Federal University of Rio Grande do Sul
School of Engineering
Department of Electrical Engineering
Undergraduate Program in Electrical Engineering

Raul Scapini Weiand

A Particle Simulator for the Determination of Cross Sections from Interaction Potentials

Porto Alegre
2023

Raul Scapini Weiand

A Particle Simulator for the Determination of Cross Sections from Interaction Potentials

Final paper submitted to the Undergraduate Program in Electrical Engineering of the Department of Electrical Engineering of the Federal University of Rio Grande do Sul, as a partial requirement for obtaining the title of Bachelor of Electrical Engineering.
Advisor: Prof. Dr. Bardo Ernst Josef Bodmann

Porto Alegre
2023

CIP - Catalogação na Publicação

Weiand, Raul Scapini
A Particle Simulator for the Determination of Cross Sections from Interaction Potentials / Raul Scapini Weiand. -- 2023.
35 f.
Orientador: Bardo Ernst Josef Bodmann.

Coorientador: Ivan Müller.

Trabalho de conclusão de curso (Graduação) -- Universidade Federal do Rio Grande do Sul, Escola de Engenharia, Curso de Engenharia Elétrica, Porto Alegre, BR-RS, 2023.

1. Simulador. 2. Partículas. 3. Potenciais. 4. Simulator. 5. Particles. I. Bodmann, Bardo Ernst Josef, orient. II. Müller, Ivan, coorient. III. Título.

Elaborada pelo Sistema de Geração Automática de Ficha Catalográfica da UFRGS com os dados fornecidos pelo(a) autor(a).

Raul Scapini Weiand

A Particle Simulator for the Determination of Cross Sections from Interaction Potentials

Final paper submitted to the Undergraduate Program in Electrical Engineering of the Department of Electrical Engineering of the Federal University of Rio Grande do Sul, as a partial requirement for obtaining the title of Bachelor of Electrical Engineering.
Advisor: Prof. Dr. Bardo Ernst Josef Bodmann

Approved on: Porto Alegre, 11 de maio de 2023.

EXAMINATION BOARD:

Prof. Dr. Bardo Ernst Josef Bodmann (Advisor)
Federal University of Rio Grande do Sul (UFRGS)

Prof. Dr. Volnei Borges
Federal University of Rio Grande do Sul (UFRGS)

Prof. Dr. Luiz Paulo Luna de Oliveira
Instituição do membro da banca

ABSTRACT

This graduation project aims to study simulations of interactions between particles in the context of transport theory considering thermal energies. The principal question addressed in this work is how to translate simulated interaction rates into cross sections, which are key quantities for more complex neutron matter interaction simulations. In the present implementation the simulator materials were considered with a single global temperature. There is no closed theory in low energy nuclear physics and there exist a variety of simulators that, to a large extent, follow established paradigms from the 1970s. As a difference in the present project a physical Monte Carlo was employed instead of a mathematical one. A case study based on different interactions and temperatures is presented and the related cross sections are derived. The results have been compared with the theory, since there is no experimental results to validate the simulator.

Keywords: Simulator; Particles; Potentials; Cross sections.

RESUMO

Este projeto de graduação visa estudar simulações de interações entre partículas no contexto da teoria de transporte considerando energias térmicas. A principal questão abordada neste trabalho é como traduzir taxas de interação simuladas em seções de choque, que são quantidades chave para simulações mais complexas de interação de nêutrons com matéria. Na presente implementação, os materiais do simulador foram considerados com uma única temperatura global. Não existe uma teoria fechada em física nuclear de baixa energia e existe uma variedade de simuladores que, em grande parte, seguem paradigmas estabelecidos na década de 1970. Como diferencial no presente projeto foi empregado um Monte Carlo físico ao invés de um matemático. Um estudo de caso baseado em diferentes interações e temperaturas é apresentado e as respectivas seções de choque são derivadas Os resultados foram comparados com a teoria, uma vez que não há resultados experimentais para validar o simulador.

Palavras-chave: Simulador; Partículas; Potenciais;.

SUMÁRIO

1 INTRODUCTION

At present, Brazil has two nuclear power plants in operation (Angra 1, 640 MWe gross / 609 MWe net and Angra 2, 1350 MWe gross / 1280 MWe net), and one under construction (Angra 3, 1405 MWe gross). The date of commercial operation is estimated for the end of 2027. Angra 1, 2 and 3 share the same location, Itaorna beach (Praia de Itaorna), municipality of Angra dos Reis, about 130 km from Rio de Janeiro as identified by the Federative Republic of Brazil (2022) [1].

Previous long-term planning studies on energy policy have outlined the desirability of building four new nuclear power plants in Brazil. The Ten-Year Energy Expansion Plan 2031 (Plano Decenal de Expansão de Energia 2031 - PDE 2031), issued by the Minister of Mines and Energy of Brazil through one of its organizations, Energy Research Company (Empresa de Pesquisa Energética - EPE), presents alternatives for the resumption of the Brazilian Nuclear Plan, which includes new plants by 2031.

On December 16, 2020, the Ministry of Mines and Energy (Ministério de Minas e Energia - MME) approved the National Energy Plan 2050 (Plano Nacional de Energia - PNE 2050). Prepared by EPE, based on MME guidelines, the plan is a support tool for the design of the planner's long-term strategy in relation to the expansion of the energy sector. In this sense, the Brazilian nuclear program, which began in the 1970s, placed Brazil at an international level in a competitive way in the nuclear area. It is common to have electrical engineers in this area, even among the pioneers in the initial phase. History itself shows the role of electrical engineers in the nuclear area. The author of this diploma work developed an interest in the nuclear area, among others, arising from disciplines studied during the undergraduation program.

A decade ago, the activities of the nuclear study group (Grupo de Estudos Nucleares – GENUC) at the Federal University of Rio Grande do Sul (Universidade Federal do Rio Grande do Sul - UFRGS) began a project to create a Brazilian simulator for neutrons transport for various applications, from energy production via fission reactions, to applications in medicine settings such as nuclear diagnostic and therapeutic methods, for example: Boron Neutron Capture Therapy (BNCT).

The simulator, despite of being in the development phase, already allows to carry out a variaty of neutron transport simulations. The implementation is based on

the physical Monte Carlo method, which simulates reactions between neutrons and target nuclei in analogy to how processes occur in real scenarios, unlike most simulators that are used to tackle issues such as criticality, shielding, among others, with effective and adaptive models.

The philosophy being followed only became possible because of changes and paradigms in programming, for example: parallelization, OpenMP and MPI (parallelization protocols). Most of the existing codes were developed when these possibilities did not exist, therefore, they continued in the tradition of old nuclear simulators and, in this sense, the GENUC's simulator makes use of the latest power computing resources.

The novelty of this procedure is due to the fact that, in models reported in the literature, the energy dependence of the cross sections is obtained from databases (GEANT [2], TRIPOLI [3], MCNP [4], Serpent [5], KENO [6], MCBEND [7], OpenMC [8] and others). On the other hand, in the proposed methodology, the cross sections [9] are obtained by continuous parameterizations between 0 MeV and 20 MeV , including the resolved and unresolved resonances, with a deviation of less than 1%, using the reference functions [10] , [11].

It should be noted that the computational paradigm adopted in this development is compatible with the GEANT platform, which in the future will allow the insertion of the simulator to be developed as a module in GEANT, which will be useful for simulating shields in the various applications in reactors, environments with neutron sources such as hospitals, baggage control at airports and others.

In the current operational state of the simulator, the materials were considered with a single global temperature, while in real situations the moderator is in a temperature range centered around 300ºC, and the fuel has a gradient from the center of the rod of approximately 2000ºC, to the surface at approximately 400°C. Details depend on model and specific fuel element geometry.

In this project, a particle simulator was developed with the purpose of translating simulations results into cross sections which are the basis for simulating interactions between neutrons and target materials. This project aimed to continue the development of a computational platform to simulate particles with emphasis on computational approaches based on a physical method. The simulations conducted and their results are useful in the matter of the studies of advanced reactors, future nuclear systems and issues related to nuclear fuel. The development of this

computational platform for particle simulations constitutes a tool both for theoretical studies for nuclear reactors and for applications. The project contributes by extending the code with new details. The present work, with its expected results, aimed to explore the possibilities of extending interaction descriptions, currently based merely on cross sections, also to approaches based on interaction potentials. It should be noted that there are a variety of simulators, some of them mentioned above, which, for the most part, follow established paradigms from the 1970s. As a difference in the project of creating a new code, one of the highlights is that the GENUC simulator is being developed using the latest advances (from 2011 onward) in high-performance computing. The present work aims to contribute in this sense.

