Neutron Diffusion

Riccardo Milioli

Contents

1 Introduction

With this presentation we want to study the diffusion of neutrons in fissile material, with different underlying geometries, where collisions between free neutrons and nuclei result in the release of secondary neutrons. As the fissile material increases in size the radioactive material will become critical when the total density of neutrons increases exponentially. The result is a chain reaction that can lead to an intense explosion. Our

goal is to find the size at which criticality occurs with calculations based on Dirichlet boundary conditions (meaning that the neutron density falls to zero at the edges, so neutrons don't escape). This is a simplified model with respect to the case of the more physical Neumann boundary conditions, which allow for neutron escape. For the $3D$ case in spherical coordinates, however, we we will perform the analysis with both boundary conditions.
First of all we need to focus on the diffusion equation, which is described by:

$$\frac{\partial n}{\partial t} = \mu \nabla^2 n. \tag{1.1}$$

Where μ is the diffusion constant and n the neutron density.
For the case of our interest the fissile material acts as a source of neutrons, so we need to include a source term. The equation (1.1) now becomes:

$$\frac{\partial n}{\partial t} = \mu \nabla^2 n + \eta n. \tag{1.2}$$

Where η is the neutron rate of formation.
Let's assume to be in $1D$ for the moment for simplicity. We solve this equation via the variable separable method, that is, we assume to be able to factorize n as:

$$n = T(t)X(x). \tag{1.3}$$

This greatly simplifies the problem that, from 1 PDE, is reduced to 2 (or more in higher dimensions) ODEs:

$$\frac{1}{T}\frac{\partial T}{\partial t} - \eta = \frac{\mu}{X}\frac{\partial^2 X}{\partial x^2} = -k. \tag{1.4}$$

Where k is a constant known as separation constant, which will be determined using the boundary conditions.
The solutions of (1.4) are:

$$T(t) = A_1 e^{(\eta - k)t} \quad \text{and} \quad X(x) = A_2 e^{i\sqrt{\frac{k}{\mu}}x}. \tag{1.5}$$

Where A_1 and A_2 are constants determined via the initial conditions. We are now ready to analyze this equation for several different geometries.

2 1D Cartesian Coordinates

We start from the $1D$ case in cartesian coordinates, hence (1.2) will look like:

$$\frac{\partial n}{\partial t} = \mu \frac{\partial^2 n}{\partial x^2} + \eta n. \tag{2.1}$$

This goes together with the boundary conditions and the initial condition. If our domain is $[0, L]$, those conditions read as:

$$\text{BC: } n(t,0) = n(t,L) = 0. \tag{2.2}$$

$$\text{IC: } n(0,x) = f(x). \tag{2.3}$$

Now we can use the same idea as in (1.3) which leads us exactly to (1.5):

$$T(t) = Ae^{(\eta-k)t} \quad \text{and} \quad X(x) = Be^{i\sqrt{\frac{k}{\mu}}x}, \tag{2.4}$$

but now we are able to find the unknown constants.
First of all notice that:

$$X(x) = Be^{i\sqrt{\frac{k}{\mu}}x} \tag{2.5}$$

$$= B_1 \cos\left(\sqrt{\frac{k}{\mu}}x\right) + B_2 \sin\left(\sqrt{\frac{k}{\mu}}x\right) \tag{2.6}$$

$$= B_2 \sin\left(\sqrt{\frac{k}{\mu}}x\right), \tag{2.7}$$

in order to ensure that $n(t,0) = 0$. Then we need to implement also $n(t,L) = 0$ and this happens if:

$$B_2 \sin\left(\sqrt{\frac{k}{\mu}}x\right) = B_2 \sin\left(\frac{p\pi x}{L}\right). \tag{2.8}$$

This allows us to find the separation constant k:

$$k = \mu\left(\frac{p\pi}{L}\right)^2 \tag{2.9}$$

Notice also that since the diffusion constant μ is, indeed, constant, (1.2) is a linear equation and therefore the superposition principle applies:

$$n(t,x) = \sum_{p=1}^{\infty} a_p \, e^{(\eta-\mu(\frac{p\pi}{L})^2)t} \sin\left(\frac{p\pi x}{L}\right). \tag{2.10}$$

Where the constant $A \cdot B_2$ has become a_p.
It's clear that for n to increase unbounded we want:

$$\eta > \mu\left(\frac{p\pi}{L}\right)^2 \tag{2.11}$$

$$L > p\pi\sqrt{\frac{\mu}{\eta}}. \tag{2.12}$$

The smallest value of L, the critical one, is obtained for $p = 1$ and the Python simulation gives (1 1D Cartesian Coordinates) $L = 11.05$ cm.
All we need to perform a dynamical simulation of the neutron diffusion are the coefficients a_p, which can be computed by exploiting the orthogonality of the sine function:

$$\int_0^L \sin\left(\frac{k\pi x}{L}\right) \sin\left(\frac{l\pi x}{L}\right) dx = \frac{L}{2}\delta_{l,k}. \tag{2.13}$$

In fact we can just take (2.10) for $t = 0$, that is just the initial condition $f(x)$ and integrate it against sine:

$$\int_0^L f(x) \sin\left(\frac{l\pi x}{L}\right) dx = \int_0^L \sum_{p=1}^{\infty} a_p \ \sin\left(\frac{p\pi x}{L}\right) \sin\left(\frac{l\pi x}{L}\right) dx \tag{2.14}$$

$$= \frac{L}{2} a_l. \tag{2.15}$$

By exchanging sum and integral.
So it follows that:

$$a_p = \frac{2}{L} \int_0^L f(x) \sin\left(\frac{l\pi x}{L}\right) dx. \tag{2.16}$$

