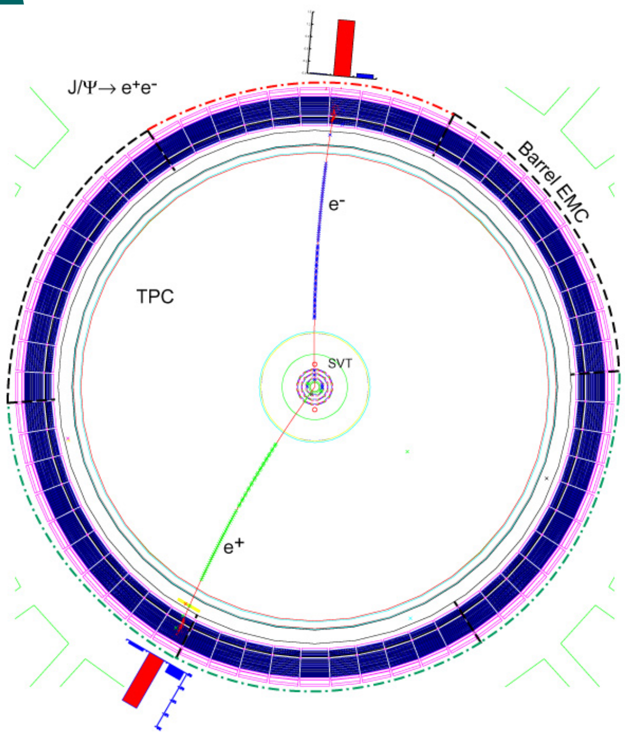
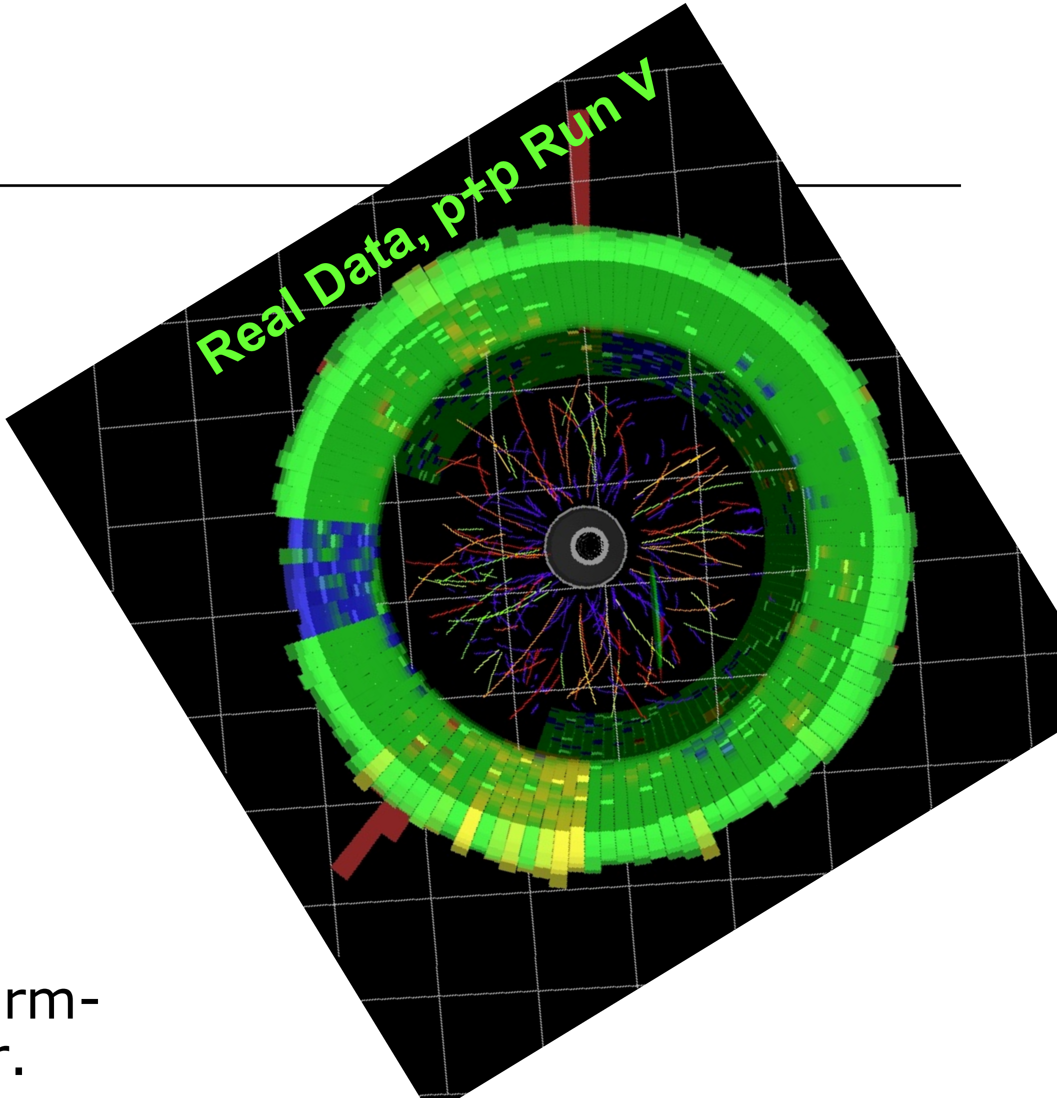
Monte Carlo





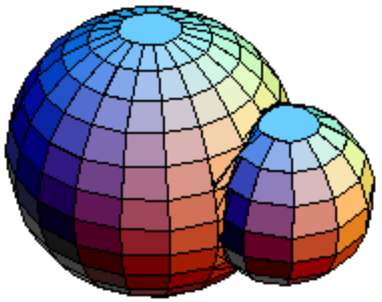
Monte Carlo simulation of
a decay of a J/ψ meson (charm-anticharm) into an e+e- pair.

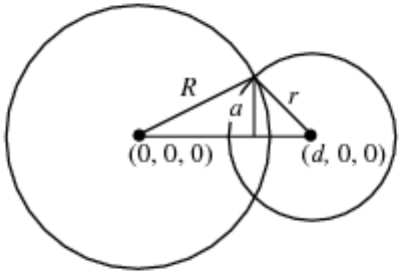
Monte Carlo methods

- Statistical Sampling
  - Name "Monte Carlo" was coined in the 1940's, random numbers → gambling (Stanislaw Ulam's uncle), probability distributions.
  - Very useful in evaluation of probability distribution functions.
  - Applications in high energy physics:
    - Simulation of high energy collision events.
    - Simulation of detector response.
  - Main idea:
    - analysis of a complicated physical system.
    - evolution of the physical system is governed by a process which is stochastic (probabilistic, non-deterministic)

