Fission Core Simulation using Swift 3

Diego Gutierrez Coronel*
IIT Department of Physics

(Dated: May 7, 2018)

This paper presents code developed to calculate certain processes of an atomic fission core in Swift 3. The program calculates different parameters that affect criticality of an spherical bomb core. It allows the user to select different sources for the bomb core, such as Enriched Uranium (U-235) and Plutonium 239. The program calculates the criticality conditions for different configurations such a bare or tampered core, as well as for different tamper materials. It also estimates the yield and efficiency of such exploding configurations. A simple animation and simulation of neutron generation and induced fission is presented using Monte Carlo methods.

Neutron induced fission was an essential discovery for the development of Nuclear Weapons in the mid nineteenth century. Before the development of the first nuclear weapons, theoretical predictions bolstered the race to develop one with Enriched Uranium U-235 as a source. It was predicted that this rare isotope of Uranium would fission under bombardment of slow neutrons with a very high probability. These slow neutrons travel at a couple thousand meters per second. Depending on the velocity of the bombarding neutrons and the nucleus struck, neutrons can undergo different processes. They could be captured by a nucleus and then this would decay to a more stable nucleus. It could scatter and deflected from the nucleus. Or it could generate fission, splitting the atom into two or three lighter nuclei, releasing a couple or more neutrons depending on the fission mechanism.

There are a few simple models for these processes, and given different assumptions, the physics of nuclear fission applied to the an atomic bomb can be understood. A simple model of an spherical nuclear bomb core is presented, demonstrating the conditions necessary to undergo fission. The demonstrations shows two possible configurations a bare core where neutrons can escape from the material. The second configuration is a bomb core surrounded by a neutron-reflective material. This is known as the tamper material, which enhances the efficiency of the bomb. A simple animation and simulation is presented to model these processes. This shows the fission-generated neutrons and their scattering inside the bomb core.

The first part of the code references a series of publications by B. Reed, on the physics of the Manhattan Project and simple educational models on nuclear weapons for physics students.

I. DISCUSSION OF THEORY

As mentioned before, in fission, a nuclei of a heavy element such as U-235 or Pu-239 splits into lighter nuclei. An example of a such reaction is shown by the equation below.

$$ {}^{1}_{0}\mathrm{n} + {}^{235}_{92}\mathrm{U} \rightarrow {}^{141}_{56}\mathrm{Ba} \rightarrow {}^{92}_{36}\mathrm{Kr} + 3\left({}^{1}_{0}\mathrm{n}\right) \tag{1} $$

This neutrons generated will induce a chain reaction. The energy released or consumed could be estimated by the excess masses from the inputs and products of the reaction. This can be in the order of a hundred Mev for a nuclear fission shown by (2), whereas common chemical reactions can be in the order of a few eV [3].

Other important parameters involved in the chain reaction are the fission and scattering cross sections, σ_f and σ_s. The sum of these two is the total or transport cross-, designated as σ_{trans}. The cross sections also relate to the mean free path of fission λ_f and transport $\lambda_t rans$ in the following way.

$$ \lambda_{trans} = \frac{1}{\sigma_{trans} n}, \lambda_f = \frac{1}{\sigma_f n} \tag{2} $$

Where n is the density of nuclei that can be calculated from the atomic mass and density of the source. The cross sections thus affect how far neutrons travel before they scatter or generate a fission process. Consequently influencing the timescales.

Similarly the number of secondary neutrons created at each fission, ν, is an important parameter. The values for U-235 and Pu-239 are 2.637 and 3.172, respectively. These parameters will be important to describe the program, specially the animation and simulation. The core radius is also an important parameter, present in the conditions for criticality.

I.1. Criticality Calculations

Since neutrons have a probability of scattering and thus escaping from the core. For the chain reaction to be maintained, the rate of the number of neutrons generated have to at least be equal to rate of neutrons lost. This point is known as criticality.

The equation that describes criticality for a spherical bare core of radius R_{core}.