As initial condition we choose:

$$f(x) = A \exp\left(\frac{-\lambda(x - \frac{L}{2})^2}{L^2}\right) \tag{2.17}$$

Where λ is some other parameter.
The Python simulation gives the following results:

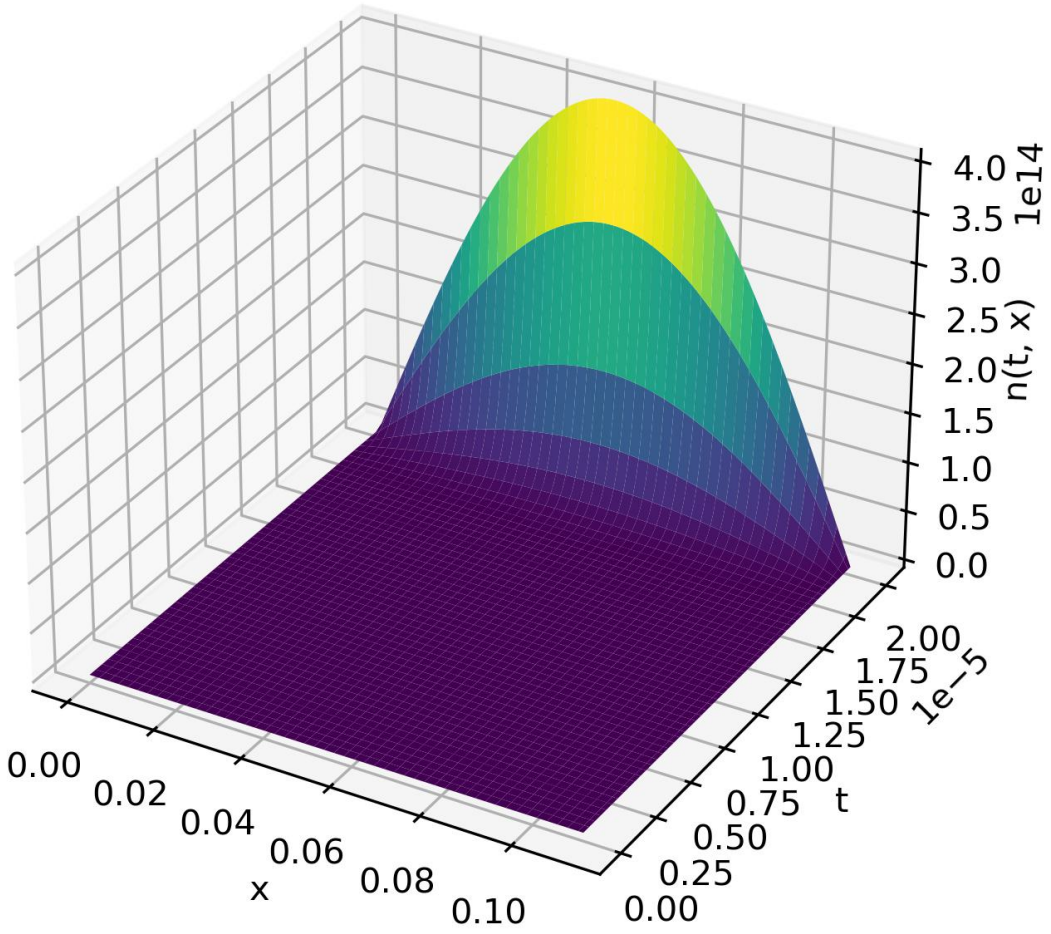

Figure 1: Plot of neutron density for $0 < t < 10^{-5}$. As we expected the neutron density blows up and it's interesting to check that if we take $L < L_{crit}$ this behaviour is not present. (From 1 1D Cartesian Coordinates.py).