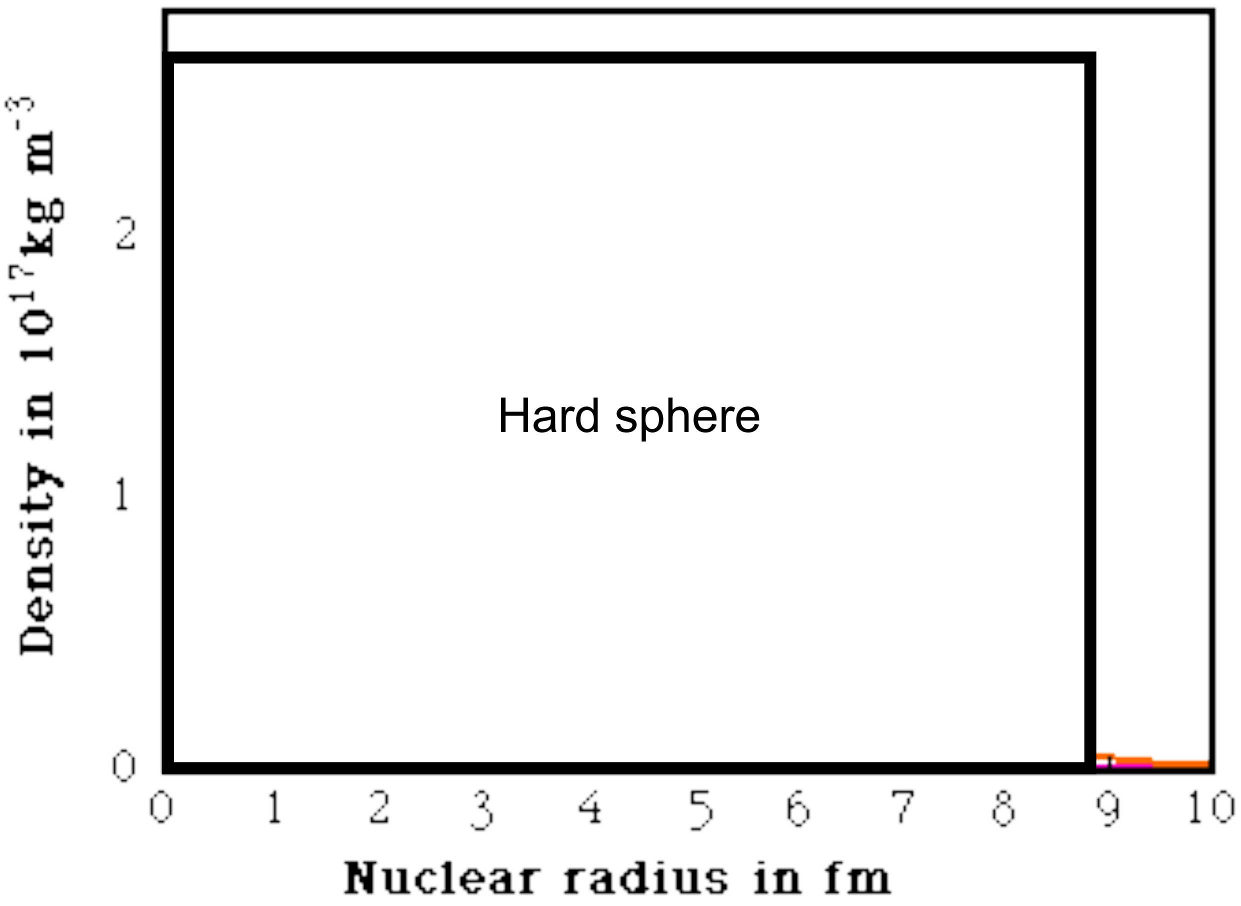
Monte Carlo Integration

- Commonly applied to multidimensional integration with complicated integrals and boundaries.
- Example: Overlap volume between two nuclei of radii R_1 and R_2, separated by a distance b.
  - Model them as spheres: analytic.
  - Realistic model: needs Monte-Carlo.





Realistic models:
Nuclear density profiles

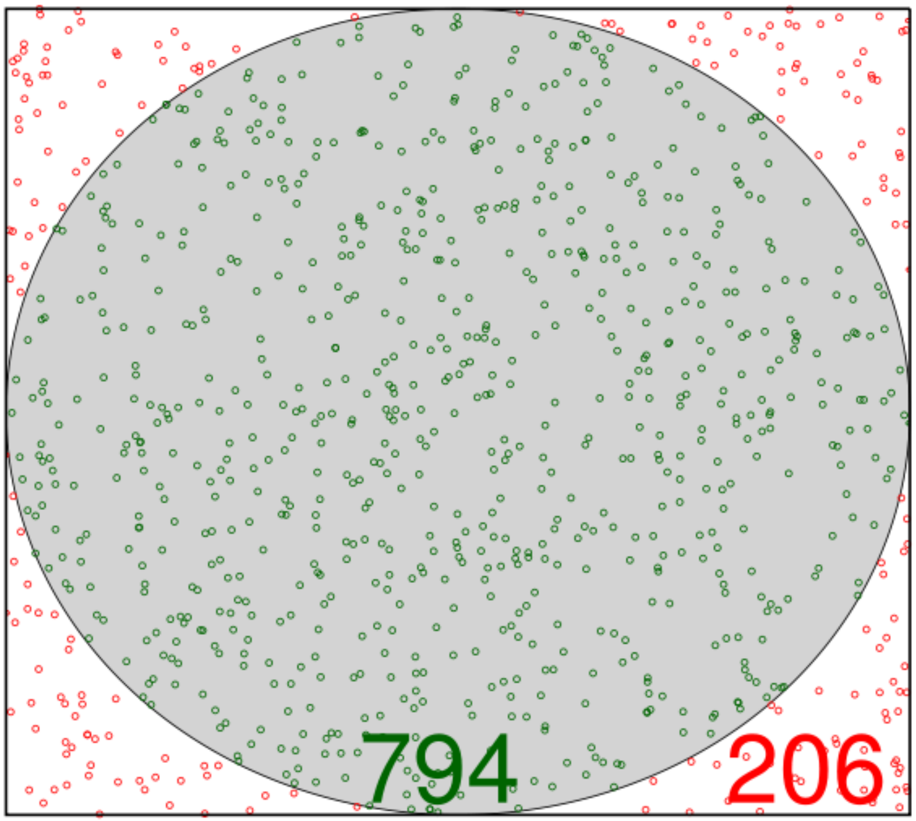


Monte Carlo integration: simple example

- Integrating a 1-D function between two limits supplied by the user.
- Algorithm:
  - generate random points (x,y) within a rectangular reference region.
  - Area of rectangle is known: A=bh
  - For each point, check to see if it falls below or above the 1-D function.
    - Make n_t total random points,
    - find n_b points with $0 < y < f(x)$
  - Estimate integral of function via:

$$\int f(x)dx \approx A\frac{n_b}{n_t}$$

Monte Carlo integration: Graphically.



- Throw n_t=1000 random points in the 2-D (x,y) space.
  - e.g. sphere of unit radius:
    - $-1 < x < 1$, b=2
    - $-1 < y < 1$, h=2
- Check if the point falls inside our limits
  - e.g. if $(x^2+y^2<1)$
  - found n_i=794
- Integral:
  - $bh*(n_i/n_t)=3.176$
  - Statistical uncertainty on n_i/n_t: $1/\sqrt{n_i}=0.03$.
  - Result = 3.18 ± 0.12

Assignment:

- Use the monte carlo integration program to find the area of a unit circle.
- You may use symmetry to restrict your calculation to positive x- and y-values.
- Use this to obtain a result for the value of π.
- Plot three histograms showing the result of 1000 pseudo experiments for calculating π,
  - one with 100 pairs,
  - one with 1000 pairs
  - one with 10000 pairs.
- For each of the histograms, show the mean and the standard deviation. Fit them with a Gaussian.
  - Understand the relation between the standard deviation obtained in each histogram and the statistical error quoted for the result of each pseudo-experiment.