* dgutie10@hawk.iit.edu

$$\left(\frac{R_{core}}{d}\right)cot\left(\frac{R_{core}}{d}\right)+\frac{1}{\eta}\left(\frac{R_{core}}{d}\right)-1=0 \qquad (3)$$

Where the parameters d and η are described by the following

$$d=\sqrt{\frac{\lambda_{core}^{fiss}\lambda_{core}^{trans}}{3\left(\nu-1\right)}} \qquad (4)$$

$$\eta=\frac{2\lambda_{trans}}{3d} \qquad (5)$$

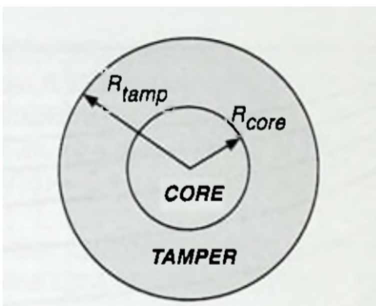

FIG. 1. Schematic of tampered core nuclear bomb

For a spherical tamper core, the equation that is given by the following:

$$\left[1+2\lambda\epsilon\left(\frac{d}{R_{core}}\right)\left(\frac{R_{core}}{R_{tamp}}\right)^{2}-\frac{R_{core}}{R_{tamp}}\right] x\left[\left(\frac{R_{core}}{d}\right)cot\left(\frac{R_{core}}{d}\right)\right]+\lambda=0 \qquad (6)$$

R_{tamp} is the radius of the tamper sphell and λ is the ratio of the transport mean free paths of the two materials (i.e. $\lambda_{tamp}^{trans}/\lambda_{core}^{trans}$). Both equations 3 and 6 define the criticality radius and thus the criticality mass multiplying the volume of the sphere by the respective density of the source. Similarly the mass of the tamper material can be found. Both can be solved numerically. However, the second conditions requires a specified ratio of the core and the tamper material shell.

One caveat to solving this equation numerically is that the radius of the core and the tamper material change with time. Therefore the criticality radius is solved for the initial core radius. After this the resulting shutdown radius is given by the following equation that describes the distance the core has expanded.

$$\Delta R=R_{shut}-R_{core}=R_{core}\left(\sqrt{\frac{R_{core}}{R_{bare}}}-1\right) \qquad (7)$$

Where R_{bare} is the criticality radius found by equation 3 and 6. Similarly, the compression ratio C_{final} defined by the cube root of the ratio of the initial core radius R_{core} and the final shutdown radius R_{shut}. Is an important parameter defining the expansion of the radius during explosion.

I.2. Efficiency and Yield Calculations

The yield which is the energy energy released by the exploding bomb, and is commonly expressed as the equivalent energy released by a certain mass of TNT. For the program the units are in kilotons of TNT ($kt = 4.2\text{x}10^{12} Joules$). The yield for a bare core is estimated by the total energy released at the shutdown time t_{shut}.

$$E\left(t_{shut}\right)\approx\frac{\alpha^{2}M_{total}\Delta R^{2}}{8\tau^{2}} \qquad (8)$$

Here alpha is also an approximated parameter known as the effective number of neutrons liberated per fission. On the other hand the yield calculated for the tampered core configuration does not depend on the same parameter $alpha$ but instead is proportional to $ln\left(\nu\right)^{2}$. And the total mass M_{total} contains the mass of the tamper shell.

I.3. Time and Numerical Scales

Timescales of the processes that neutrons undergo in a bomb core will be important to describe the animation later on. As mentioned before the cross-sections affect the mean free path and these at the same time influence the time it takes for fission or scattering events to occur in the bomb core. As seen in reference [3], the time it takes to fission a kilogram of U-235 is in the order of a microsecond. However, for this estimate no losing or escaping of neutrons is assumed. After all criticality is achieved when the rate of fission equals that of escaping neutrons. Therefore, a simulation that takes into account the escape of neutrons and the caveat that as more fission events occur the probability of fission is higher. Similarly, assuming the mean speed of neutrons inside the core is close to $2x10^{7}m/s$, the time between fission events is found to be in the order of almost 10 ns. The actual time for U-235 and Pu-239 is $8.36\text{x}10^{7}m/s$ and $7.23\text{x}10^{7}m/s$. It is also worth noting that the fission mean free path, i.e. the distance neutrons travel before fission on average, is almost twice the value of one critical radius for a bare core. This means that the probability of neutrons escaping or causing fission scales with the radius. This will be considered in the simulation.

II. DISCUSSION OF THE CODE AND PROGRAM

The executable of the program displays a GUI with two tabs. One shows calculations such as the resulting parameters from conditions met for criticality, such as the critical mass and radius, and the efficiency and yield of a nuclear weapon with a particular configuration and source. The second tab is comprised of an animation of a 2-dimensional representation of a nuclear bomb and the generation of neutrons from fission as well as the scattering of the neutrons inside the spherical core.