The general objective is to contribute in the next steps of the development of the simulator, that is, in the extension of the simulator's functionality. The specific objective is to contribute to the exploration of the implementation of temperature effects due to thermal movement of particles.

The objectives of the project in which this activity fits are the development and application of the so-called enabling technologies that can solve operational problems and real demands of the Almirante Álvaro Alberto Nuclear Power Station (Central Nuclear Almirante Álvaro Alberto - CNAAA), Angra dos Reis, and analyze issues of strategic interest to Eletronuclear regarding the possible classes of reactors to be acquired or developed in the future in Brazil.

The specific objective of this undergraduation program project consists in the implementation of the thermal effects in the interactions between particles at any temperature.

The adopted philosophy (physical Monte Carlo) for the creation of the Monte Carlo simulator opens the possibility of implementing the effects of temperatures directly in the interactions. Even with a homogenized material medium it is possible to take into account local temperature effects. The thermal energies in the Maxwell-Boltzmann distribution, kinetic energy and also potentials that represent the vibration and rotation energies of molecules or chains of atoms in a solid must be taken into account.

1.1 MOTIVATION

An analysis of the interaction between particles can lead to a better understanding of how nuclear energy can be produced. Therefore, there is a need to study this problem with a view to future implementations in the GENUC's simulator.

Nuclear power is a low-carbon source of energy, because, unlike coal, oil or gas power plants, nuclear power plants practically do not produce CO_2 during their operation. Nuclear reactors generate close to one-third of the world’s carbon free electricity and are crucial in meeting climate change goals as identified by the International Atomic Energy Agency - IAEA (2022).

1,04% (1.99GW) of the power of the operational Brazilian electrical matrix comes from two thermonuclear power plants as identified by the Brazilian Electricity Regulatory Agency (Agência Nacional de Energia Elétrica - ANEEL) (2023).

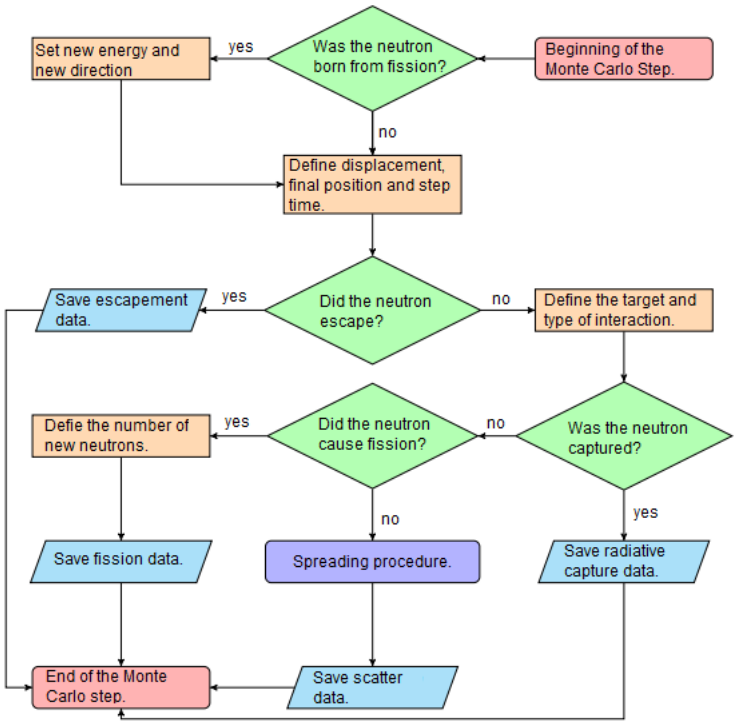
Motivated by the desire to increase the share of electricity production by nuclear reaction, it is necessary to develop knowledge in the area.

1.2 PRESENTATION OF THE PROBLEM

For cross-section measurements, sophisticated infrastructures are needed. In this context, simulators are essential since they provide a laboratory for all types of experiments. They are used since it is not possible to experiment in practice easily. On the other hand, simulators are easy to produce results without needing real measuring equipment. Therefore, to add more physical details to the simulation, the present proposal aims to allow considering any potentials and temperatures for each particle interactions and motion, respectively.

In measurements, for example the number of non-deflected beam particles, are determined for a specific kinetic energy fo the neutron and thermal motion due to a given temperature. In close analogy to experiments, the results obtained are discussed in the Case Study section with its analysis and discussion. The present contribution focusses on one operational component, which describes the scattering contribution to the set of interactoins and is connected to the described in Figure 1.

Figura 1 – Monte Carlo step flowchart



Source: Self elaboration (2023).

Each decision is made based on probabilities which relie on the value of the cross section. Most of the simulators use a database built from experiments to get these values. However, that takes time. The group's solutions is fitting a curve to this database. This procedure saves time. However, some reactions still do not have a database, since making the experiments is impractical or even impossible. Also, getting information from database cost time. The objective of this diploma project is to create a simulator that possibilitates to calculate the value of the cross sections for differents physics contexts. This values are going to be utilized by the group's simulator to determine the probabilities at each decision as described in Figure 1.

2 LITERATURE REVIEW

2.1 MONTE CARLO

The name Monte Carlo was applied to a class of mathematical methods first by scientists developing nuclear weapons at Los Alamos in the 1940s. The essence of the method is the invention of games of chance whose behavior and outcome can be used to study some interesting phenomena. Although there is no essential connection to computers, the effectiveness of numerical or simulated play as a serious scientific pursuit is greatly enhanced by the availability of modern digital computers as identified by Kalos and Whitlock (2004).

The Monte Carlo method is widely applied in several physics and engineering problems, such as radiation transport, statistical physics, quantum mechanics and nuclear theory, more specifically in neutron transport, context in which the present undergraduation program project is inserted.

There are two basic ways to approach the use of Monte Carlo methods to solve the transport equation: mathematical technique for numerical integration and computational simulation of a physical process. The mathematical approach is useful for importance sampling, convergence, variance reduction, random sampling techniques, eigenvalue calculation schemes, as identified by Brown (2016). The microscopic simulation approach is useful for collision physics, tracking, counting, and more. Monte Carlo methods solve integration problems. Most Monte Carlo theories deal with fixed source problems and using the concept of continuous distributions. Another type of problems that make use of distributions are the eigenvalues and are typically used for criticality calculations and reactor physics topics.

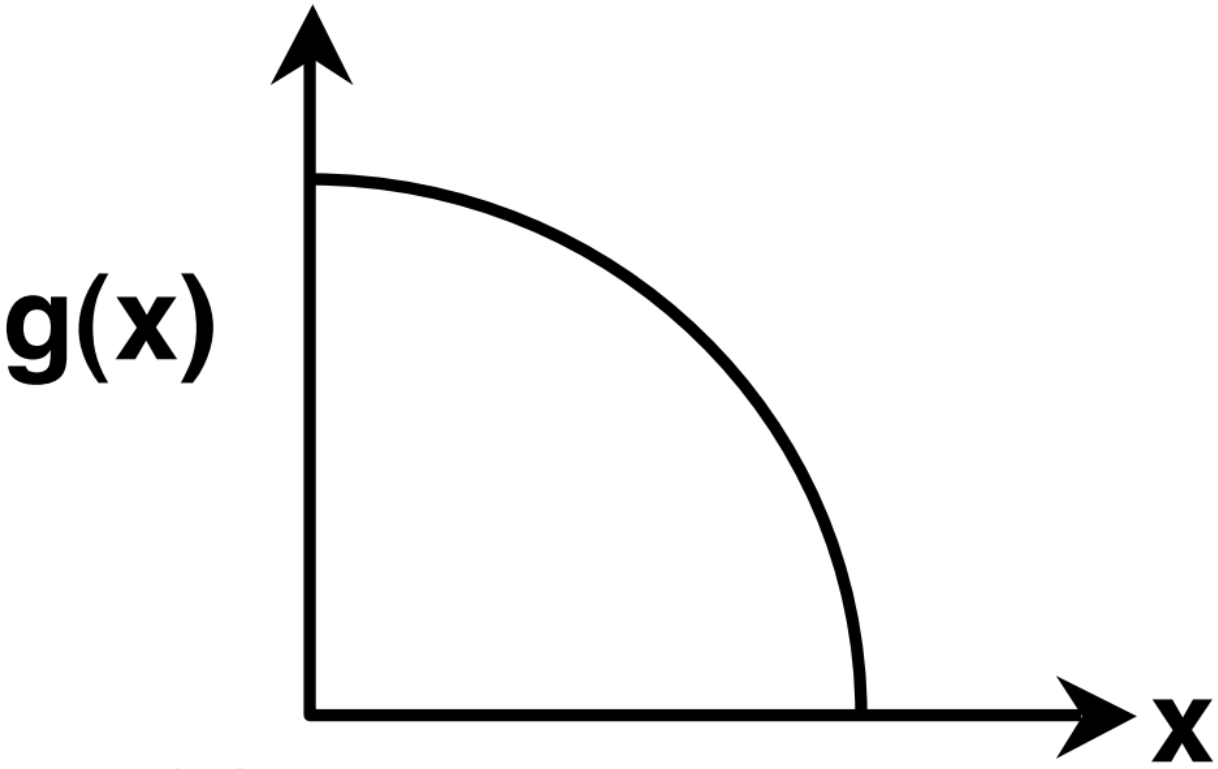
Simple Monte Carlo Example:

Evaluate Equation 1 with Equation 2, that represents the area under the curve in the Figure 2.

$$G = \int_0^1 g(x)dx \quad (1)$$

$$g(x) = \sqrt{1 - x^2} \quad (2)$$

Figura 2 – Monte Carlo example



Fonte: Brown (2016).