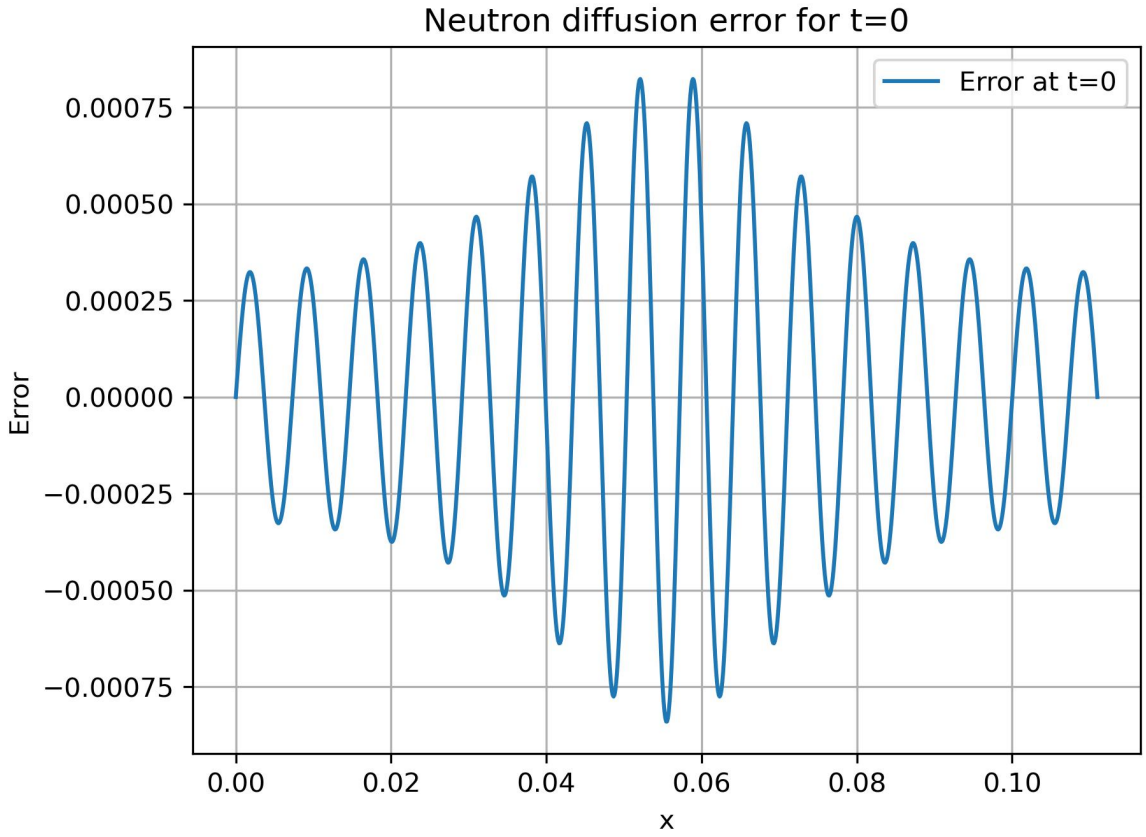

Figure 2: Plot of error: $n(0,x) - f(x)$. Our choice for $f(x)$ was a good choice since the initial neutron density is very close to it. (From 1 1D Cartesian Coordinates.py).

3 2D Cartesian Coordinates

We can now move to the $2D$ case, on a domain $[0, L_x] \times [0, L_y]$, in cartesian coordinates. Therefore (1.2) reduces to:

$$\frac{\partial n}{\partial t} = \mu \frac{\partial^2 n}{\partial x^2} + \mu \frac{\partial^2 n}{\partial y^2} + \eta n. \tag{3.1}$$

With boundary conditions and initial conditions:

$$\text{BC: } n(t,0,y) = n(t,L_x,y) = 0. \tag{3.2}$$
$$\text{BC: } n(t,x,0) = n(t,x,L_y) = 0. \tag{3.3}$$
$$\text{IC: } n(0,x,y) = f(x,y). \tag{3.4}$$

Again we postulate that $n(t,x,y) = T(t)X(x)Y(y)$, which leads us to:

$$\frac{1}{T}\frac{\partial T}{\partial t} - \eta = \frac{\mu}{X}\frac{\partial^2 X}{\partial x^2} + \frac{1}{Y}\frac{\partial^2 Y}{\partial y^2} = -\alpha. \tag{3.5}$$

From 1 PDE we obtained 3 ODEs, since the member of (3.5) involving both X and Y can be further split into 2 ODEs by variable separation:

$$\frac{d^2X}{dx^2} = \frac{-\alpha_1}{\mu}X, \tag{3.6}$$

$$\frac{d^2Y}{dy^2} = \frac{-\alpha_2}{\mu}Y. \tag{3.7}$$

Where $\alpha = \alpha_1 + \alpha_2$.
These equations have the same structure of the ones we met in Section 1, so we can solve them and implement the boundary conditions to find:

$$n(t,x,y) = \sum_{p,q=1}^{\infty} a_{pq}\, e^{(\eta-\alpha)t} \sin\left(\frac{p\pi x}{L_x}\right) \sin\left(\frac{q\pi y}{L_y}\right). \tag{3.8}$$

The separation constant α is still unknown, but for (3.8) to hold we must have:

$$\alpha_1 = \mu\left(\frac{p\pi}{L_x}\right)^2, \tag{3.9}$$

$$\alpha_2 = \mu\left(\frac{q\pi}{L_y}\right)^2. \tag{3.10}$$

Therefore it follows that (3.8) takes the form:

$$n = \sum_{p,q=1}^{\infty} a_{pq}\, e^{\left(\eta - \mu\pi^2\left(\frac{p^2}{L_x^2}+\frac{q^2}{L_y^2}\right)\right)t} \sin\left(\frac{p\pi x}{L_x}\right) \sin\left(\frac{q\pi y}{L_y}\right). \tag{3.11}$$

The neutron density will increase unbounded if the argument of the exponential is positive:

$$\eta > \mu\pi^2\left(\frac{p^2}{L_x^2} + \frac{q^2}{L_y^2}\right). \tag{3.12}$$

We can assume to be working in a square $L_x = L_y = L$ and invert (3.12) to find L. The smallest value of L is for $p = q = 1$:

$$L_{crit} = \pi\sqrt{\frac{2\mu}{\eta}}. \tag{3.13}$$

The Python simulation gives $L_{crit} = 15.62$ cm.
Now we're only missing the coefficient a_{pq}, which can be found in the same way we did in Section 2:

$$a_p = \frac{4}{L^2}\int_0^L \int_0^L f(x,y) \sin\left(\frac{p\pi x}{L}\right) \sin\left(\frac{q\pi y}{L}\right) dxdy. \tag{3.14}$$

As initial condition we choose:

$$f(x,y) = \frac{16xy}{L^2}\left(1-\frac{x}{L}\right)\left(1-\frac{y}{L}\right). \tag{3.15}$$

The Python simulation gives the following results:

