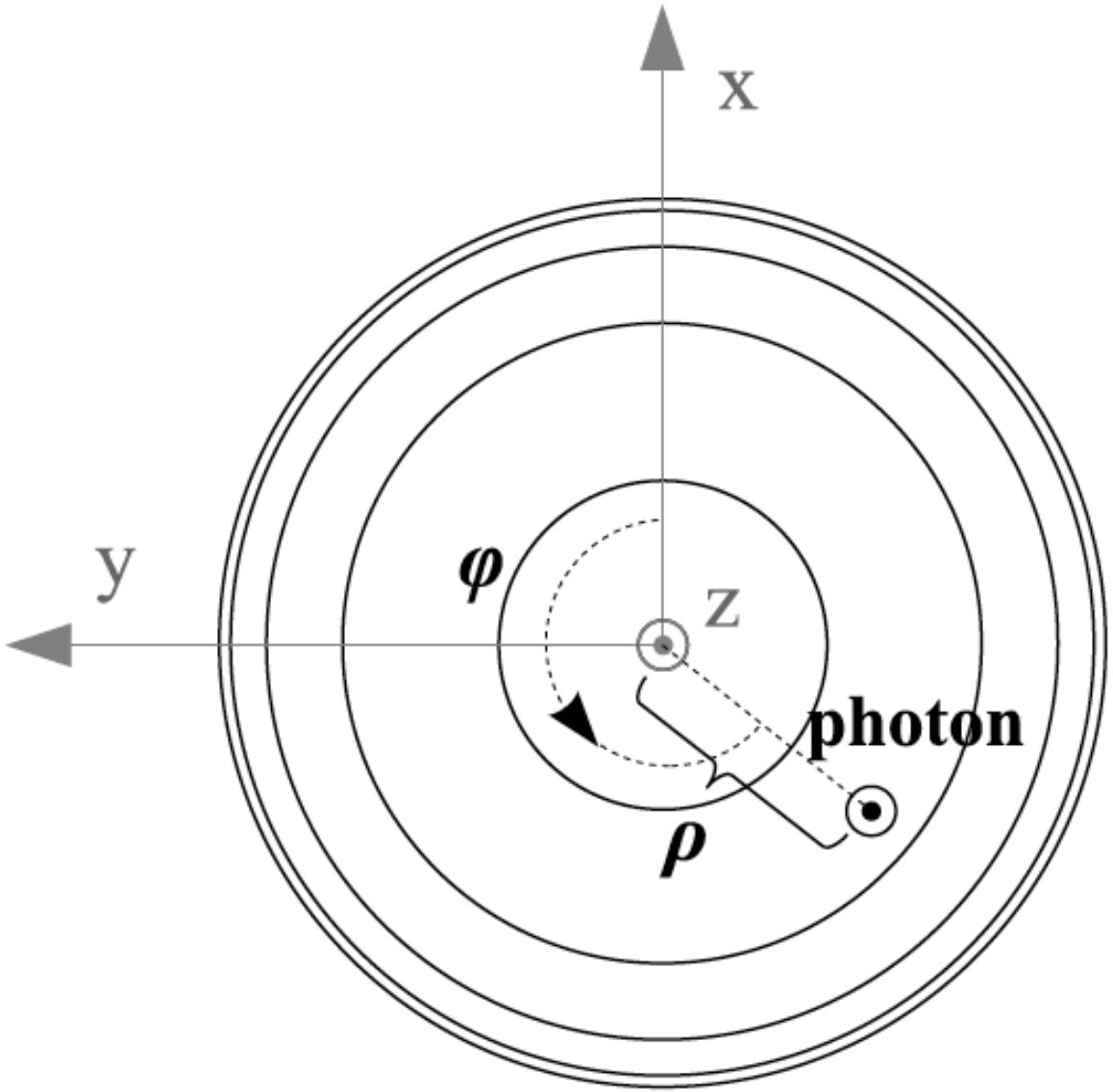
$$x_i = \begin{Bmatrix} 0\ if\ dart\ goes\ outside \\ 1\ if\ dart\ goes\ inside\ positive \\ -1\ if\ dart\ goes\ inside\ negative \end{Bmatrix}$$