Probability Distribution Functions: Stochastic Processes in Physics

- Stochastic variables:
  - Variables that fluctuate from one realization of a system to another.
    - Thermal effects.
    - Manufacturing uncertainties.
    - Quantum processes.
- Simple example: The 1-D Random Walk.

Random Walk and Stochastic Processes

- RW: The particle will take N steps. At each step, there is a 50/50 chance for the particle to move right or left.

Physical System is characterized by parameters that vary randomly.

- RW: two discrete values: ±1 distance units
  - + is right, - is left

- Want to calculate a global parameter that can be evaluated or measured.
  - RW: Total initial displacement from the origin after N steps.
- General problem: Predict the probability that the global variables possess a specific value when averaged over all trial experiments.
  - RW: Probability that the walk terminates at a given displacement from the origin.

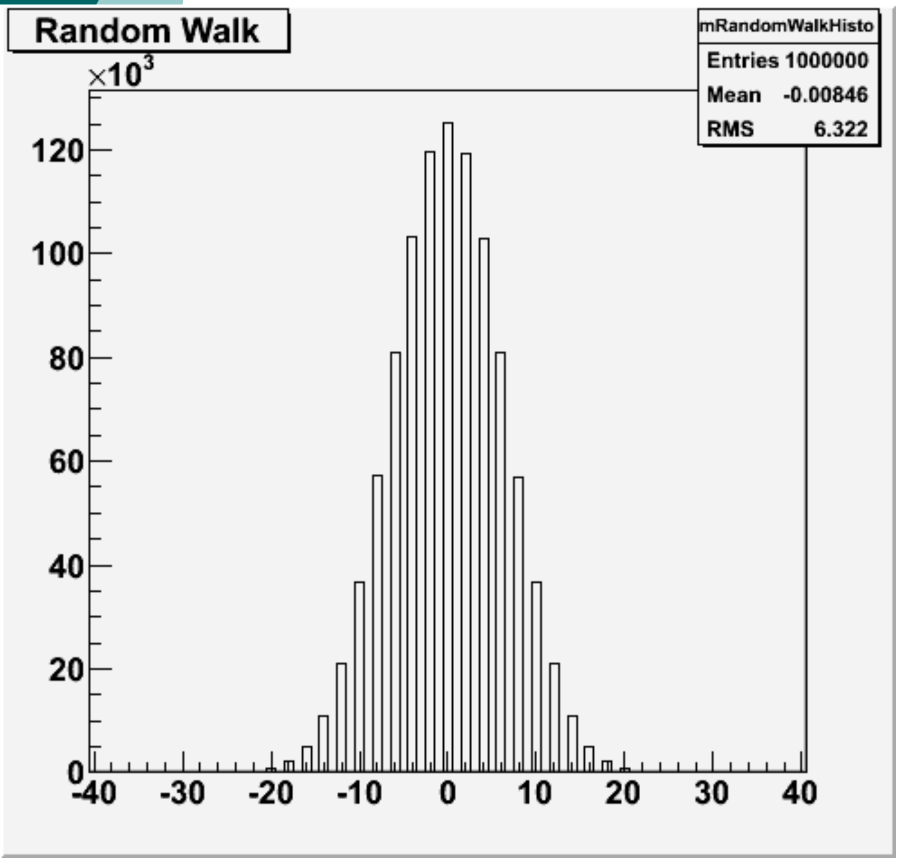
Making a random walk program in root:

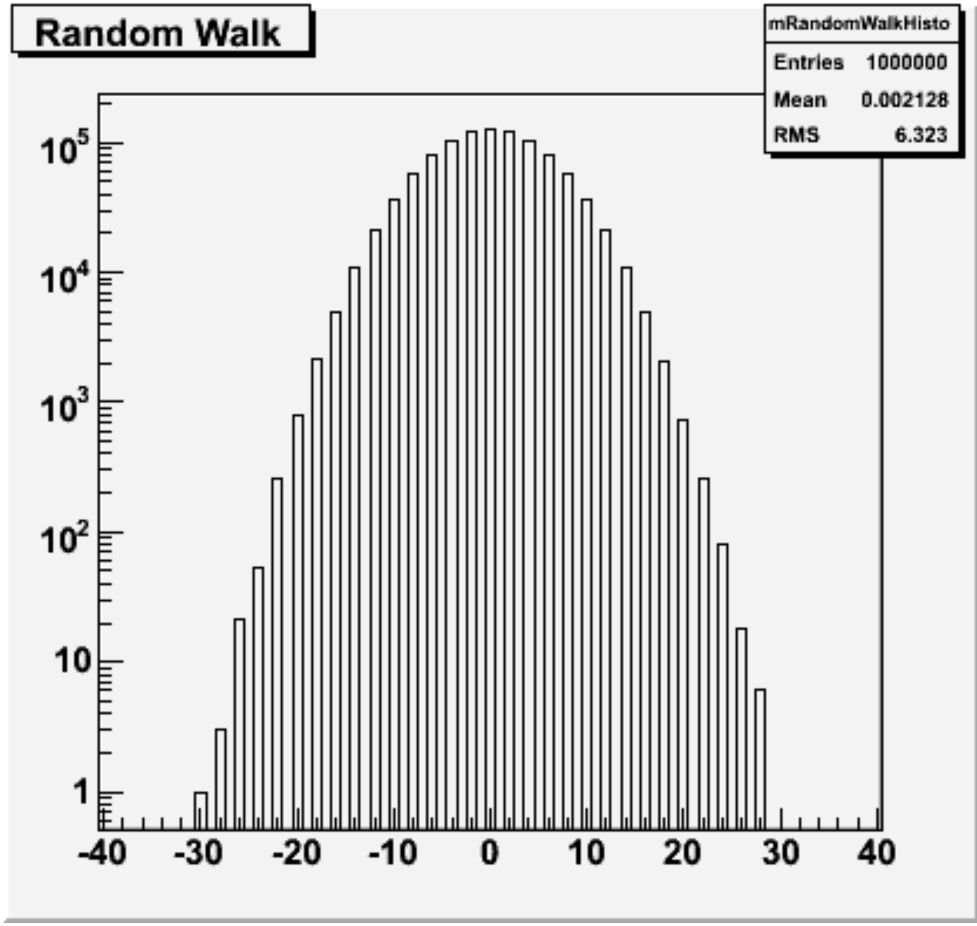
```
int numberOfSteps = 40;
 int numberOfRealizations = 1e4;


TH1D* mRandomWalkHisto = new TH1D
   ("mRandomWalkHisto","Random
   Walk",2*numberOfSteps+1,-numberOfSteps-
   0.5,numberOfSteps+0.5)
```