Mathematical Approach:

For $k = 1, \dots, N$: choose $\hat{x}_k$ randomly from $(0, 1)$

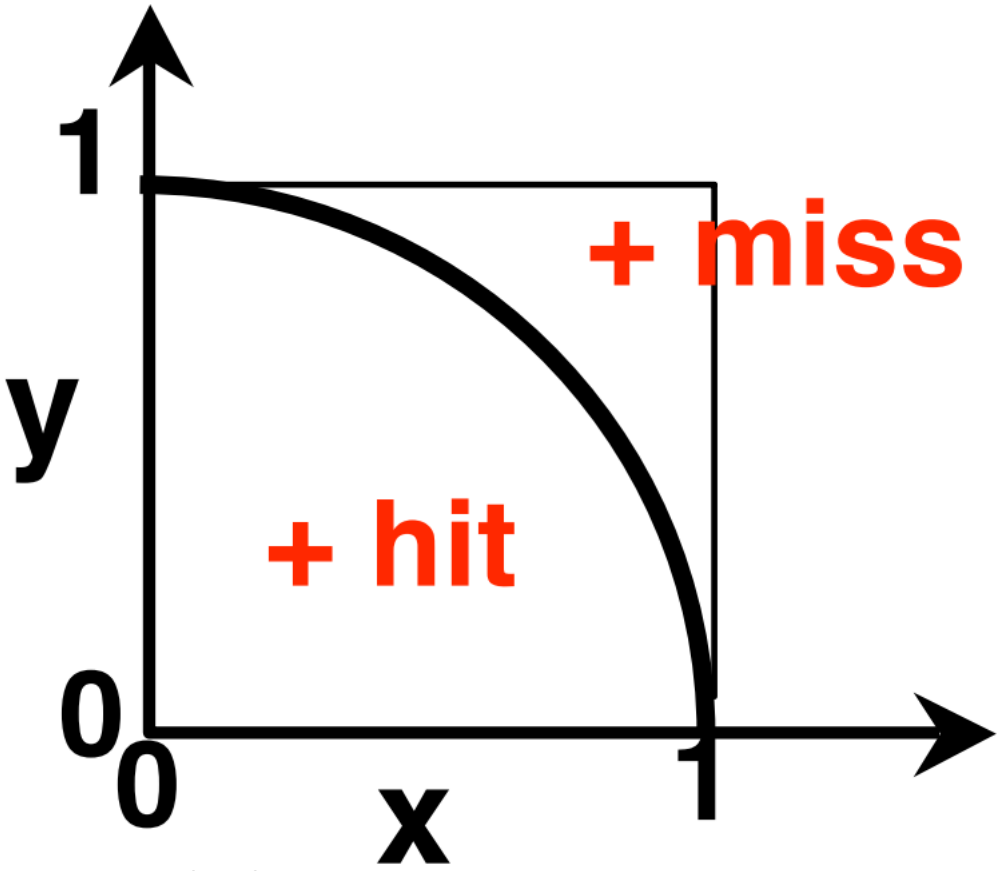
$$G = (1 - 0) \cdot [average\ value\ of\ g(x)] \approx \frac{1}{N} \cdot \sum_{k=1}^{N} g(\hat{x}_k) = \frac{1}{N} \cdot \sum_{k=1}^{N} \sqrt{1 - x_k^2}$$

Simulation approach:

“dart game”

For $k = 1, \dots, N$: choose $\hat{x}_k$, $\hat{y}_k$ randomly from $(0, 1)$, if $\hat{x}_k^2 + \hat{y}_k^2 \leq 1$, mark a “hit”.

Figura 3 – Simulation approach



Source: Brown (2016).

$$G = [area\ under\ the\ curve] \approx (1 \cdot 1) \cdot \frac{number\ of\ hits}{N}$$

Monte Carlo is often the method of choice for applications with integration in many dimensions. Examples: high energy physics, particle transport, financial analysis, risk analysis, process engineering, and others.

The mathematical approach of the Monte Carlo method involves distributions, therefore, reducing information to those focused on the specific analysis of interest. The physical approach of the Monte Carlo method uses the philosophy of simulation data generation and post analysis. First, particles are simulated with their propagations and interactions, data is generated, and then multiple analyzes are explored based on these generated data. The downside is that it needs more data storage, however, with advances in digital technology, this demand, nowadays, does not impose restrictions, on the contrary, new computational paradigms open the way for innovative approaches.

The mathematical Monte Carlo method was common to use, because there were no conditions, such as the parallelization that is necessary for physical Monte Carlo simulations due to its demand for a high sampling number.

2.2 COMPUTATIONAL CODES

The simulator developed in C++ language by GENUC at the School of Engineering (Escola de Engenharia) at UFRGS stands out. The description of the initial code developments are documented in de Camargo (2011). Subsequent advances and progress are documented in de Camargo et al. (2013), Barcellos et al. (2015, 2017b) and Barcellos et al. (2017a), with a shielding simulation presented in reference in Barcellos et al. (2019). This neutron transport simulator, which uses the physical Monte Carlo method, considers the cross sections as continuous functions in the energy spectrum, obtained from the linear interpolation of data available in the nuclear libraries, so that these functions are implemented in the simulator code, thus reducing the computational cost involved when compared to the others that determine the cross sections directly accessing databases.

A first contribution of this work to advance developments is to enrich the simulator highlighted above through the implementation in C of interactions, with temperature effects, between particles through collisions and potentials.

Computational codes for neutron transport found in the literature

Probabilistic codes, [3], [4], [5], [6], [7], [8]

1. COG - An LLNL Monte Carlo code for criticality safety analysis and general radiation transport (http://cog.llnl.gov).

2. OpenMC - A Monte Carlo code from MIT (open source).

3. MCNP - A LANL Monte Carlo code for general radiation transport.

4. RMC - A Monte Carlo code from Tsinghua University - Department of Engineering Physics for general radiation transport.

5. KENO/Shift - An ORNL Monte Carlo code for criticality analysis.

6. MCBEND - An ANSWERS Software Service Monte Carlo code for general radiation transport.

7. Serpent - A Finnish Monte Carlo code for neutron transport.

8. TRIPOLI – A 3D Monte Carlo and continuous energy code for general transport by CEA, France.

Serpent is the most similar to the GENUC's simulator, but it does not follow the 2014 parallelism protocols, considered the current standard. Processing time, physical space and memory space are no longer problems, so the pioneering approach of the group's simulator in this direction is possible. Serpent does not have as many features as the group's, and the group's can be used by other simulators as a module, Serpent does not, both in simulation and in parallelization.

The traditional approach of the mathematical Monte Carlo method solves the Boltzmann transport equation, while the physical Monte Carlo simulates the particles by trajectories and interaction vertices. The second approach, adopted in this work, is only possible through the computational resources currently available.

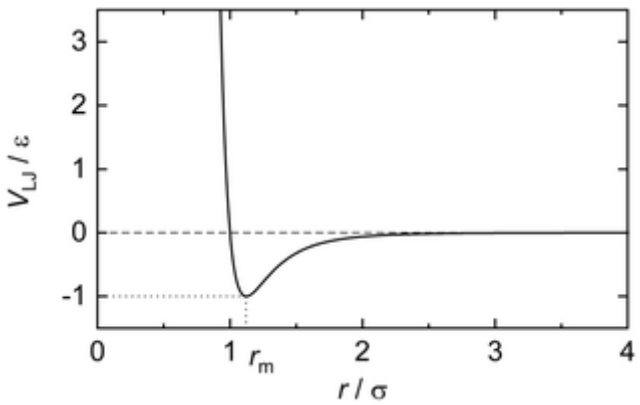
2.3 LENNARD-JONES POTENTIAL

Lennard-Jones potetential, named after the British mathematician John Edward Lennard-Jones, is a pair potential. The potential models repulsive and attractive interactions as described by Equation 3.

$$V(r) = 4\varepsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right] \quad 3$$

where r is the distance between two interacting particles, ε is the depth of the potential well and σ is the distance at which the potential is zero as described by Figure 4.