PRACTICE #3: LINEAR POLARIZATION IN SCATTERING

$$(I, Q, U, V) = \frac{1}{N} \sum_{i=1}^{N} (I_i^0, Q_i^0, U_i^0, V_i^0)$$

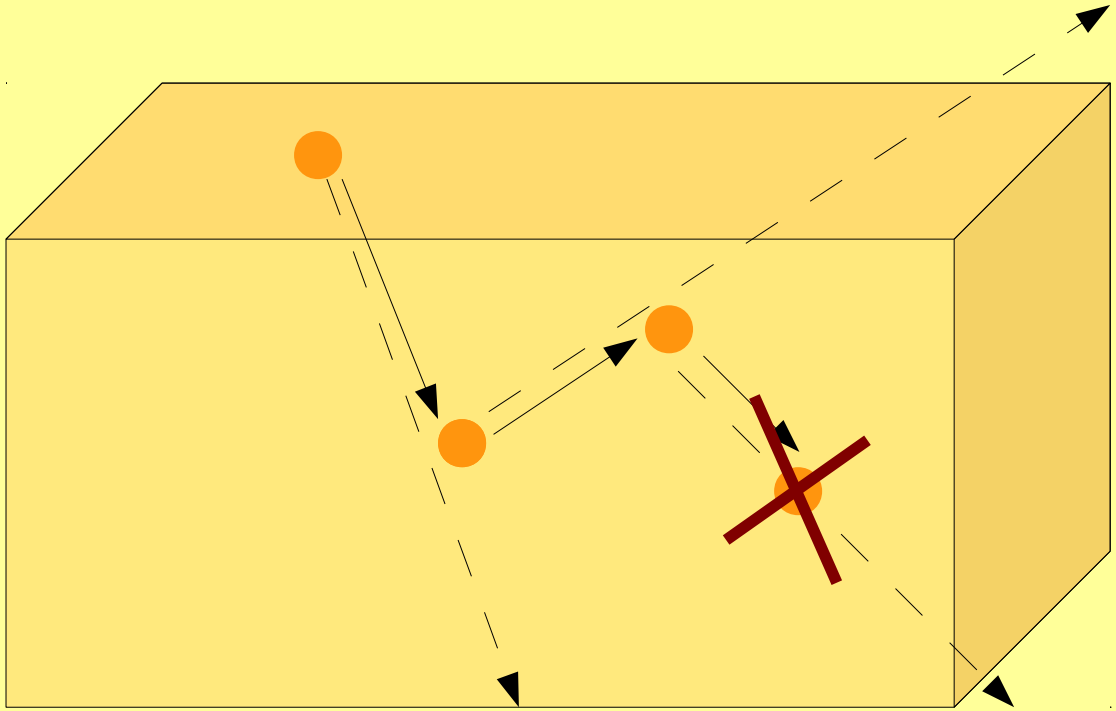


PRACTICE #3: LINEAR POLARIZATION IN SCATTERING



$$\varphi \in [0, 2\pi]$$
$$\rho = \rho_1 + \sqrt{\xi(\rho_2^2 - \rho_1^2)}$$

PRACTICE #3: LINEAR POLARIZATION IN SCATTERING



Particles:

- Any size distribution.
- Any refractive index (even optically activity).
- Any geometry.
- Any orientation (aligned or not).

Any optical thickness of the coma (single or multiple scattering).

Inhomogeneous coma.

Packet of photons (W)
- Improves statistics.
- Reduces computation time.

W=1 at the beginning
Ends when W<Wmin

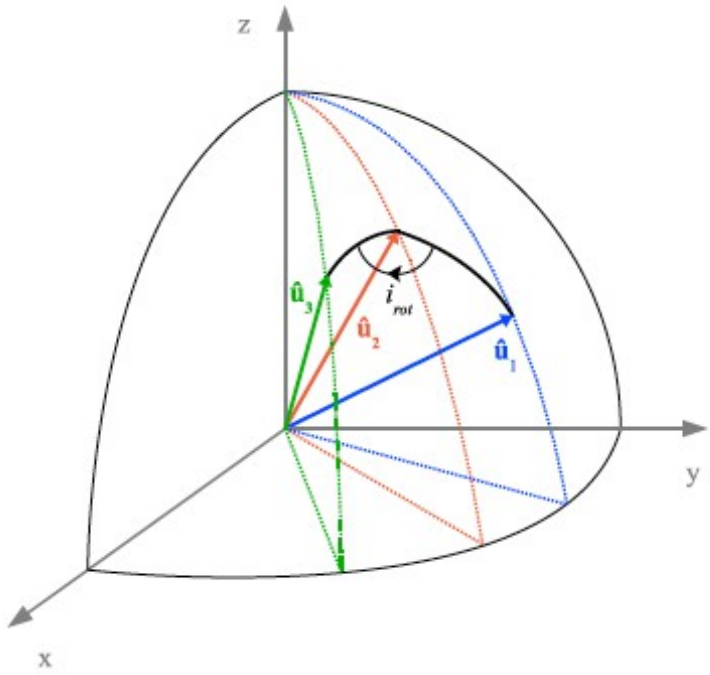
PRACTICE #3: LINEAR POLARIZATION IN SCATTERING

- φ uniformly randomly obtained.
- Choosing the scattering direction:
  $p(\theta) \propto [F_{11}I_0 + F_{12}Q_0 + F_{13}U_0 + F_{14}V_0] \sin\theta.$
- Change of the scattering plane.

$$\begin{pmatrix} I \\ Q \\ U \\ V \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos 2i_{rot} & \sin 2i_{rot} & 0 \\ 0 & -\sin 2i_{rot} & \cos 2i_{rot} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} I_0 \\ Q_0 \\ U_0 \\ V_0 \end{pmatrix}$$