II.1. Criticality, Efficiency, and Yield Calculations

The graphic user interface lets the user select two of the regular sources for a bomb core such as Uranium-235 and Pu-239. It also allows for a user defined material with the following parameters: Density(g/cm^3), atomic weight(g/mol), elastic scattering cross section(bn), fission cross section(bn), secondary neutrons per fission.

Similarly, it lets the user to select the bare or tampered configuration. For the latter configuration, a material can be chosen from, Aluminum, Beryllium Oxide, Lead, Tungsten-carbide, and Depleted Uranium (U-238). By pressing the button calculate the program will calculate the critical mass and radius of the selected configuration as well as two plots. One showing the function that defines the criticality condition. The second being a plot for the Yield as a function of tamper mass (for tamper configuration). The condition for criticality described by equations 3 and 6 is solved numerically by using simple Newton-Raphson Method to find the root of this function.

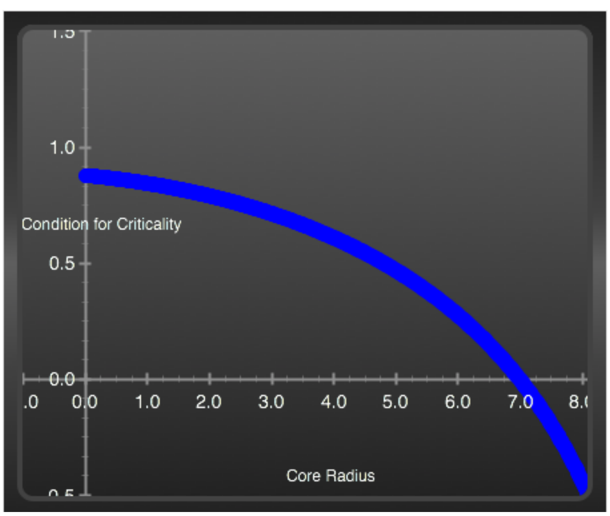

FIG. 2. Function describing the condition for criticality for Uranium-235 core with a tamper shell of Tungsten-Carbide. Same configuration as the Little Boy.

The user can also input simple parameters such as the mass of the bomb core, and the mass of the tamper material, for simplicity. The program will output the resulting number of critical masses in the core and the ratio of the tamper and core radii, as well as the calculations for the yield, efficiency and the distance the core radius expanded.

II.2. Animation / Simulation

This animation was created with the purpose to generate an interactive simulation that would as accurately as possible resemble the generation of neutrons before a nuclear core reaches criticality. The animation is a simple NSView class display with a timer that iterates a function animation() that uses random sampling to generate neutrons in the core.

Again the core of the fissile material is represented by a sphere. The animation shows a cross-sectional view of the a bare core. The program inputs a number of iterations, the initial number of neutrons that hit a nucleus. The first thing the animation does is generate a the initial number of points in the display view box with random positions. It then goes over the points that were outside the core radius and marks them as red points or neutrons that escaped the core. For those inside the core a random number, say r, was generated to define the point as a fission or an scattering event. Thus the probability of fission is given as the ratio of the fission and total cross section (i.e. $r \leq \sigma_f/\sigma_{tot}$), a scattering event would be given by the complement of this ratio. For the scattering events, the position of the neutron in the next time step is calculated to check if the neutron would escape or if it should be counted in the next time step. Following this, different counters would record the number of fissions, escaping neutrons and alive neutrons to check fission at the next time step. The program then outputs the number of fissions, escaping neutrons, and the rate of each one of these events.

The time step for each source is taken to be the time it takes a neutron to create another fission or elastically scatter, in the order of 10^-9s. Therefore after each time step the neutrons have traveled on average the distance given by the transport mean free path for each material (for example, 3.59 cm for U-235).

III. DISCUSSION OF THE RESULTS

III.1. Calculations

The calculations were compared with those from the references. An example of the results for different configurations of the bomb core are shown by the table below.