Figura 4 – Lennard-Jones potential



Source: Wikipedia (2023).

The Lennard-Jones potential has its minimum at $r = 2^{\frac{1}{6}}\sigma$.

2.4 CROSS SECTION

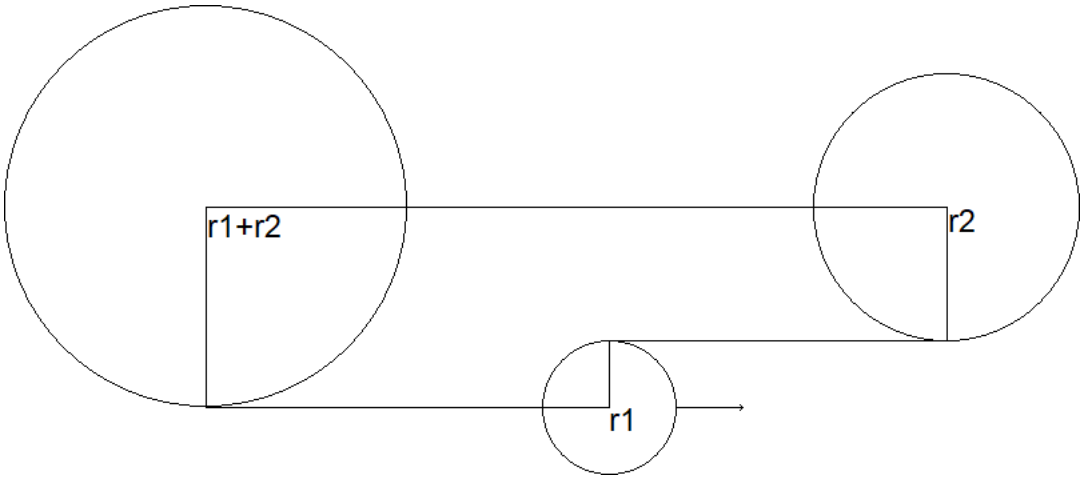
Physics of particle processes is dominated by quantum theories due to the microscopic scale and the characteristic of the strong interaction that is short ranged. Even then, it is instructive to analyze the classical collision process as an approach to studying reactions between particles.

For this purpose particle-particle interaction scenario are considered. In classic scenarios the collision happens through contact, therefore, both the projectile and the target must be associated with a size, in this case a radius, assuming that both have spherical geometry.

The “size” of the atom or nucleus is related to a quantity called the cross section, which depends on the specific reaction whether at the atomic or nuclear level. Some examples are scattering, capturing and others. To elaborate the concept of the cross session using phenomenological arguments, consider a jet of particles that extends homogeneously in an area A perpendicular to the direction of propagation, where each particle has a radius r_1 and the volumetric density of projectile particles is n_p. This particle jet impinges on a layer of target particles where the thickness is Δx, the target particle radius is r_2 and the target particle density is n_a. To quantify the interaction between the projectile particles and respectively the target, the number of collisions and the deviation of the trajectories in relation to the initial direction, that is, those that suffer interaction contain this information.

The design of the cross section can be elaborated using a purely geometrical analogy. A collision between two particles with radii r_1 and r_2 respectively happens when the center of mass of the projectile is in an area centered on the center of mass of the target particle with effective area $\sigma_g = \pi(r_1 + r_2)^2$. Note, that in this way the property (geometric extension) of the projectile and target are united in the geometric (microscopic) cross section σ_g.

Figura 5 – Cross section σ



Source: Self elaboration (2023).

Assuming that the thickness of the layer Δx is thin enough that there is no overlapping of the areas σ_g then the probability for a collision to occur is proportional to and can be calculated by summing all the areas σ_g contained in the volume $A\Delta x$ (i.e. number of target particles N_a) in this volume in relation to the area (particle beam extension).

$$p = \frac{N_a \sigma_g}{A} \tag{4}$$

The reconstruction for arbitrary thickness x can be determined by composing x by a number of layers with thickness Δx. Using the probabilistic argument then the part of the particles that undergo collision ΔN and therefore are deviated from the initial direction is given by

$$\Delta N = -pN = -\frac{N_a \sigma_g N}{A} = -\frac{n_a \Delta x A \sigma_g N}{A} = -n_a \Delta x \sigma_g N \tag{5}$$

where the minus sign indicates the decrease in the intensity of the initial jet and this quantity is independent of the extension of the particle jet A. In the infinitesimal limit for the thickness Δx we obtain

$$\frac{dN}{dx} = -n_a \sigma_g N \tag{6}$$

with the solution

$$N = N_0 e^{-n_a \sigma_g x} = N_0 e^{-\Sigma x} \tag{7}$$

where $\sum$ is the so-called macroscopic cross section (more appropriate would be the interaction coefficient, here the scattering) which indicates the number of collisions that a projectile particle undergoes per unit length traveled in the medium composed of the target particles. On the other hand, the number of deviated particles is $N_0\left(1 - e^{-\Sigma x}\right)$.

In the presented considerations, only a difference between collision-deflected and non-colliding particles was made, so the (microscopic) cross section represents a total cross section.

3 CASE STUDY

3.1 ADOPTED METHODOLOGY

Regarding the methodology used, to carry out the simulations, a computer with a 64-bit processor, frequency of 1.8 GHz, 2 cores (4 threads), 4GB of RAM memory, operating system Windows 10, 64 bits was used. Noted that, in order to carry out the analyses, there is no need for high-performance computational resources, which becomes essential after the implementation of what was developed and validated in this diploma work.

Potentials and their effects on the nuclear system was carried out with a stand-alone program to evaluate the effects of interest before extending the group's simulator code under development. The software necessary to carry out the simulations was developed and implemented by the author of the work. The code was written in the ANSI C language and the compiler needed to generate the executable was gcc (GNU Free Software License). To edit the source code, the free platform CodeBlocks release 20.03 was used. The graph of the cross section as function of the temperature was made by the author in a Python program.

First, all the libraries, constants and parameters are defined. Since the objective is to build a laboratory with the simulator, the program is structured so it can be easily manipulated, activating or deactivating different modules such as gravity, Coulomb and Lennard-Jones potencials that have been implemented or different executions of collision. Many variations are easily achievable by changing few lines of the code. The screen coordinates ranges from -1 to 1 both on the horizontal axis and the vertical axis. The positions of the target particles are randomly choosen between the values of the wall. All beam particles are allocated at the most left, with horizontal values of -1 and the vertical position are randomly choosen.

The problem has the characteristic of a system of many poly energetic particles in motion but with interaction only between pairs. For an approach there are two possibilities (a) continuum via distributional quantities (densities, scalar and angular fluxes, etc.) or (b) considering particles. Not all distributions are known so these must be generated. Two possibilities: (a) starting with a distribution and deforming it according to criteria (acceptance and rejection in mathematical Monte Carlo); (b) accumulating trajectories and interactions of particles according to their microscopic characteristics (sampling higher than (a) but need not accept or reject

events.). Physical Monte Carlo is more generic, simulations are performed generating a set of data, post-analysis to generate distributions from the data that can be parameterized to obtain approximate representations in analytical form.

After all being set up, then the implementation is, roughly speaking, tracking and interaction. For each particle, it is calculated, at each iteration, the resulting acceleration that acts on the particle at a position in relation to the other particles and the potentials under analysis as described by Equation 8. Note: Throughout the document, m is mass and v is velocity.

$$a = \frac{F}{m} = \frac{-\nabla V}{m} \tag{8}$$

The procedure adopted for the two-dimensional problem is separated for each collision into two one-dimensional problems, normal and tangential to the surface, and only the normal undergoes changes, so a one-dimensional formalism is sufficient.

An elastic collision is a collision in which kinetic energy is conserved. This means that there is no energy lost as heat, sound during collision, etc. In an elastic collision, both kinetic energy and momentum are conserved (the total before and after the collision remains the same).

Momentum is the product of mass and velocity:

$$p = m \cdot v \tag{9}$$

The kinetic energy of an object is one-half times its mass times the square of its velocity:

$$KE = \frac{1}{2}mv^2 \tag{10}$$

Subscripts 1 and 2 refer to one of the two colliding objects. An apostrophe after the variable means that the value is obtained after the collision.