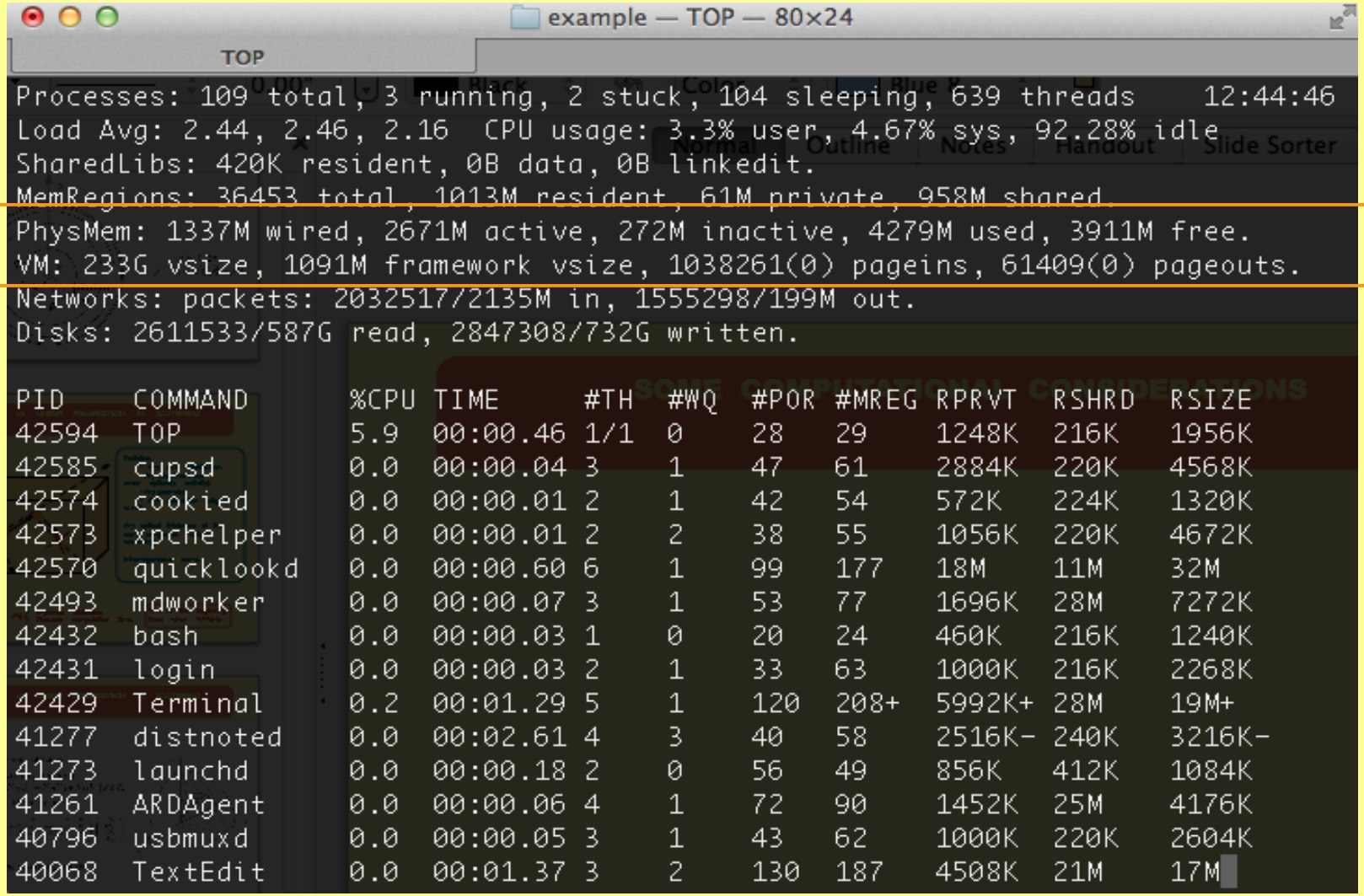
.

- Multiplying by the scattering matrix.



SOME COMPUTATIONAL CONSIDERATIONS

$~top



PARALLELIZE YOUR CODE

Classical method: MPI. New easier alternative: ***OpenMP*** (Fortran and C).

How it works:

```
      PROGRAM OMP_SUM2
      INTEGER NMAX
      PARAMETER(NMAX=20000)
      INTEGER I
      REAL A(NMAX),C(NMAX)
      DO I = 1,NMAX
        A(I) = I * 1.0
        C(I)=1.
      ENDDO
C$OMP PARALLEL shared(A,B,C,NMAX) private(I,J)
C$OMP DO
      DO I = 1,NMAX
         DO J=1,I
            C(I)= C(I)+A(J)
         END DO
      ENDDO
C$OMP END DO
C$OMP END PARALLEL
      WRITE(*,*)C(NMAX)
      END
```

Compile: **$~gfortran omp_sum2.f -o omp_sum2**

Sequential omp_sum2

Compile: **$~gfortran -fopenmp omp_sum2.f -o omp_sum2**

Parallel omp_sum2

Execute: **$~time ./omp_sum2**

Download

http://www.iaa.es/~dani/swap/monte_carlo.zip