Code for loop

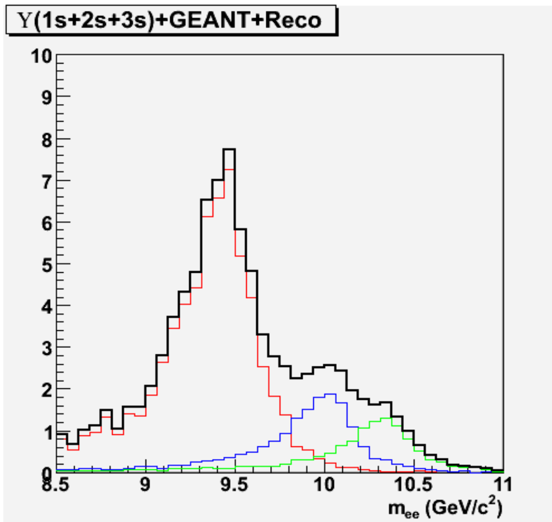
```
TRandom3 rnd3(0); // initialize random number generator with unique seed
 for (int iRealization = 0;
     iRealization<numberOfRealizations;
     ++iRealization) {
   int position = 0;
   for (int iStep = 0; iStep<numberOfSteps; ++iStep) {
     double a = rnd3.Rndm(); //random number between 0-1
     double step = 1;
     if (a<0.5) step=-1;
     // at this point, 50% of the time step will be 1
     // and 50% of the time step will be -1
     position += step;
   } // loop over steps
   //cout << "Realization " << iRealization << ", position " << position <<
    endl;
   mRandomWalkHisto.Fill(position);
 }// loop over realizations
```

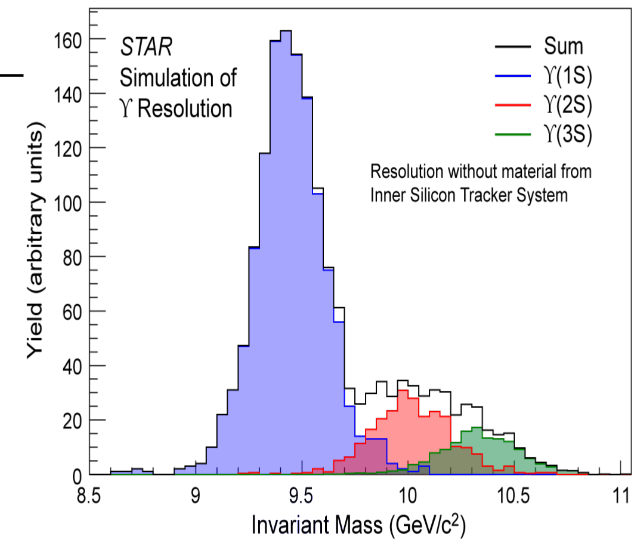
Result: Random Walk Histogram

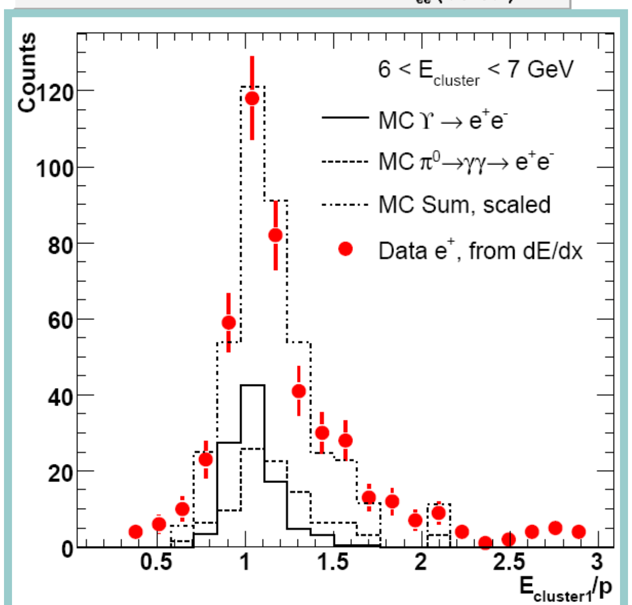


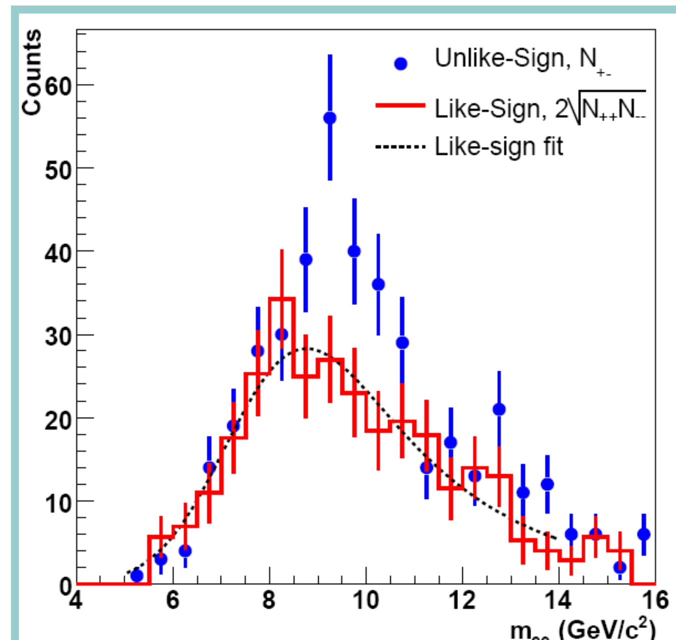


Examples of Histograms of Random Distributions









Assignment: Random-Walk

- Code the Random Walk program in ROOT.
  - Modify it to use a 2-D Histogram to do a 2-D random walk with unit length steps in which the angle that the walker describes with respect to any fixed axis is a uniformly distributed random variable on $[0, 2\pi]$. Use 40 steps, and also use unit width.
- Chapter 10 from Klein-Godunov
  - 1. Decay of monoenergetic pions. τ=2.6 x 10-8 s. E=200 MeV. Sample of 10^8 pions. How many survive after 20 m? (40 points)
  - 2. Same as 1, but with a Gaussian distribution of energies: μ_E=200 MeV, σ_E=50 MeV. (30 points)