We write the conservation of momentum and kinetic energy as two equations:

Moment conservation:

$$m_1 v_1 + m_2 v_2 = m_1 v_1' + m_2 v_2' \tag{11}$$

Conservation of kinetic energy:

$$\frac{1}{2}m_1v_1^2+\frac{1}{2}m_2v_2^2=\frac{1}{2}m_1{v_1'}^2+\frac{1}{2}m_2{v_2'}^2 \quad 12$$

Combining these two equations results in the final velocities (after the collision) of objects 1 and 2:

$$v_1'=\frac{v_1(m_1-m_2)+2m_2v_2}{m_1+m_2} \quad 13$$

$$v_2'=\frac{v_2(m_2-m_1)+2m_1v_1}{m_1+m_2} \quad 14$$

This result allows one to find the velocity of two objects after experiencing a one-dimensional elastic collision. This result is used for the two-dimensional case.

A 7-step process follows to find the new velocities of two objects after collision. The basic objective of the process is to project the velocity vectors of the two objects onto vectors that are normal (perpendicular) and tangent to the collision surface. This gives a normal component and a tangential component for each velocity. The tangential components of the velocities are not modified by the collision, because there is no force along the tangent line to the collision surface. The normal components of the velocities undergo a one-dimensional collision, which can be computed using the one-dimensional collision formulas presented above. Then the unit normal vector is multiplied by the normal speed after the collision to obtain a vector that has a direction normal to the collision surface and a magnitude that is the normal component of the velocity after the collision. The same is done with the unit tangential vector and the tangential component of velocity. Finally, the new velocity vectors are found by adding the normal velocity and tangential velocity vectors for each object.

1. Find the unit normal and unit tangential vectors. The unit normal vector is a vector which has a magnitude of 1 and a direction that is normal (perpendicular) to the surface of the particles at the point of collision. The unit tangential vector is a vector with magnitude 1 which is tangent to the paticle surfaces at the point of collision. First, find the normal vector. This is done by taking a vector whose components are the difference between the coordinates of the centers of each object. Let x_1, x_2, y_1 and y_2 be the x and y coordinates of the particles' centers. The normal vector $\vec{n}$ is:

$$\vec{n} = (x_2 - x_1, y_2 - y_1)^T \quad 15$$

Where T is the transpose matrix. Next, find the unit vector of $\vec{n}$, which we will call $\overrightarrow{un}$. This is done by dividing $\vec{n}$ by its magnitude:

$$\overrightarrow{un} = \frac{\vec{n}}{|\vec{n}|} = \frac{\vec{n}}{\sqrt{n_x^2 + n_y^2}} \quad 16$$

Next, the unit tangential vector is needed. It is found from the unit normal vector:

$$\overrightarrow{ut} = \left(-un_y, un_x\right)^T \quad 17$$

2. Create the initial velocity vectors (before the collision) $\vec{v}_1$ and $\vec{v}_2$. These are just the x and y components of the velocities put into vectors:

$$\vec{v}_1 = \left(v_{1x}, v_{1y}\right)^T \quad 18$$

$$\vec{v}_2 = \left(v_{2x}, v_{2y}\right)^T \quad 19$$

Note that this step is not necessary if the velocities are already represented as vectors, which is the case of the program. The positions, velocities and acelerations are created in the vector format. This step assumes that the velocities are initially represented as x and y components.

3. After the collision, the tangential component of the velocities is unchanged and the normal component of the velocities can be found using the one-dimensional collision formulas given above. It is necessary to solve the velocity vectors $\vec{v}_1$ and $\vec{v}_2$ into normal and tangential components. To do this, project the velocity vectors onto the unit normal and unit tangential vectors by taking the dot product of the velocity vectors and the unit normal and unit tangential vectors. Let v_{1n} be the speed (simple number, not a vector) of object 1 in the normal direction. Let v_{1t} be the speed (simple number, not a vector) of object 1 in the tangential direction. Let v_{2n} be the speed (simple number, not a vector) of object 2 in the normal direction. Let v_{2t} be the speed (simple number, not a vector) of object 2 in the tangential direction. These values are found by projecting the velocity vectors onto the unit normal and unit tangential vectors, which is done by taking the dot product:

$$v_{1n} = \overrightarrow{un} \cdot \vec{v}_1 \quad (20)$$

$$v_{1t} = \overrightarrow{ut} \cdot \vec{v}_1 \quad (21)$$

$$v_{2n} = \overrightarrow{un} \cdot \vec{v}_2 \quad (22)$$

$$v_{2t} = \overrightarrow{ut} \cdot \vec{v}_2 \quad (23)$$

4. The new tangential velocities (after the collision) are found. The tangential components of velocity do not change after the collision, because there is no force between the objects in the tangential direction during the collision. So the tangential velocities are simply the same as the old ones:

$$v'_{1t} = v_{1t} \quad (24)$$

$$v'_{2t} = v_{2t} \quad (25)$$

Remember that the apostrophe after the variable means “after the collision”.

5. The new normal speeds are found. This is where one-dimensional collision formulas are used. The velocities of the two objects along the normal direction are perpendicular to the surfaces of the objects at the point of collision, so this really is a one-dimensional collision.

$$v'_{1n} = \frac{v_{1n}(m_1 - m_2) + 2m_2 v_{2n}}{m_1 + m_2} \quad (26)$$

$$v'_{2n} = \frac{v_{2n}(m_2 - m_1) + 2m_1 v_{1n}}{m_1 + m_2} \quad (27)$$

6. Convert the scalar normal and tangential velocities into vectors. Multiply the unit normal vector by the normal scalar velocity and you get a vector that has a direction that is normal to the surface at the point of collision and that has a magnitude equal to the normal component of the velocity. It is simulate for the tangential component.

$$\vec{v}'_{1n} = v'_{1n} \cdot \overrightarrow{un} \quad (28)$$

$$\vec{v}'_{1t} = v'_{1t} \cdot \overrightarrow{ut} \quad (29)$$

$$\vec{v}'_{2n} = v'_{2n} \cdot \overrightarrow{un} \quad (30)$$

$$\vec{v}'_{2t} = v'_{2t} \cdot \overrightarrow{ut} \quad (31)$$

7. Find the final velocity vectors by adding the normal and tangential components for each object:

$$\vec{v}_1' = \vec{v}_{1n}' + \vec{v}_{1t}' \quad (32)$$

$$\vec{v}_2' = \vec{v}_{2n}' + \vec{v}_{2t}' \quad (33)$$

Now, the final velocity (after the collision) for each object as vectors have been calculated. Since the mass of the target particles are way bigger than the mass of the beam particles, it is a good aproximation to set the values of velocity back to zero. This way, the target particles will stay fixed.

The velocity and position of each particle, at each iteration, are updated according to, respectively, its acceleration and velocity as described by the Equation 34 and Equation 35. An apostrophe after the variable means the value is taken after the update.

$$v' = v + at \quad (34)$$

$$s' = s + vt \quad (35)$$

The particles, if there are no potentials, move freely until the collision, classically it is purely geometric, it depends on the size of the particles. But in physical scenarios with potentials, particles are determined to interact in terms of a force or potential. The movement is only through the center of mass, but a size is chosen for each particle, necessary to deal with collisions.

Classical treatment is adequate in view of the short range of strong nuclear interaction. Particles interact isotropically, that is, they do not have directions with different interaction intensities. There are effective potentials that represent the nuclear interaction with the characteristics: long-range attraction, short-range repulsion.

Unlike other simulators that use the projectile, the target or both, through distributional quantities, the hypothesis of this project is that both the projectile and the target are used discretly. In the classical hypothesis, there is interaction only on contact so that collision partners do not penetrate each other, with an infinite contact repulsive force. When one puts more physics and an interaction in terms of physical interaction, at that moment this is no longer the case, because there is an interaction at a distance, the particles no longer need physical contact to perceive the presence of the others. The distance that one particle passes from the other will define the deviation in the trajectory of both. There is no closed theory in the nuclear area, where closed theory means, for example, Maxwell's equations of electromagnetism

that unequivocally determine the potential, so experiments with different types of potentials are needed.

The potential gradients define the interactions (forces) that have intensities dependent on the distance of the participating particles in the collision. For simplicity, the problem is treated in two dimensions. Interactions at an instant are considered only between pairs of particles, i.e. simultaneous forces between three particles are disregarded. The resultant force at each particle is the superposition of the forces between the pairs. To determine the movements, the laws of conservation of energy and momentum are used.

Although everything is being implemented out with only two dimensions, new ones are simple to implement since it is just another columns. Actually, v[particle_number][0], which represents the velocity in the horizontal x dimension, v[particle_number][1], which represents the velocity in the vertical y dimension and v[particle_number][0], which represents the velocity in the depth z dimension, are all being calculated. Similarly, for other vector involved such as positions and acelerations. On the other hand, for a better view, the z dimension have initial parameters so that it always stays zero. However, it would be easy to manipulate a third, forth, fitfh, or even more dimensions, since it is already preperad to do so.