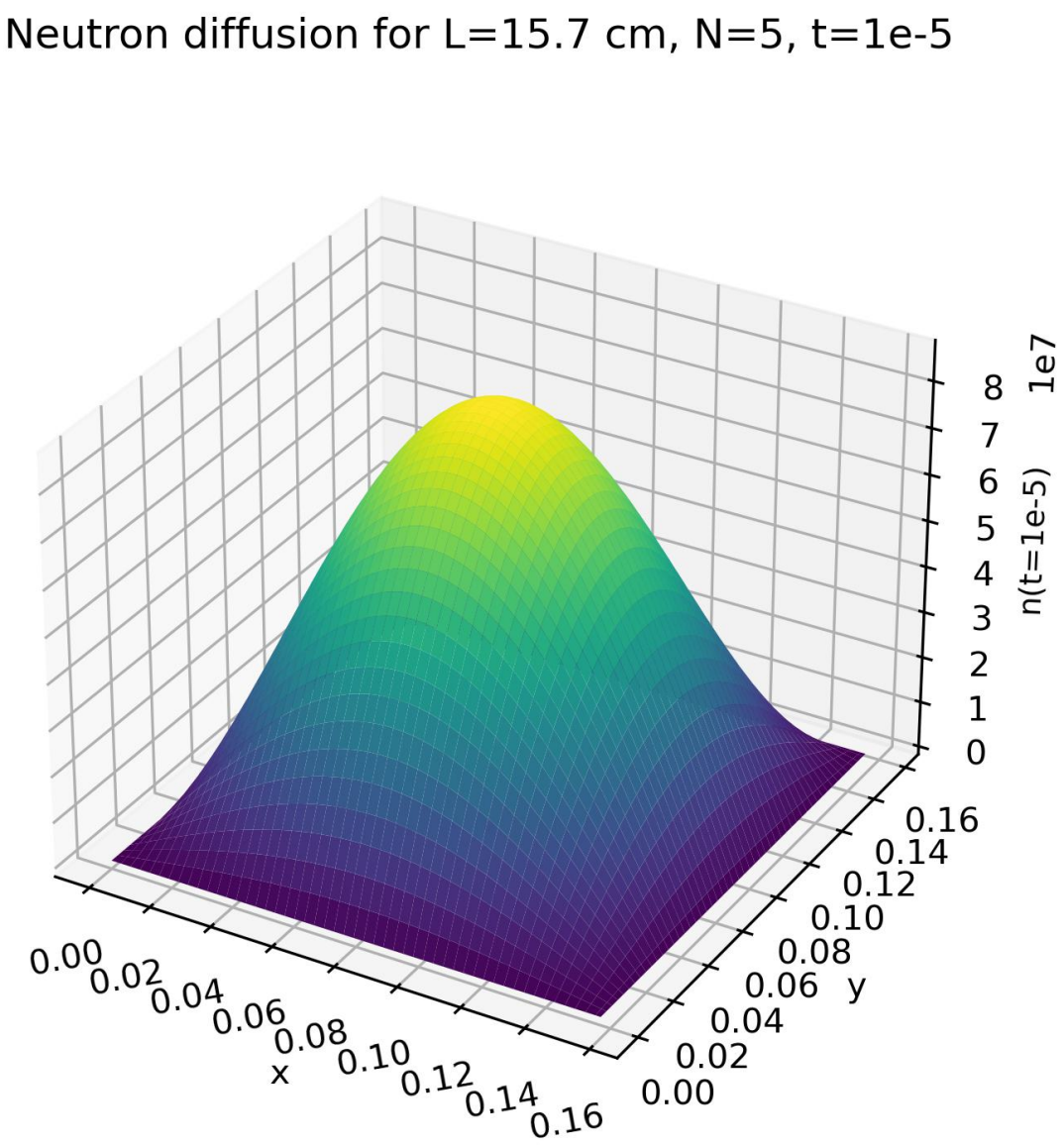

Figure 3: Plot of the neutron density for $t = 10^{-5}$. As we expected the neutron density blows up and again it's interesting to check that if we take $L < L_{crit}$ this behaviour is not present. (From 2 2D Cartesian Coordinates.py).