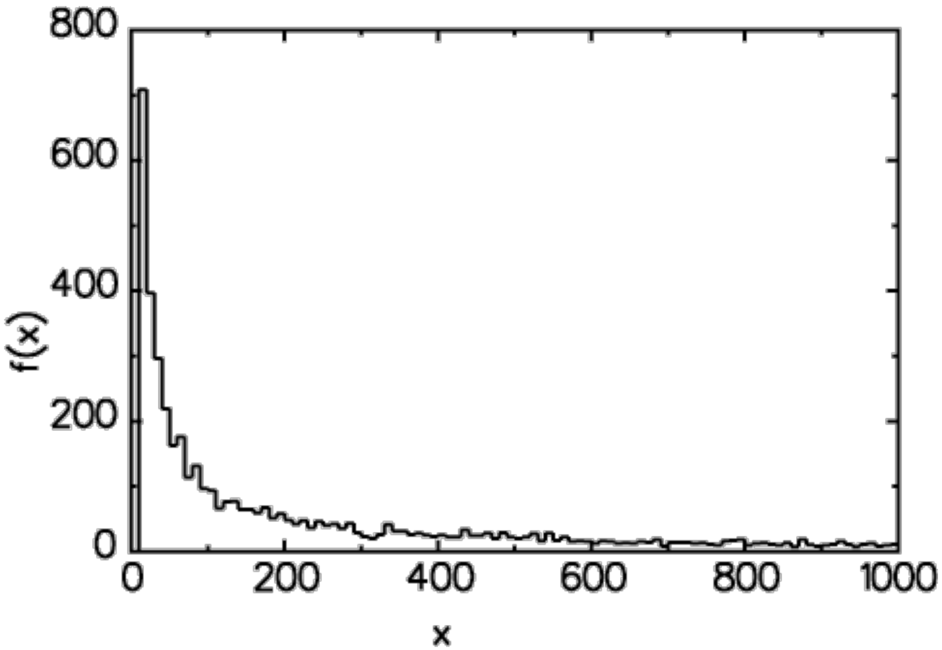
Functional Inversion Method

- **All** cumulative distributions have a p.d.f. that is uniformly distributed
  - If $y=F(x)$, then the p.d.f. $g(y)$ is uniform in (0,1) for any $F(x)$ being a cumulative distribution of a given p.d.f. $f(x)$
- Hence, the cumulative provides a mapping from the range of x to the range (0,1).
  - We can invert this! Go from a uniform back to the given p.d.f.
- Algorithm:
  - Throw r uniformly in (0,1)
  - Find x such that $\int_{-\infty}^{x} f(x')dx' = r$
  - Fill histogram of x values, it will be distributed according to $f(x)$
- Penalty: must perform the integral numerically if the function doesn't have a nice integral form

Example: 1/x distribution

- Let $f(x) = \frac{a}{x}$ for $0 < x_{min} < x < \infty$
- We need: $\int_{x_{min}}^{x(r)} \frac{a}{x'} dx' = a \ln \frac{x}{x_{min}} = r$

$$x = x_{min} e^{r/a}$$



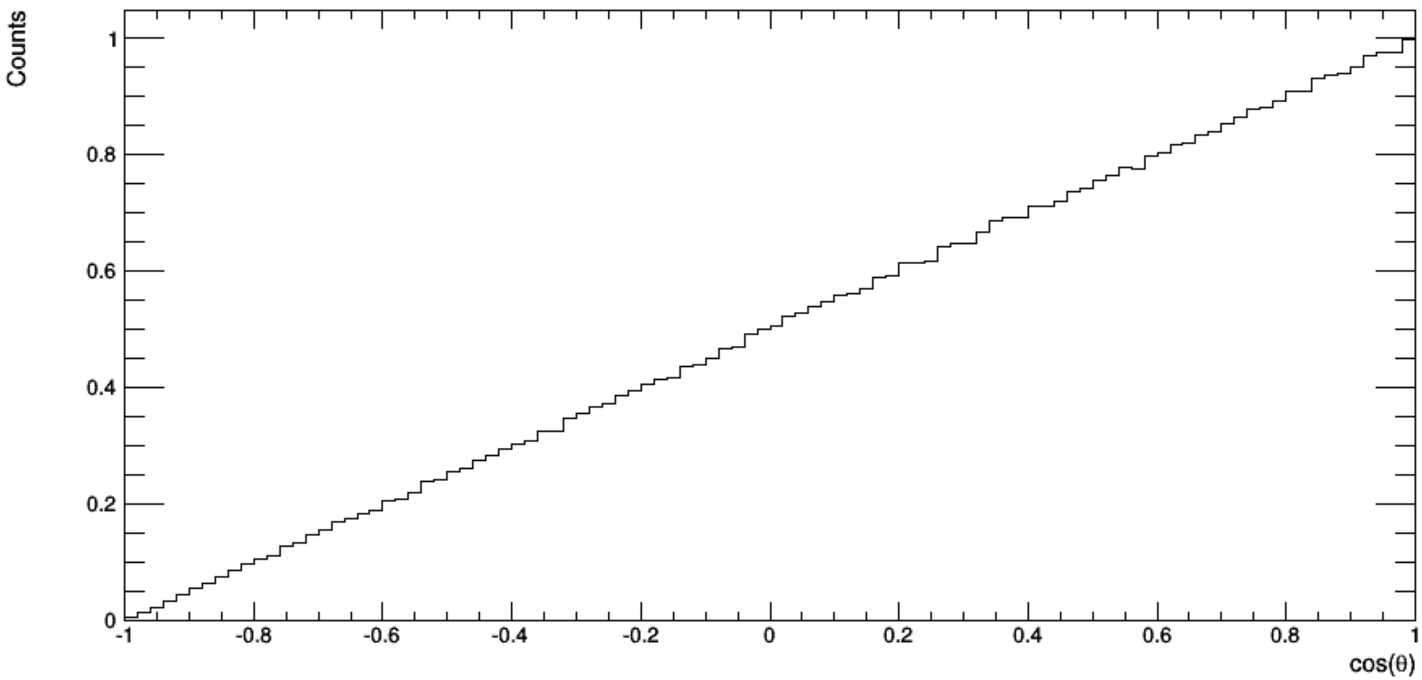
a=0.1

x_{min}=10

Example: 1+cos(θ) distribution

- Let x=cos(θ), for (-1<x<1)

$$\int_{-\infty}^{x(r)} \frac{1+x'}{2}\,dx' = r \quad \rightarrow \quad x(r) = \boxed{}$$

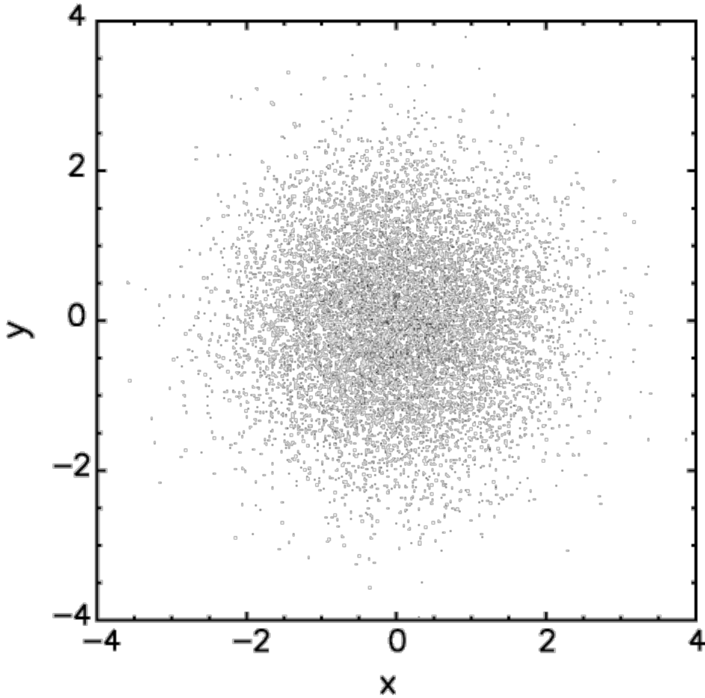


Gaussian random numbers

- Difficult to apply function inversion trick to Gaussian form
  - No closed form exists for the cumulative, $\Phi(x)$
- Other methods to get around this.
- Example: Box-Muller Transform
  - Key idea: use 2-D instead of 1-D
  - Obtain 2 Gaussian random numbers from 2 uniform random numbers
  - A 2-D Gaussian with mean=0 in both directions and equal σ is radially symmetric
  - The 2-D Gaussian *can* be integrated in closed form!
    - The extra r makes all the difference

$$dN \propto e^{-x^2/2} e^{-y^2/2} dx dy$$

$$\propto e^{-r^2/2} 2\pi r dr$$



Gaussian random number, Algorithm

- Generate ξ, ϕ using uniform distributions
- Obtain r from ξ using:

$$\frac{\int_0^{r(\xi_r)} r' e^{-r'^2/2} dr'}{\int_0^{\infty} r' e^{-r'^2/2} dr'} = \xi_r \quad \Rightarrow r(\xi_r) = \sqrt{-2 \ln \xi_r}$$