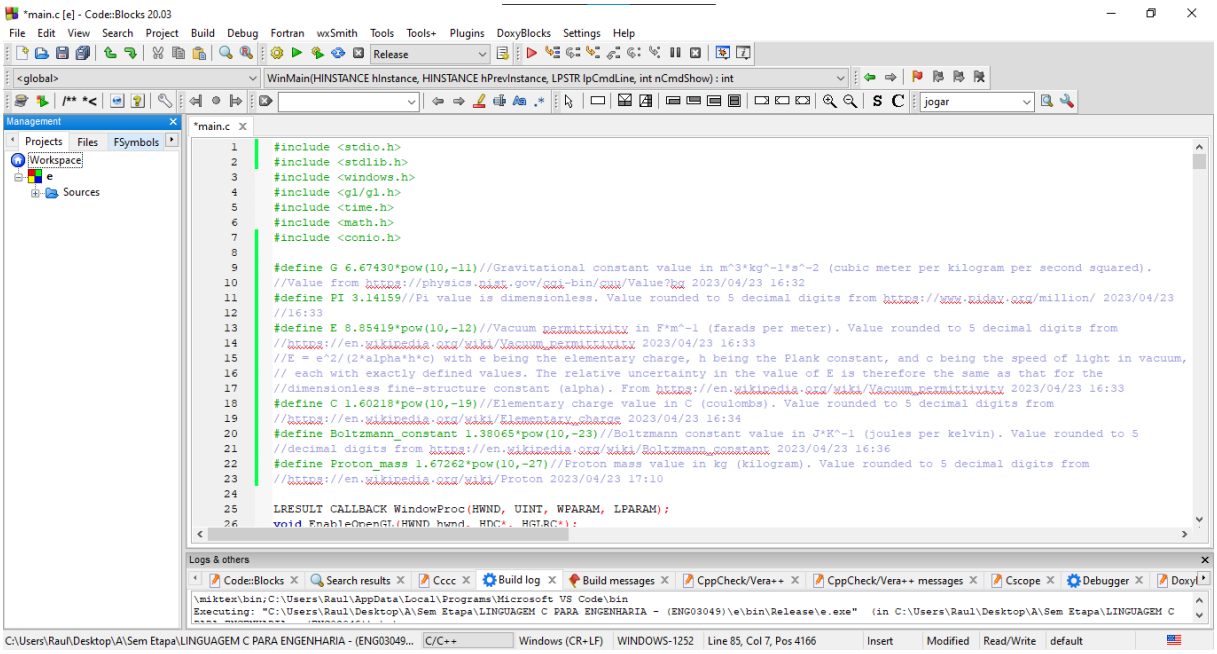
Every beam particle that pass through the sensor area is marked as a non-deflected particle and acounted to the total number of particles sensed by the detector. The simulation is terminated when all beam particles extrapolate the screen coordinates. At this moment, the “sensor.txt” file is created and the important informations are written. The file is closed, OpenGL is shut down and the window is explicitly destroyed.

The simulations have been rerunned for different temperatures with the beam particles with fixed velocitites.

3.2 RESULTS OBTAINED

Figure 7 show the CodeBlocks ambient and beginning of the code.

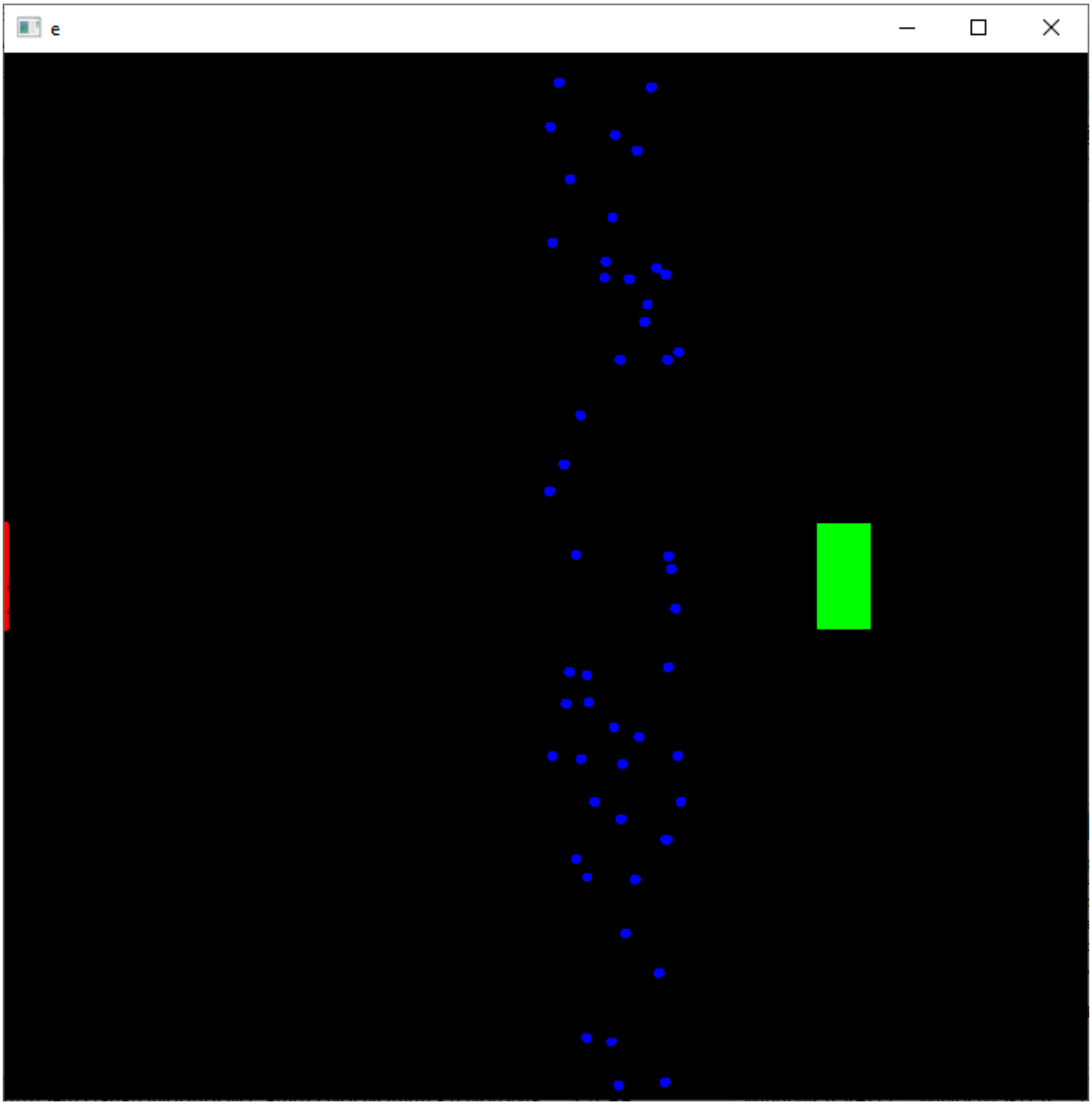
Figure 6 – Code ambient



Fonte: Self elaboration (2023).

Runing the code, the first thing generated are the paricles as shown in Figure 8.