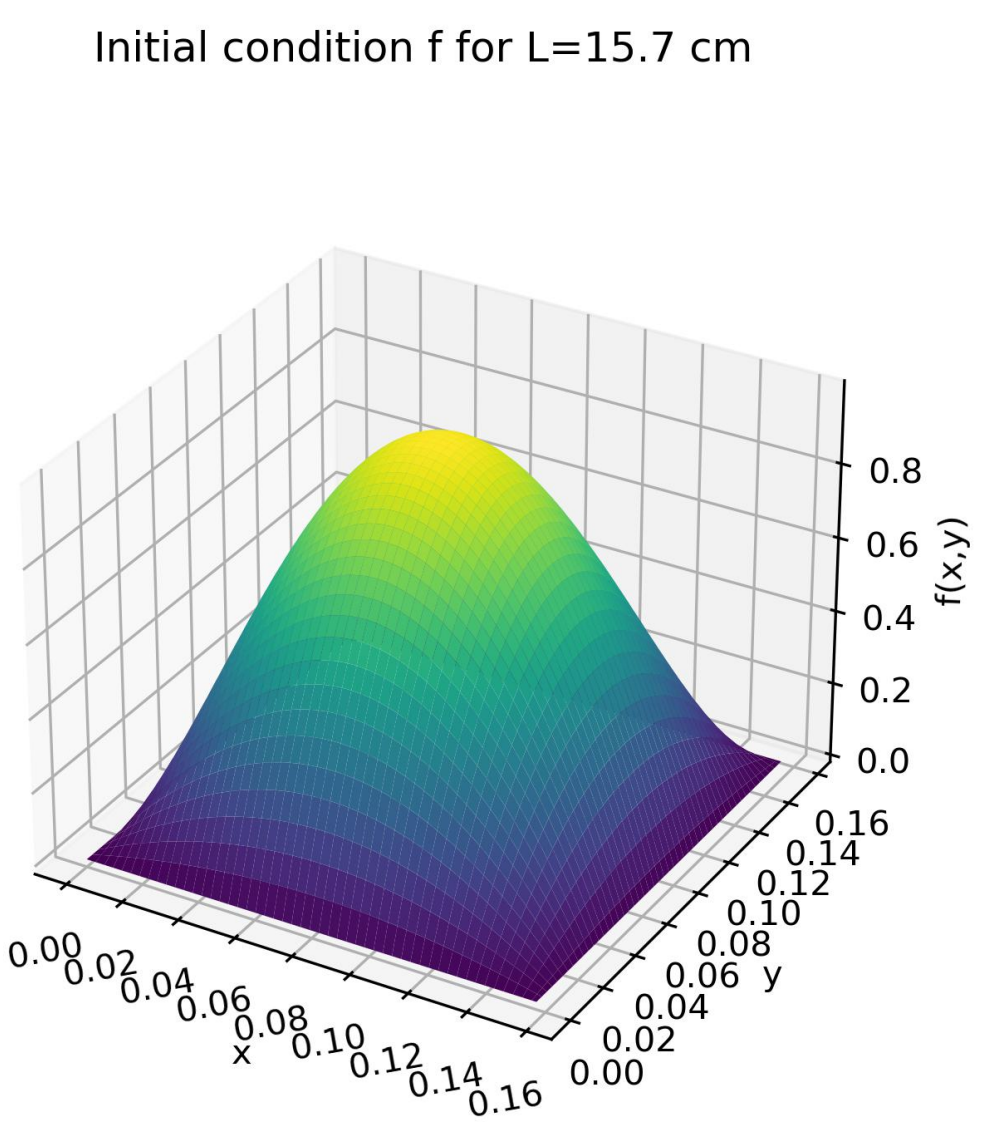

Figure 4: Plot of the initial condition $f(x, y)$. By comparing with (Figure 3) it's easy to see that the neutron density blows up very fast. (From 2 2D Cartesian Coordinates.py).

4 3D Cartesian Coordinates

We can move to the $3D$ case, with a domain $[0, L_x] \times [0, L_y] \times [0, L_z]$, in caretesian coordinates, hence (1.2) will take the form:

$$\frac{\partial n}{\partial t} = \mu \frac{\partial^2 n}{\partial x^2} + \mu \frac{\partial^2 n}{\partial y^2} + \mu \frac{\partial^2 n}{\partial z^2} + \eta n. \tag{4.1}$$

With boundary conditions and initial conditions:

$$\text{BC: } n(t, 0, y, z) = n(t, L_x, y, z) = 0. \tag{4.2}$$
$$\text{BC: } n(t, x, 0, z) = n(t, x, L_y, z) = 0. \tag{4.3}$$
$$\text{BC: } n(t, x, y, 0) = n(t, x, y, L_z) = 0. \tag{4.4}$$
$$\text{IC: } n(0, x, y, z) = f(x, y, z). \tag{4.5}$$

We can use the same ideas we used in the previous sections (just notice that now $\alpha = \alpha_1 + \alpha_2 + \alpha_3$) to find:

$$n(t, x, y, z) = \sum_{p,q,r=1}^{\infty} a_{pqr}\, e^{\left(\eta - \mu\pi^2\left(\frac{p^2}{L_x^2} + \frac{q^2}{L_y^2} + \frac{r^2}{L_z^2}\right)\right)t} \sin\left(\frac{p\pi x}{L_x}\right) \sin\left(\frac{q\pi y}{L_y}\right) \sin\left(\frac{r\pi y}{L_z}\right). \tag{4.6}$$

Again we find L_{crit} by requiring n to increase unbounded, assuming $L_x = L_y = L_z = L$ and fixing $p = q = r = 1$:

$$L_{crit} = \pi \sqrt{\frac{3\mu}{\eta}}. \tag{4.7}$$

The Python simulation gives $L_{crit} = 19.14$ cm, $V_{crit} = 7007$ cm^3 and $M_{crit} = 131$ kg. Once again we are only missing the coefficients a_{pqr} and once again they can be found as:

$$a_p = \frac{8}{L^3} \int_0^L \int_0^L \int_0^L f(x, y, z) \sin\left(\frac{p\pi x}{L}\right) \sin\left(\frac{q\pi y}{L}\right) \sin\left(\frac{r\pi z}{L}\right) dxdydz. \tag{4.8}$$

As initial condition we choose:

$$f(x, y) = \frac{8xyz}{L^3} \left(1 - \frac{x}{L}\right) \left(1 - \frac{y}{L}\right) \left(1 - \frac{z}{L}\right). \tag{4.9}$$

The Python simulation gives the following results:

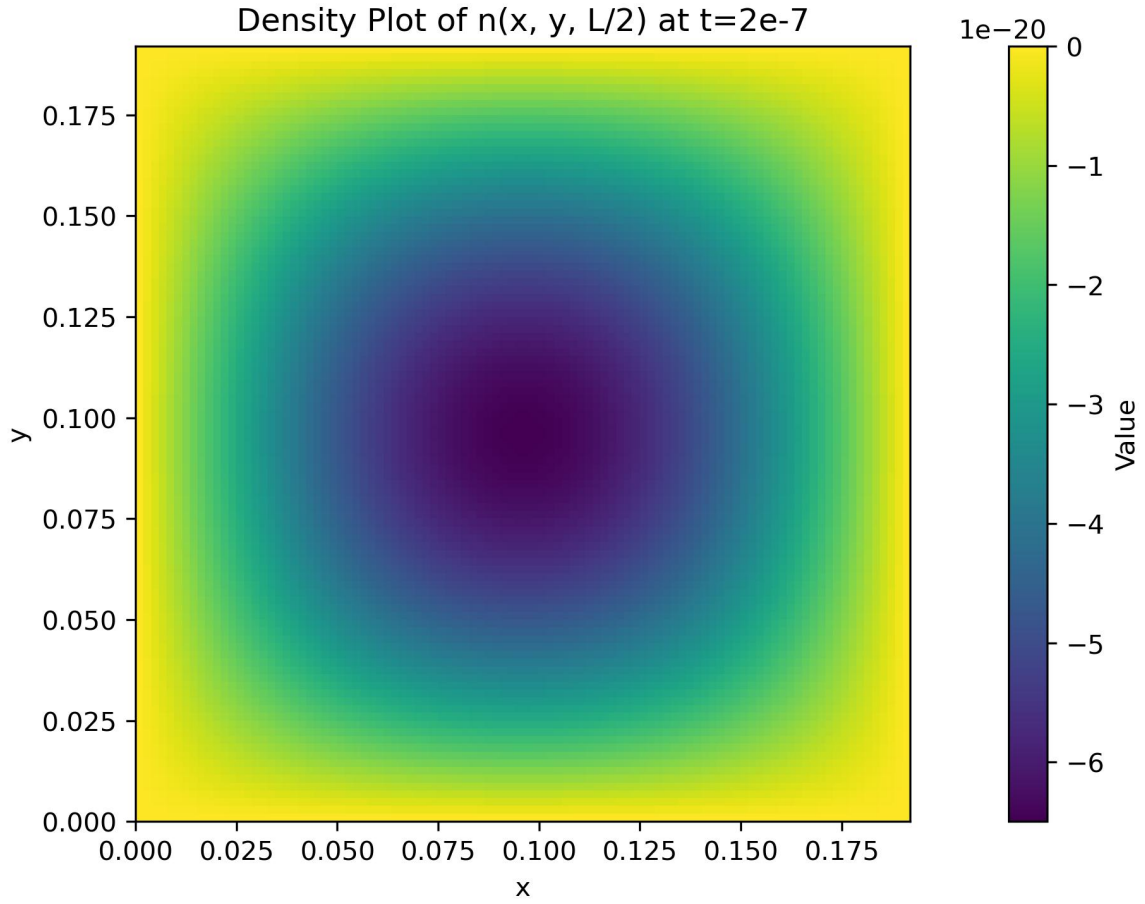

Figure 5: Plot of the neutron density for $t = 2 \cdot 10^{-7}$ and $z = L/2$. (From 3 3D Cartesian Coordinates.py).

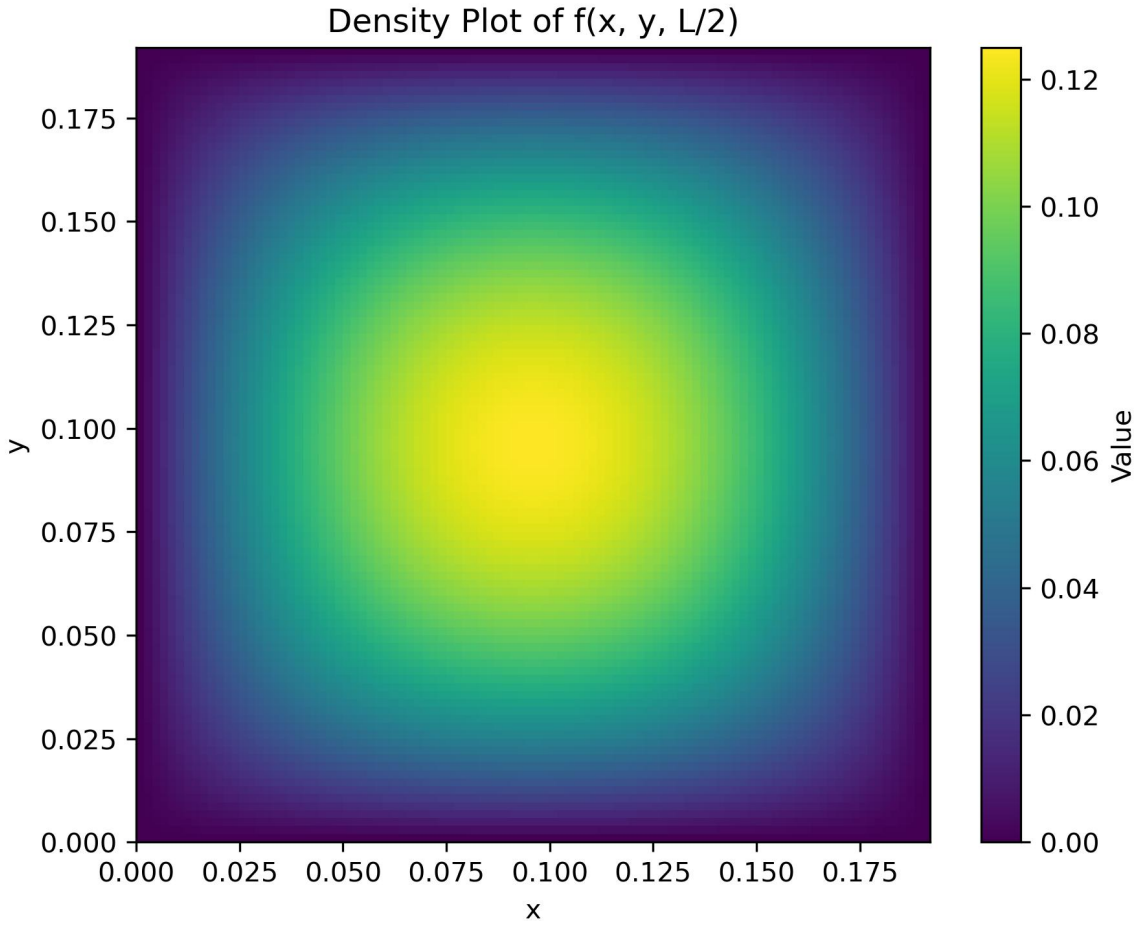

Figure 6: Plot of the initial condition $f(x, y, z)$ for $z = L/2$. (From 3 3D Cartesian Coordinates.py).