The values for the tamper of Tungsten-carbide shown in table I are the same values used to estimate the yield and efficiency of the Little Boy bomb. Calculations are very similar to those in reference [1]. As mentioned in

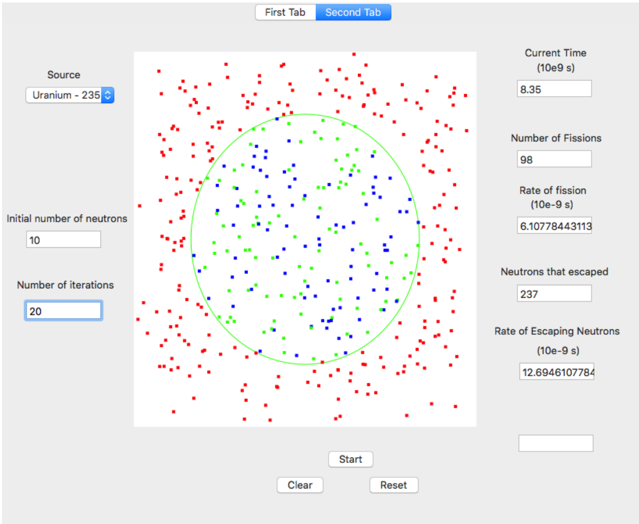

FIG. 3. GUI window displaying the animation of an Uranium-235 core. The red points represent neutrons that have escaped, green points represent fission events, and blue points represent scattering events.

TABLE I. Example of output calculations from the program for a tamper core configuration with 65.8 kg of U-235 and a tamper mass of 552kg

Tamper	Efficiency(%)	Yield(kt)	ΔR
Beryllium Oxide	3.59	40.2	3.05
Tungsten-carbide	2.72	30.5	2.08
Depleted Uranium	1.85	20.7	1.45
Aluminum	1.79	20.0	1.41
Lead	1.48	16.6	1.22

this paper the configuration was not exactly the same and there is a a lot of variation changing parameters such as the mass or the secondary neutrons generated for the particular source.

They also tried to calculate these parameters for the Fat Man bomb deployed at Nagasaki. The configuration for Fat Man, was a spherical core of 6.3 kg of Pu-239 and a tamper shell of Depleted Uranium of 229.6 kg. The values found by the program presented on this paper were a yield of 2.63 kt and an efficiency of 2.45%. It differs from the numbers found in [1]. This is mostly because they allow the initial and final radii to change with different initial and final compression ratios.

III.2. Animation / Simulation

The animation runs well for reasonable values. For instance values of the secondary neutrons generated in between 2-5, number of initial particles of values between 10-20.And a number of iterations in between 10 - 40. The variable coefficient radii in the code allows for an increasing probability of neutrons generated to be inside the core. Allowing the fission rate to increase with time. The final results from the total number of fission vary given the random sampling however it estimates the number of nuclei that have fissioned by looking at the total number of neutrons. It is worth noting, that this is just an estimate for a small number of particles, it is clear that recording the actual position of each number of neutrons, instead of randomly sampling the events, would be considerably more computationally expensive. The number of calculations would increase exponentially as ν_{eff}^G where ν_{eff} is the effective number of secondary neutrons generated, and G is the number of generations.

IV. CONCLUSIONS

To conclude the program works well for calculations and estimations of the yield and the efficiency of different configurations of bomb cores. It varies for very slight changes in different parameters such as the secondary neutrons per fission, the tamper characteristics, among others. But this allows to see the differences between the different configurations and even user-defined sources. There are a few variations from other calculations such as those in reference [1]. In that paper, they allow to change the compression ratios however this are not guaranteed to satisfy criticality conditions. This program directly solves for them and directly implements in the yield and efficiency calculations.

The animation is a good example of the processes neutrons undergo during explosion. Different things could have been added to the animation such as a time varying radius, which would agree with the conditions and calculations for criticality. Similarly, multiprocessing would have been something that could have benefited the program to reduce running time. The animation could also be extended to implement the reflection of neutrons by a tamper configuration. This would show a decrease in the time needed to reach criticality. For this the neutrons that are sampled as escaping neutrons would have an additional calculation to check if in the next time step some will reflect back to the core.

[1] B. C. Reed, American Journal of Physics 86, 105 (2018).
[2] B. C. Reed and K. Rohe, American Journal of Physics 82, 972 (2014).
[3] B. C. Reed, The Manhattan Project: a Very Brief Introduction to the Physics of Nuclear Weapons (Morgan & Claypool Publishers, San Rafael, CA, 2017).