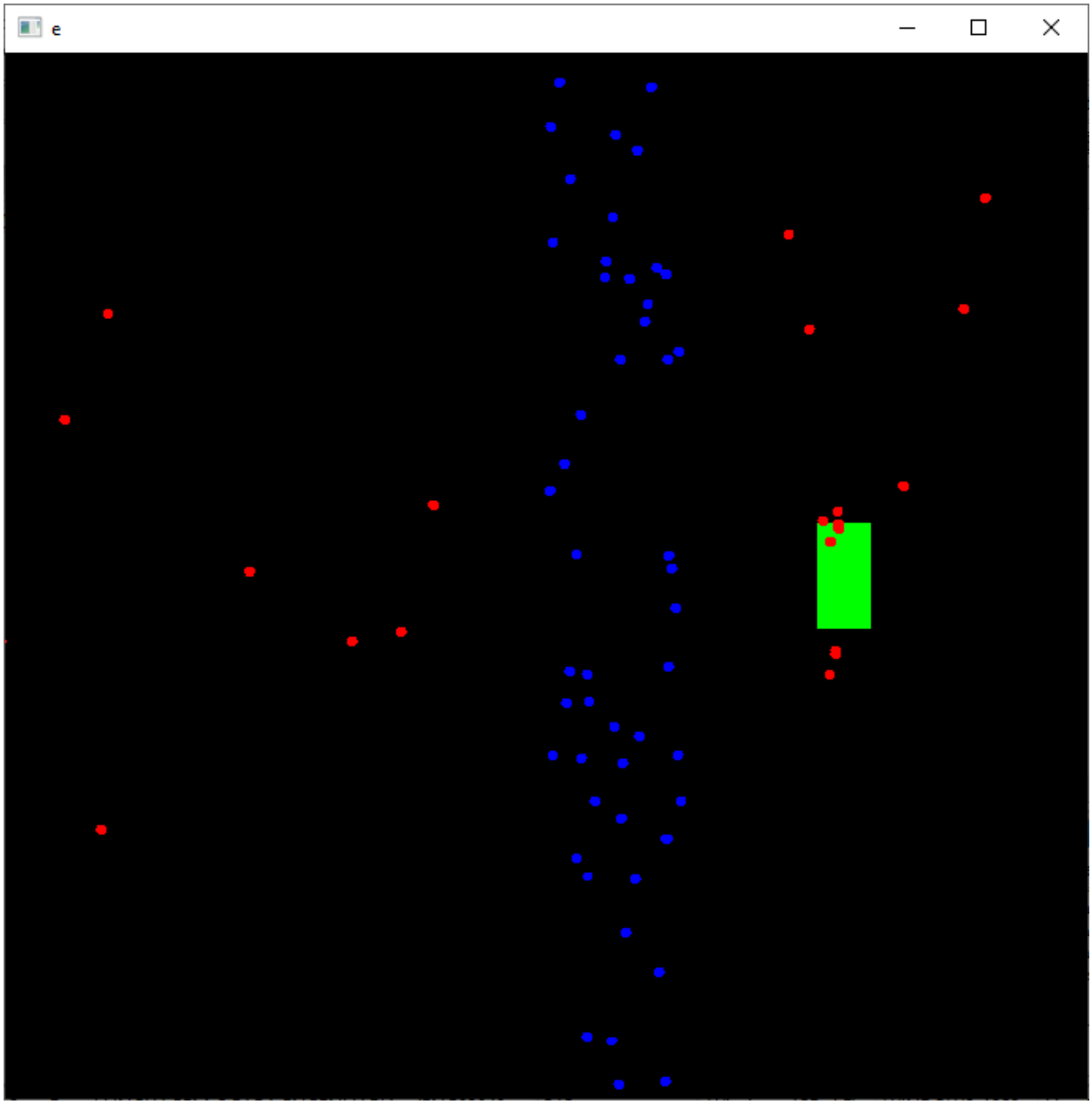
Figure 7 – Initial positions



Fonte: Self elaboration (2023).

The particles in blue are the target particles, the red particles are the beam particles and the square in green represents the detector. As the program continues, the particles move and interact with each other. Beam particles do not interact with each other, same for the target particles. Figure 9 show the program in a later moment.

Figure 8 – Particle positions



Fonte: Self elaboration (2023).

By the end of the execution, the “sensor.txt” file have been written.

Figure 9 – File written

sensor.txt - Bloco de Notas

Arquivo Editar Formatar Exibir Ajuda

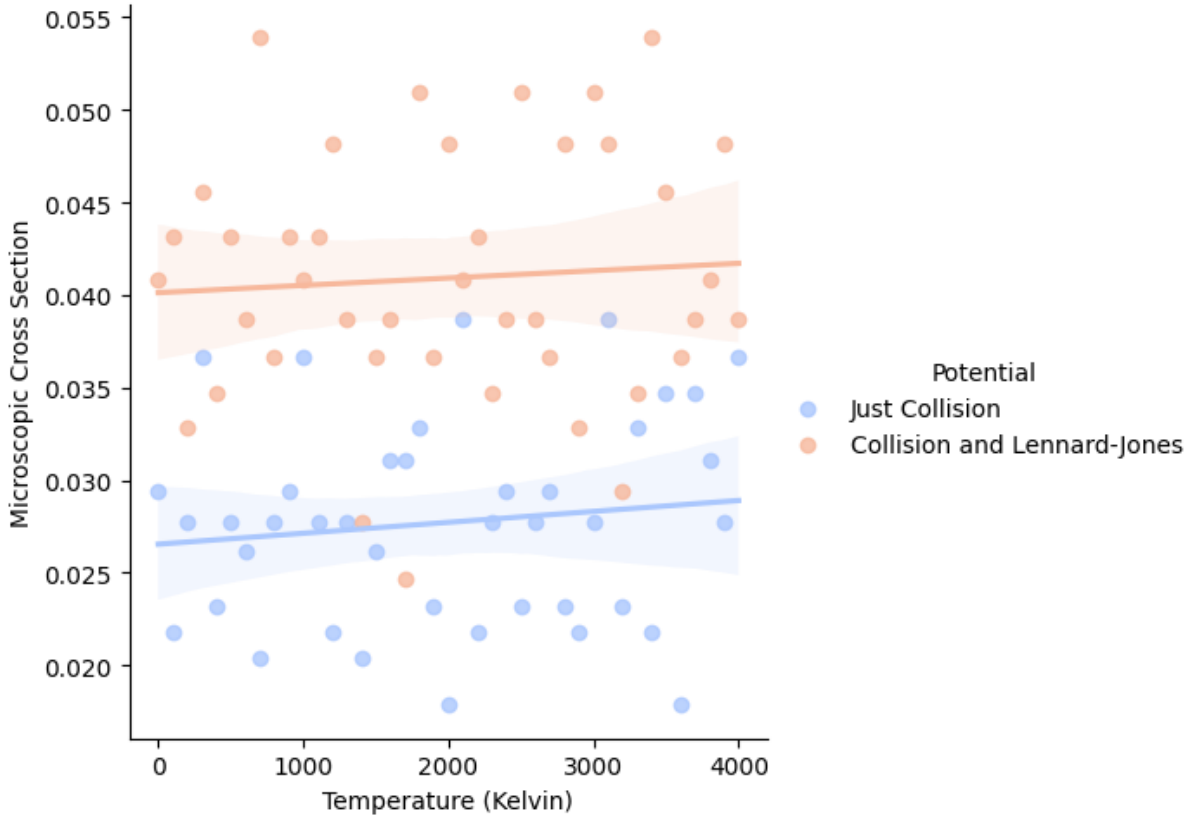
```
Run number: 1
Seed: 371
Temperature (Kelvins): 0.000000
Total number of simulated particles: 100
Number of target particles: 50
Number of beam particles: 50
Number of particles detected: 16
Number of deflected particles: 34
Macro Cross Section: 4.557737
Micro Cross Section: 0.045577

Run number: 2
Seed: 371
Temperature (Kelvins): 100.000000
Total number of simulated particles: 100
Number of target particles: 50
Number of beam particles: 50
Number of particles detected: 17
Number of deflected particles: 33
Macro Cross Section: 4.315239
Micro Cross Section: 0.043152

Run number: 3
Seed: 371
Temperature (Kelvins): 200.000000
Total number of simulated particles: 100
Number of target particles: 50
Number of beam particles: 50
Number of particles detected: 15
Number of deflected particles: 35
Macro Cross Section: 4.815891
Micro Cross Section: 0.048159

Run number: 4
```

Ln 1, Col 1 100% Windows (CRLF) UTF-8

Fonte: Self elaboration (2023).

The information is plotted as described in Figure 11.

Figure 10 – Regression



Fonte: Self elaboration (2023).

The values are distinguished between the experiments with collision and Lennard-Jones potential and the experiments with just the collision module activated.

3.3 CORRECTIONS

Some particles undergo multiple scattering before passing through the detector and are counted just the same as the particles that go straight to the detector. Particles that undergo some scattering but still pass through the detector are also counted. Although this does not interfere in the results, the author recognizes that the code is not optimized and could be more organized.

For future works, it is needed to see what is the important physics that must be taken into account when creating the simulation environment to determine the cross sections. For example, in certain contexts the collisions occurs with phonos rather than the target nuclei. But once the tool have been created, it is possible to apply it to get a final cross section. Defining the potential, and once the cross section have been simulated, which is beam particles' energy and target particles' temperature dependent, it is possible to use them in the GENUC's simulator since the cross sections values are used in the probabilities of each decision as described in Figure 1.

The tool created make way for various other reactions, vibrations, rotations, encoded in potentials. Now that the tool have been created by the author, once the physics context have been defined, there exists a tool ready to calculate the cross section.

Future works can explore more details, more potentials, not only scalars, since with the energies in the reactor the most suitable potential is also a tensor. Also the Maxwell-Boltzmann distribuition shall be implemented instead of using only the statistical mean.