Clearly the result of (Figure 5) makes no physical sense since we get a negative density, despite the initial condition being physical. For this reason I tried to replicate the plot of (Figure 5) also in Mathematica, which gives the correct result instead. My hypothesis is that it has to do with how Mathematica and Python treat floating points.

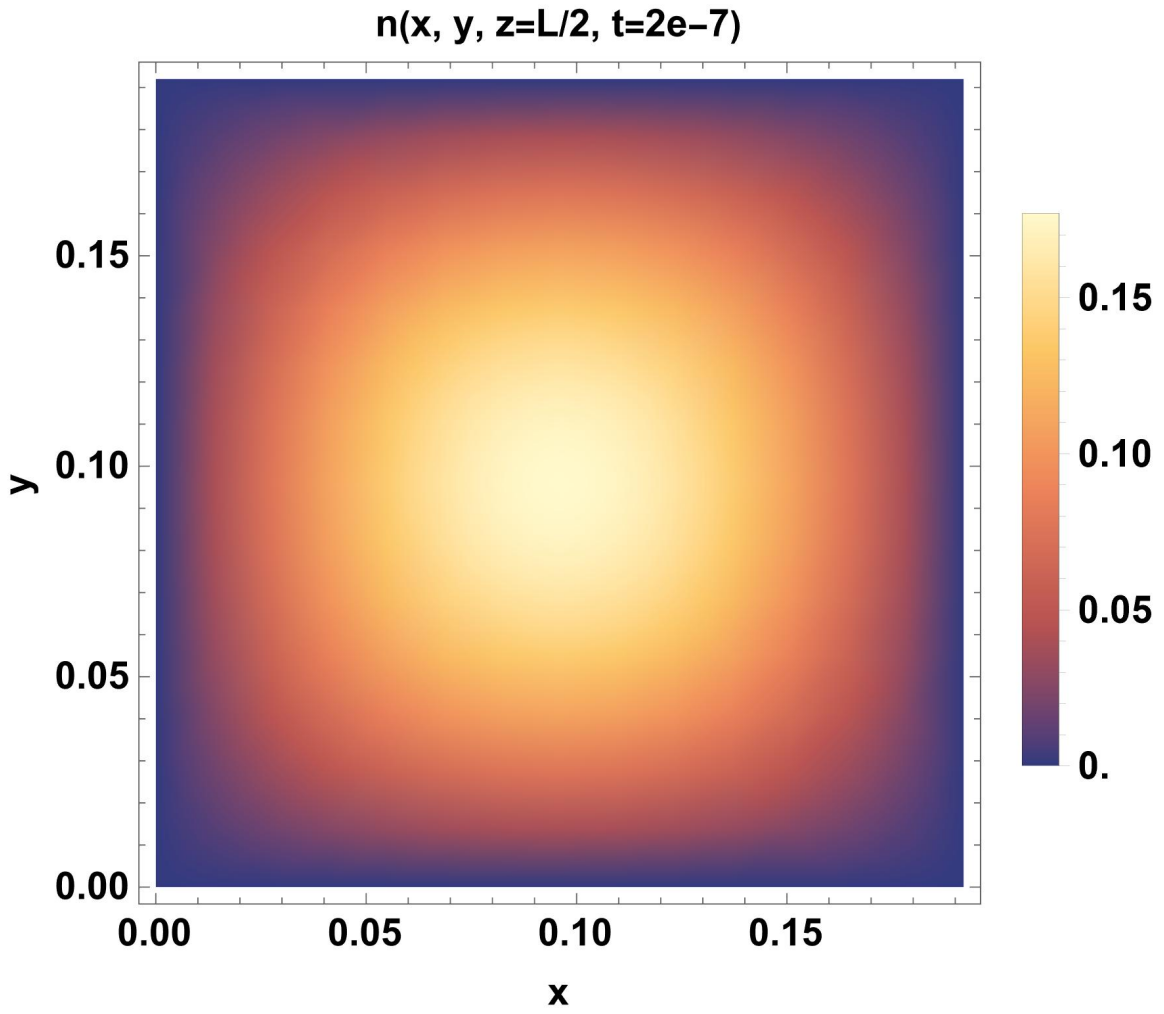

Figure 7: Plot of the initial condition $f(x, y, z)$ for $z = L/2$ in Mathematica. (From 3 Density plot 3d cartesian.nb).

5 3D Cylindrical Coordinates

We can now move to the $3D$ case in cylindrical coordinates, where (1.2) takes the form:

$$\frac{\partial n}{\partial t} = \mu \left(\frac{\partial^2}{\partial r^2} + \frac{1}{r} \frac{\partial}{\partial r} \right) n + \mu \frac{\partial^2 n}{\partial z^2} + \eta n. \tag{5.1}$$

The derivative term of the right hand side of (5.1) is the ∇^2 operator in cylindrical coordinates, where the term $\frac{\partial^2 n}{\partial \phi^2}$ is missing due to radial symmetry.

