In the colliding of nuclei of different elements, many nucleons are scattered in directions adhering to physical laws, but the events occur so rapidly on scales so small that it is nigh impossible to preemptively determine how many nucleons will interact in the collision. The Glauber Model, named after Roy Glauber, is a technique used to predict the physical values of interest, namely number of binary collisions and participating particles as a function of impact parameter. The two types of Glauber models include the Optical-limit approximation and Monte Carlo model. The optical approach makes several assumptions about the colliding nuclei that allow one to create analytical expressions that describe the impact and determines the number of interacting nucleons along with the number of binary-binary collisions. The Monte Carlo follows a more geometric approach by distributing nucleons radially following a nuclear charge density profile unique to each element. The nucleons are given random azimuthal angles and a projection of this distribution is utilized to determine which individual nucleons participate in the collision. This approach assumes the nucleons travel straight throughout the entire event and can collide with many opposing nucleons without having its path deterred. The simulations in this report exclusively use the Monte Carlo approach to Glauber modeling.

The Monte Carlo approach requires the nuclear charge distributions in order to build realistic nuclei to collide. Several methods can be used to create this distribution (listed below); all of which are well defined thanks to De Vries’ et al. 1987 paper *Nuclear Charge Density by Elastic Electron Scattering* (). An example is the two parameter Fermi model that is used to create 197-gold nuclei, where a wood-saxon density profile is created from a mean field potential on the nucleons. The equation describes the force felt by each nucleons, and the potential can be utilized to map out a probability function for the radial position of each nucleon. In all methods, a distribution function, *ρ(r)*, is sampled from to give each nucleon in the nucleus a certain radial position, *r*. The nucleon is then assigned to different azimuthal directions that allow the nucleus to be built in three dimensions. When sampling from the radial distribution, the Distribute1D function samples so that no radial position on the sphere is more likely to be chosen than another. The azimuthal directions are sampled from the arccosine function from -1 to 1 and –π to π. This gives a flat, even distribution over the entire sphere that prevents oversampling near the poles. In essence, sampling in this fashion is the spherical equivalent to sampling over a uniform function in one dimension.

The interaction distance at which two nucleons can be considered to have collided is also needed to run the program. This distance is directly related to the inelastic cross section of the nucleons, itself a function of beam energy. Since we are only colliding different ions, we need to understand the most basic ion collision, proton-proton. If one can accurately model a proton-proton collision, then one can model heavier ion collisions as a conglomeration of many individual proton-neutron collisions. The particle data group gathers large amounts of data about elastic and total proton-proton cross section from many experiments over the world, and compiles all this data in one compact source. The program pulls the data in real time and fits curves to both elastic and total cross sectional areas. The proton-proton inelastic cross section, , is given by the curve of the total cross section minus the elastic cross section. The cross sectional area is converted to a radial distance using the equation that relates area of a circle to radius: or .

This program correlates number of interacting particles and collisions to impact parameter, b, which is simply the distance between the centers of the nuclei during the collision. Because we are assuming every individual nucleon travels straight throughout the collision, the impact parameter can be drawn as the distance between the centers of the nuclei when projected onto a flat plane. The applicable use of this data is to allow insight on the collisions, such as the ones taking place at the relativistic heavy ion collider (RHIC), to determine in how many particles are being created as a result of the collision. Figure 1 shows the results for RHIC following similar procedures to present the relation between number of participating nucleons, Npart, number of collisions, Ncoll, and impact parameter for gold-gold and copper-copper collisions at center-of-mass energies, , of 200 GeV.

The code is entirely open source, native to iPython, importable, and freely available on Github. Emphasis was put on accumulating all the necessary data needed to run these types of simulations with ease of use. The following steps outline how to run the program for Au+Au and Cu+Cu collisions at 200 GeV. –Notebook code+comments on running the code with explanation on what is happening at each step- We can compare the results of the program with expected results of simulations run at RHIC for gold-gold and copper-copper collisions at center-of-mass energy 200 GeV.

RHIC will also be performing He-3 and Au-197 collisions at 200 GeV in the future. The program can easily simulate these types of collisions to provide expected outcomes of the collisions. –Code for He-3+Au-197 Collision + Results- We see a large spread in number of collisions and number of participants for similar impact parameters, a phenomenon that we observe in the collision of ions of greatly different sizes. This signifies that the orientation and geometry of the ions upon impact greatly affects the result of the collision (±5/48, 10.4% for perfect collisions increasing for larger impact parameter), but did not have as much an effect as the impact parameter itself. We contrast this to gold-gold collisions, where the impact parameter was essentially the sole determining factor for number of collisions observed and specific geometry had little effect other than to provide minuscule deviations in the data (±110/1500, 7.3% for perfect collisions).