4 CONCLUSION

The problem of calculating the cross section given the physics context, as described in the introduction, have been successfully accomplished. The values of the macroscopic and microscopic cross section are beeing calculated based on the physics scenario that have been set up from Equation 7.

Consistency is observed, what is expected was precisely the difference between with and without the potential. As the temperature increase, the microscopic cross section increase, as expected. The width of the cross section distribution should increase with increasing temperature. The temperature influences are more expressive when the particle velocities of the beam are close to the thermal velocities of the target particles.

Some reactions still do not have a database, because making the experiments is impractical, or even impossible. Also, getting information from database cost time. That is why the simulations exist, to calculate rapidly this and others values independtly of experiments in the real world.

REFERÊNCIAS

[1] FEDERATIVE REPUBLIC OF BRAZIL. Convention on nuclear safety. Ninth national report for the joint eighth and ninth review meeting in march 2023. August 2022.

[2] S. Agostinelli, J. Allison, K. Amako, J. Apostolakis, H. Araujo, P. Arce, M. Asai, D. Axen, S. Banerjee, G. Barrand, F. Behner, L. Bellagamba, J. Boudreau, L. Broglia, A. Brunengo, H. Burkhardt, S. Chauvie, J. Chuma, R. Chytracek, G. Cooperman, G. Cosmo, P. Degtyarenko, A. Dell'Acqua, G. Depaola, D. Dietrich, R. Enami, A. Feliciello, C. Ferguson, H. Fesefeldt, G. Folger, F. Foppiano, A. Forti, S. Garelli, S. Giani, R. Giannitrapani, D. Gibin, J.J. Gómez Cadenas, I. González, G. Gracia Abril, G. Greeniaus, W. Greiner, V. Grichine, A. Grossheim, S. Guatelli, P. Gumplinger, R. Hamatsu, K. Hashimoto, H. Hasui, A. Heikkinen, A. Howard, V. Ivanchenko, A. Johnson, F.W. Jones, J. Kallenbach, N. Kanaya, M. Kawabata, Y. Kawabata, M. Kawaguti, S. Kelner, P. Kent, A. Kimura, T. Kodama, R. Kokoulin, M. Kossov, H. Kurashige, E. Lamanna, T. Lampén, V. Lara, V. Lefebure, F. Lei, M. Liendl, W. Lockman, F. Longo, S. Magni, M. Maire, E. Medernach, K. Minamimoto, P. Mora de Freitas, Y. Morita, K. Murakami, M. Nagamatu, R. Nartallo, P. Nieminen, T. Nishimura, K. Ohtsubo, M. Okamura, S. O'Neale, Y. Oohata, K. Paech, J. Perl, A. Pfeiffer, M.G. Pia, F. Ranjard, A. Rybin, S. Sadilov, E. Di Salvo, G. Santin, T. Sasaki, N. Savvas, Y. Sawada, S. Scherer, S. Sei, V. Sirotenko, D. Smith, N. Starkov, H. Stoecker, J. Sulkimo, M. Takahata, S. Tanaka, E. Tcherniaev, E. Safai Tehrani, M. Tropeano, P. Truscott, H. Uno, L. Urban, P. Urban, M. Verderi, A. Walkden, W. Wander, H. Weber, J.P. Wellisch, T. Wenaus, D.C. Williams, D. Wright, T. Yamada, H. Yoshida, and D. Zschiesche. Geant4—a simulation toolkit. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 506(3):250 – 303, 2003.

[3] E. Brun, F. Damian, C.M. Diop, E. Dumonteil, F.X. Hugot, C. Jouanne, Y.K. Lee, F. Malvagi, A. Mazzolo, O. Petit, J.C. Trama, T. Visonneau, and A. Zoia. Tripoli-4, cea, {EDF} and {AREVA} reference monte carlo code. Annals of Nuclear Energy, 82:151 – 160, 2015. Joint International Conference on Supercomputing in Nuclear Applications and Monte Carlo 2013, {SNA} + {MC} 2013. Pluri- and Trans-disciplinarity, Towards New Modeling and Numerical Simulation Paradigms.

[4] Jeho Jeong, Nathan E. White, and Sudarshan K. Loyalka. Three-dimensional transport theory: Evaluation of analytical expressions of williams and verification of {MCNP}. Annals of Nuclear Energy, 86:80 – 87, 2015. Tribute to a {GIANT}.

[5] Jaakko Leppänen, Maria Pusa, Tuomas Viitanen, Ville Valtavirta, and Toni Kaltiaisenaho. The serpent monte carlo code: Status, development and applications in 2013. Annals of Nuclear Energy, 82:142 – 150, 2015. Joint International Conference on Supercomputing in Nuclear Applications and Monte Carlo 2013, SNA + MC 2013. Pluri- and Trans-disciplinarity, Towards New Modeling and Numerical Simulatio Paradigms.

[6] B.T. Rearden, L.M. Petrie, D.E. Peplow, K.B. Bekar, D. Wiarda, C. Celik, C.M. Perfetti, A.M. Ibrahim, S.W.D. Hart, M.E. Dunn, and W.J. Marshall. Monte carlo capabilities of the {SCALE} code system. Annals of Nuclear Energy, 82:130 – 141, 2015. Joint International Conference on Supercomputing in Nuclear Applications and

Monte Carlo 2013, {SNA} + {MC} 2013. Pluri- and Trans-disciplinarity, Towards New Modeling and Numerical Simulation Paradigms.

[7] Simon D. Richards, Chris M.J. Baker, Adam J. Bird, Pat Cowan, Nigel Davies, Geoff P. Dobson, Tim C. Fry, Albrecht Kyrieleis, and Paul N. Smith. {MONK} and mcbend: Current status and recent developments. Annals of Nuclear Energy, 82:63 – 73, 2015. Joint International Conference on Supercomputing in Nuclear Applications and Monte Carlo 2013, {SNA} + {MC} 2013. Pluri- and Trans-disciplinarity, Towards New Modeling and Numerical Simulation Paradigms.

[8] Paul K. Romano, Nicholas E. Horelik, Bryan R. Herman, Adam G. Nelson, Benoit Forget, and Kord Smith. Openmc: A state-of-the-art monte carlo code for research and development. Annals of Nuclear Energy, 82:90 – 97, 2015. Joint International Conference on Supercomputing in Nuclear Applications and Monte Carlo 2013, {SNA} + {MC} 2013. Pluri- and Trans-disciplinarity, Towards New Modeling and Numerical Simulation Paradigms.

[9] M.B. Chadwick, M. Herman, P. Obložinský, M.E. Dunn, Y. Danon, A.C. Kahler, D.L. Smith, B. Pritychenko, G. Arbanas, R. Arcilla, R. Brewer, D.A. Brown, R. Capote, A.D. Carlson, Y.S. Cho, H. Derrien, K. Guber, G.M. Hale, S. Hoblit, S. Holloway, T.D. Johnson, T. Kawano, B.C. Kiedrowski, H. Kim, S. Kunieda, N.M. Larson, L. Leal, J.P. Lestone, R.C. Little, E.A. McCutchan, R.E. MacFarlane, M. MacInnes, C.M. Mattoon, R.D. McKnight, S.F. Mughabghab, G.P.A. Nobre, G. Palmiotti, A. Palumbo, M.T. Pigni, V.G. Pronyaev, R.O. Sayer, A.A. Sonzogni, N.C. Summers, P. Talou, I.J. Thompson, A. Trkov, R.L. Vogt, S.C. van der Marck, A. Wallner, M.C. White, D. Wiarda, and P.G. Young. Endf/b-vii.1 nuclear data for science and technology: Cross sections, covariances, fission product yields and decay data. Nuclear Data Sheets, 112(12):2887 – 2996, 2011.

[10] Dayana Q. de Camargo. Um modelo estocástico de simulação neutrônica considerando o espectro e propriedades nucleares com dependência contínua de energia. PhD thesis, Universidade Federal do Rio Grande do Sul. Escola de Engenharia. Programa de Pós-Graduação em Engenharia Mecânica, Porto Alegre, RS, Brasil, 2011.

[11] Dayana Q. de Camargo, Bardo E.J. Bodmann, Marco T. de Vilhena, Sergio de Queiroz Bogado Leite, and Antonio Carlos Marques Alvim. A stochastic model for neutrons simulation considering the spectrum and nuclear properties with continuous dependence of energy. Progress in Nuclear Energy, 69:59 – 63, 2013. Research & Developments in Advanced Nuclear Reactors.

[12] 2-Dimensional Elastic Collisions Without Trigonometry. 2006 Chad Berchek.

[13] INTERNATIONAL ATOMIC ENERGY AGENCY.