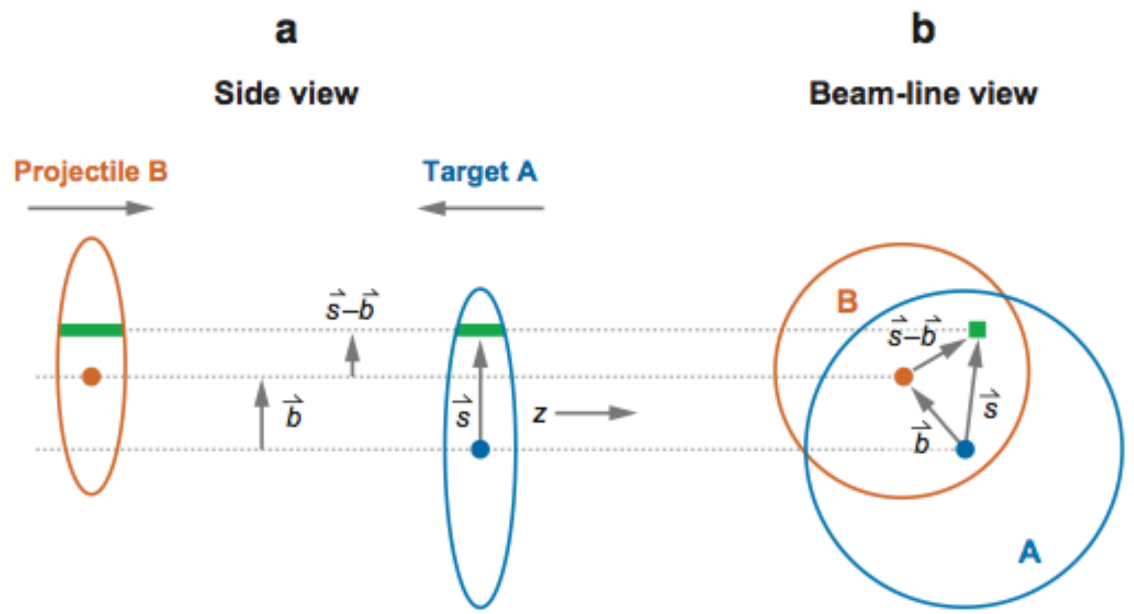
- With (r,ϕ) calculate
  - $x=r \cos \phi$
  - $y=r \sin \phi$
- (x,y) will be distributed according to Gaussian distribution

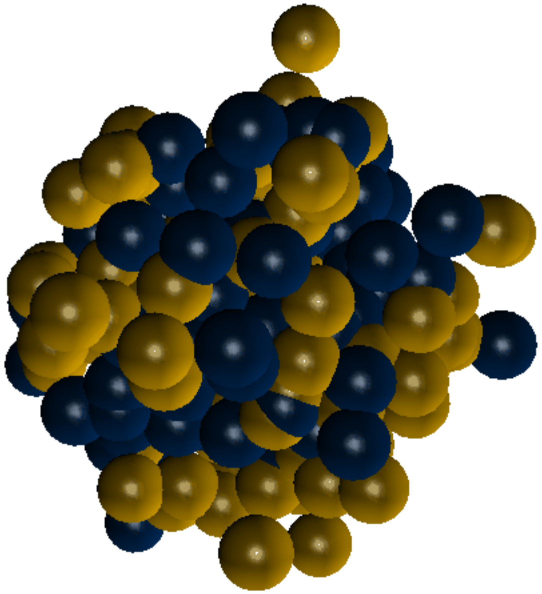
Monte Carlo Integration

- in one or two dimensions, MC integration converges slowly with N_{MC}
- in many dimensions MC integration converges much more rapidly than "grid" approaches
  - Example: Trapezoidal rule integral in d dimensions: $1/n^{2/d}$
- for high energy physics:
  - multidimensional phase space often needed
  - Monte Carlo integration almost always wins
- when we generate lots of MC events in our samples, in fact this is what we are doing: approximating analytic integrals for, say, the acceptance or observed kinematic distributions by a MC integral
- converges like $1/\sqrt{N}$ always, regardless of dimension

Monte Carlo Model of nuclear collisions

- Nuclear Collisions, Glauber model



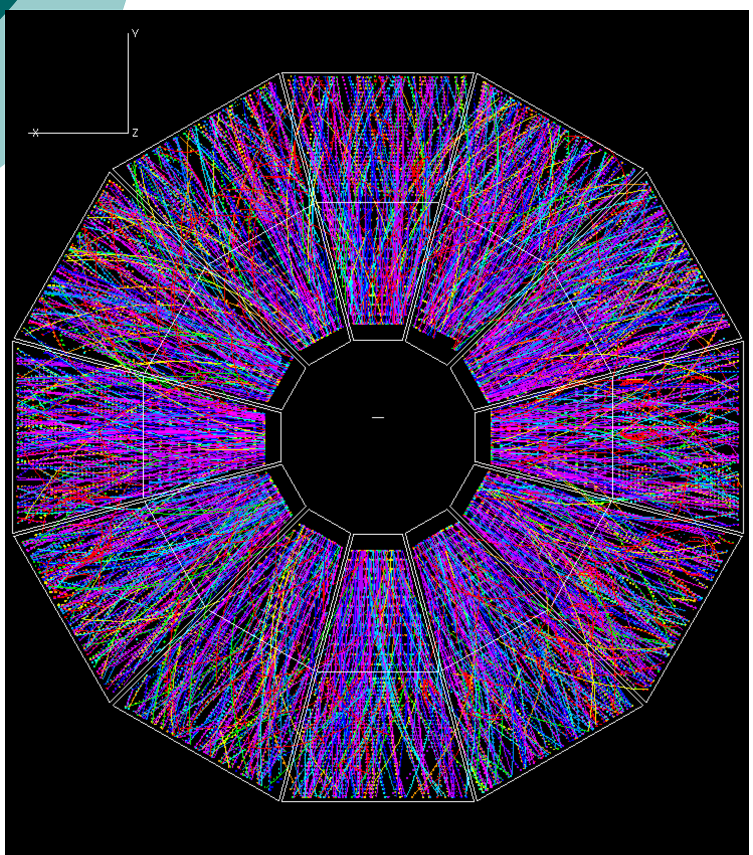


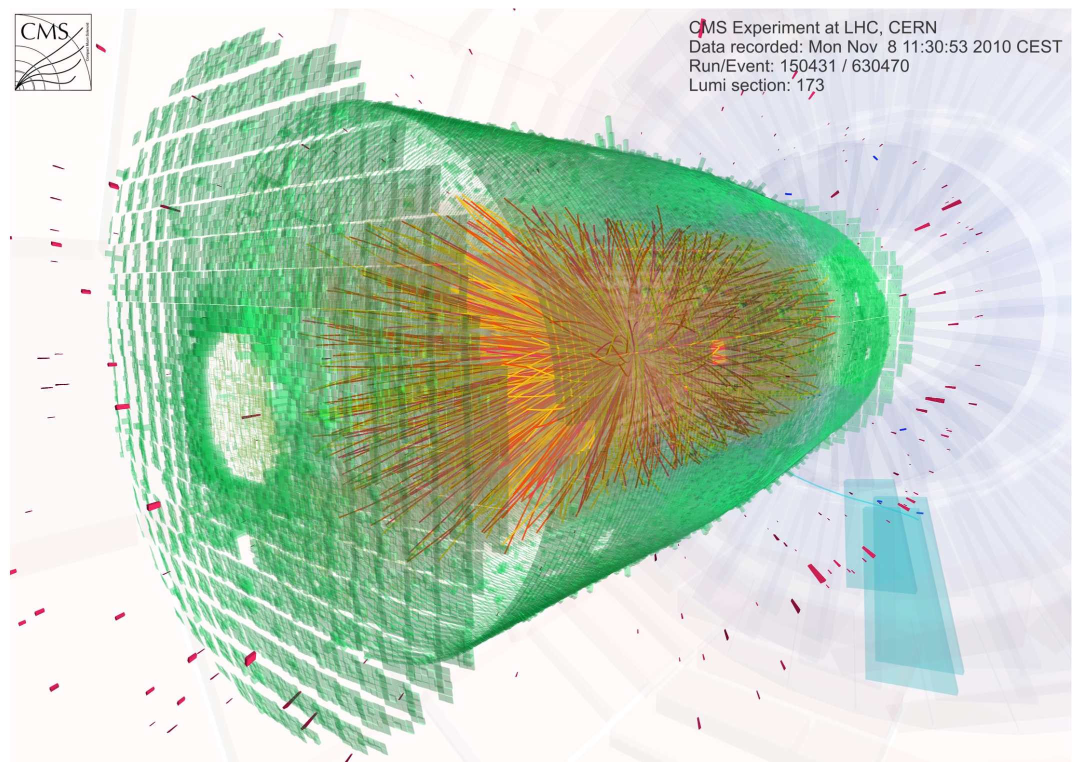
Monte Carlo Model of Nuclear Collisions

1. Nuclear Density Function
   - Make plots of the nuclear density for the Pb nucleus
2. Distribution of nucleons in the nucleus
   - Using the nuclear density function, write a function that will randomly distribute A nucleons in the nucleus (A=208 for Pb).
   - Make a plots of the x-y, and x-z coordinates of the nucleons in sample nucleus.
     - You will need to distribute them in 3D. You can use spherical polar coordinates, then convert to cartesian.

Calculate particle multiplicity

- Event displays





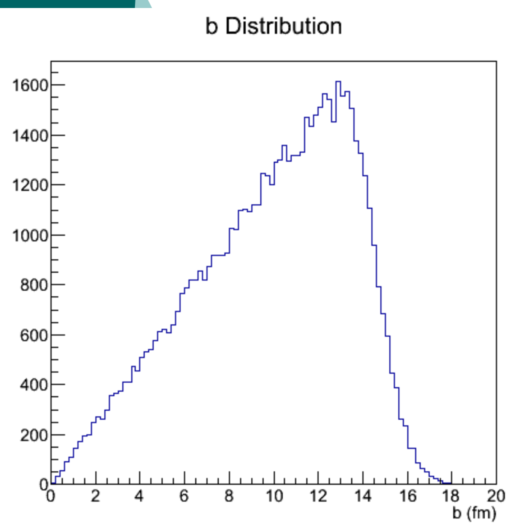
Final Project: Monte Carlo Model of Nuclear Collisions

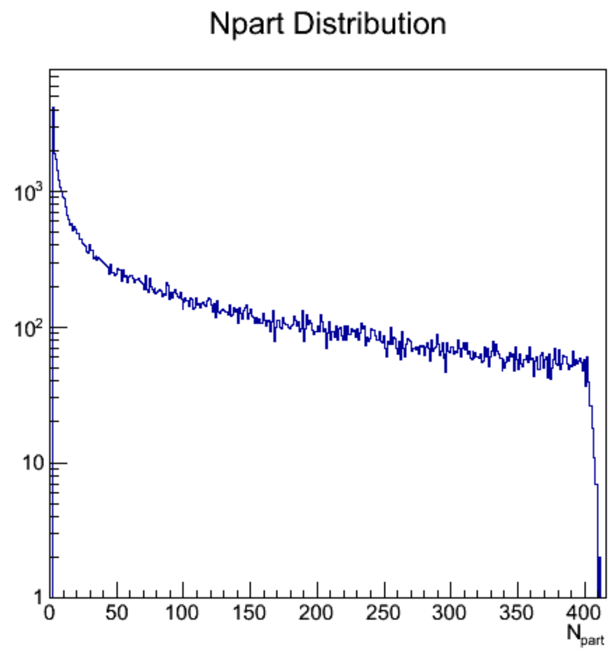
3. Impact Parameter, b
   - Make a plot of the impact parameter probability distribution
   - For b = 6 fm, make an example collision between two nuclei. Plot the x-y coordinates of the nucleons in each nucleus.
4. Number of collisions, Number of participants
   - For each pair of nucleons (one from nucleus A, one from nucleus B), check if there is a collision.
     - Nucleon-Nucleon Collision:
       - Find the distance d in the x-y plane between each nucleon-nucleon pair (the z axis is the beam axis, see slide 6)
       - Collision: when $d^2 < \sigma/\pi$. Use σ = 60 mb (where 1 b = 10^{-28} m^2).
     - Any nucleon that collides is called a “participant”. Color each participant a darker color.
     - Count the number of nucleon-nucleon collisions.

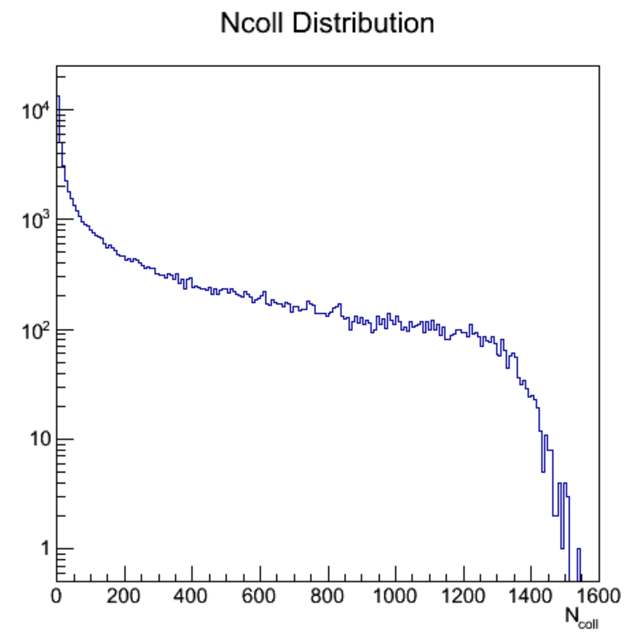
Final Project: Monte Carlo Model of Nuclear Collisions

5. Many collisions!
   - Simulate 10^6 nucleus-nucleus collision events.
   - Draw a random impact parameter from the distribution (P(*b*) proportional to *b*).
   - Calculate Npart, Ncoll for each collision.
   - For those events where there was an interaction (Ncoll>1), fill histograms of
     - the impact parameter, *b*.
     - the number of participants
     - the number of collisions
   - In part II of the project, we will model particle production, and compare it against data.

Find Npart, Ncoll, b distributions



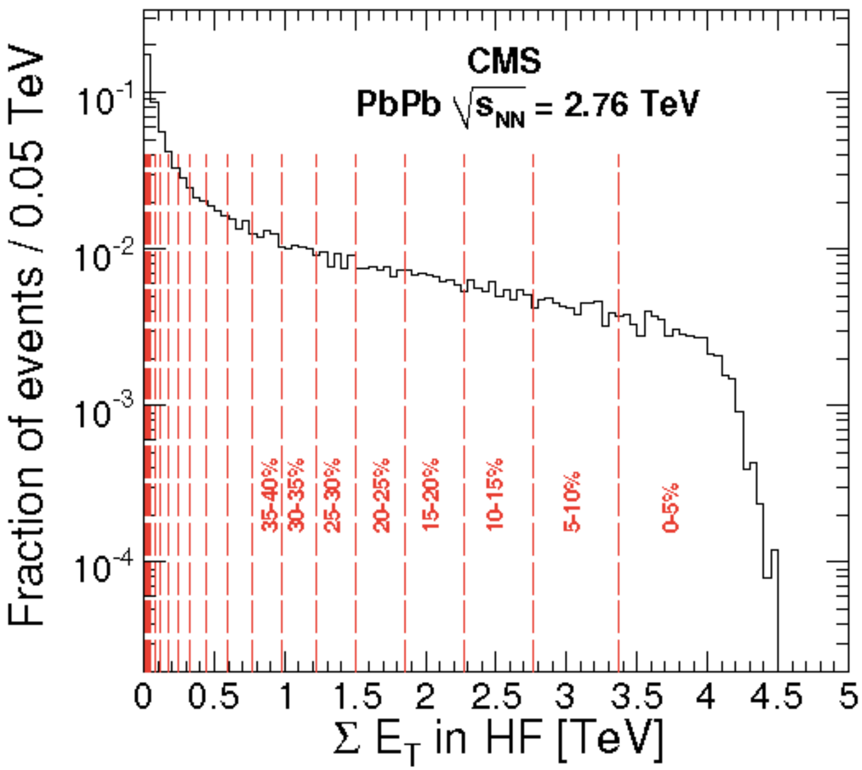




- Centrality determination in Nuclear Collisions
  - Mapping of probability
  - Highly probable events: large b, small Npart, Ncoll. "Peripheral Events"
  - Low probability: small b, large Npart, Ncoll

Comparing to Experimental data: CMS example

- Each nucleon-nucleon collision produces particles.
  - Particle production: negative binomial distribution.
- Particles can be measured: tracks, energy in a detector.
- CMS: Energy deposited by Hadrons in "Forward" region

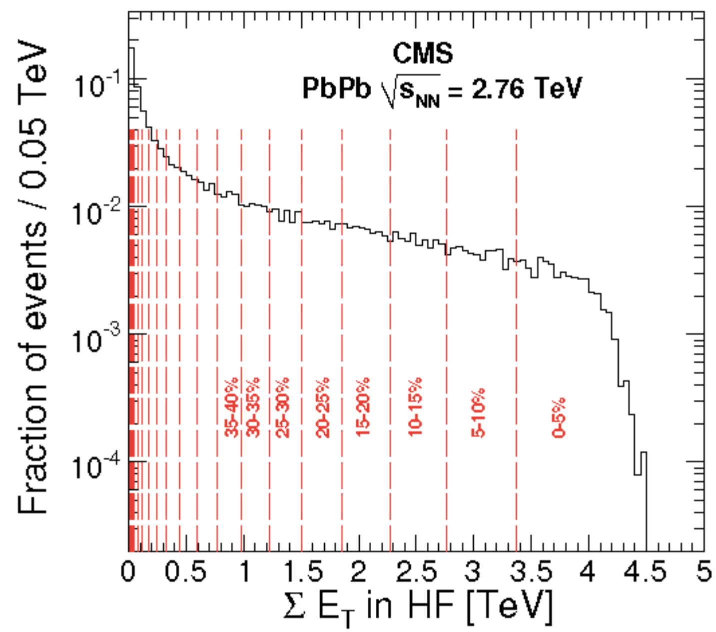


Centrality Table in CMS

From CMS MC Glauber model.

- CMS: HIN-10-001,
- JHEP 08 (2011) 141

- Estimate the numbers for the different "centrality classes" from your own calculation.
- Give average values of Ncoll, Npart, and b for centrality classes in steps of 10% of the total Ncoll distribution.



Centrality	0-5%	5-10%	10-15%	15-20%	20-25%	25-30%
N_{part}	381 ± 2	329 ± 3	283 ± 3	240 ± 3	203 ± 3	171 ± 3
Centrality	30-35%	35-40%	40-45%	45-50%	50-55%	55-60%
N_{part}	142 ± 3	117 ± 3	95.8 ± 3.0	76.8 ± 2.7	60.4 ± 2.7	46.7 ± 2.3
Centrality	60-65%	65-70%	70-75%	75-80%	80-85%	85-90%
N_{part}	35.3 ± 2.0	25.8 ± 1.6	18.5 ± 1.2	12.8 ± 0.9	8.64 ± 0.56	5.71 ± 0.24

Relativistic Kinematics

- You are given:
  - Kinetic Energy, Pion Mas
  - And we know: $E = \gamma mc^2, KE = E - mc^2$
  - Therefore: $\gamma = \frac{E}{mc^2} = \frac{KE + mc^2}{mc^2} = \frac{KE}{mc^2} + 1$
  - From which we can get the pion velocity: $\beta = \sqrt{1 - 1/\gamma^2}$

Lorentz Boost transformations

- Given the boost velocity β, and the corresponding Lorentz factor γ, if we measure the time and space separation of two events in one frame, the corresponding separations in the frame moving with velocity β are given by:

$$\begin{pmatrix} c\Delta t \\ \Delta x \end{pmatrix} = \begin{pmatrix} \gamma & \beta\gamma \\ \beta\gamma & \gamma \end{pmatrix} \begin{pmatrix} c\Delta t' \\ \Delta x' \end{pmatrix}$$