The boundary conditions and initial conditions are:

$$\text{BC: } n(t, r_1, z) = 0. \tag{5.2}$$
$$\text{BC: } n(t, r, 0) = n(t, x, L) = 0. \tag{5.3}$$
$$\text{IC: } n(0, r, z) = f(r, z). \tag{5.4}$$

Where r_1 is the radius of the cylinder and $z = 0,\ L$ are the base and the top of the cylinder respectively.
To solve (5.1) we exploit the same method used in the previous sections. We postulate the existence of a solution of the form:

$$n(t, r, z) = T(t)R(r)Z(z). \tag{5.5}$$

We plug (5.5) into (5.1) to get:

$$\frac{1}{T}\frac{\partial T}{\partial t} - \eta = \frac{\mu}{R}\frac{\partial^2 R}{\partial r^2} + \frac{\mu}{rR}\frac{\partial R}{\partial r} + \frac{\mu}{Z}\frac{\partial^2 Z}{\partial z^2} = -l. \tag{5.6}$$

Once again from 1 PDE we obtained 3 ODEs, since the second member of (5.6) can be futher split into 2 ODEs by variable separation:

$$\frac{\mu}{R}\frac{\partial^2 R}{\partial r^2} + \frac{\mu}{rR}\frac{\partial R}{\partial r} + l = -\frac{\mu}{Z}\frac{\partial^2 Z}{\partial z^2} = k. \tag{5.7}$$

Where k is a second separation variable.
The ODEs we get are therefore:

$$\frac{d^2 Z}{dz^2} = -\frac{k}{\mu}Z, \tag{5.8}$$
$$\frac{d^2 R}{dr^2} + \frac{1}{Rr}\frac{dR}{dr} + \frac{l-k}{\mu}R = 0. \tag{5.9}$$

The solutions for $T(t)$ and $Z(z)$ are, once we implement the boundary conditions

$$T(t) = Ae^{(\eta - l)t} \tag{5.10}$$
$$Z(z) = B\sin\left(\frac{p\pi z}{L}\right). \tag{5.11}$$

This means that:

$$k = \mu\left(\frac{p\pi}{L}\right)^2 \tag{5.12}$$

Equation (5.9) is the most interesting. It turns out it's a Bessel equation and thus has solution:

$$R(r) = CJ_0\left(r\sqrt{\frac{l}{\mu} - \left(\frac{p\pi}{L}\right)^2}\right). \tag{5.13}$$

Now we can determine the separation constant l since, to satisfy the BC, we want $R(r_1) = 0$. In fact we can parameterize l as:

$$l = \mu \left(\frac{\alpha_q^2}{r_1^2} + \frac{p^2 \pi^2}{L^2} \right). \tag{5.14}$$

Where α_q is the q-th zero of J_0.
Now we can write the neutron density. Making use once again of the superposition principle one finds:

$$n(t, r, z) = \sum_{p,q=1}^{\infty} a_{pq} \ \exp\left(\frac{\eta r_1^2 L^2 - \mu(\alpha_q^2 L^2 + p^2 \pi^2 r_1^2)}{r_1^2 L^2} t \right) \sin\left(\frac{p\pi z}{L} \right) J_0\left(\frac{\alpha_q}{r_1} r \right). \tag{5.15}$$

However by an appropriate choice of the initial condition we can have all $a_p q = 0$ for $p > 1$, so (5.15) reduces to:

$$n(t, r, z) = \sum_{q=1}^{\infty} a_{1q} \ \exp\left(\frac{\eta r_1^2 L^2 - \mu(\alpha_q^2 L^2 + p^2 \pi^2 r_1^2)}{r_1^2 L^2} t \right) \sin\left(\frac{\pi z}{L} \right) J_0\left(\frac{\alpha_q}{r_1} r \right). \tag{5.16}$$

As usual we find the criticality parameters by requiring unboundedness of the neutron density and setting $q = 1$:

$$L_{crit} = \pi \sqrt{\frac{3\mu}{\eta}}, \tag{5.17}$$

$$r_{crit} = \alpha_1 \sqrt{\frac{3\mu}{2\eta}}. \tag{5.18}$$

The Pyhton simulation gives $L_{crit} = 19.14$ cm, $r_{crit} = 10.36$ cm, $V_{crit} = 6450$ cm^3 and $M_{crit} = 121$ kg.
Now we just have to find a_{1q} and then we can proceed with the full dynamical simulation. Those coefficients are found using once again orthogonality, but for the Bessel functions this time. In fact, given a function $f(x)$ we can perform a generalized Fourier expansion, in the region $0 < x < x_0$ using Bessel function:

$$f(x) = \sum_{n=0}^{\infty} c_n J_m\left(\frac{\alpha_n}{x_0} x \right). \tag{5.19}$$

Where α_n is the n-th zero of J_m. It follows that:

$$\int_0^{x_0} x f(x) J_m\left(\frac{\alpha_q}{x_0} x \right) dx = \int_0^{x_0} \sum_{n=0}^{\infty} c_n J_m\left(\frac{\alpha_n}{x_0} x \right) x J_m\left(\frac{\alpha_q}{x_0} x \right) dx \tag{5.20}$$

$$= \sum_{n=0}^{\infty} \frac{x_0^2}{2} c_n J_{m+1}^2(\alpha_n) \, \delta_{n,q} \tag{5.21}$$

$$= \frac{x_0^2}{2} c_q J_{m+1}^2(\alpha_q). \tag{5.22}$$

We can invert this to find c_q and in this way we find:

$$a_{pq} = \frac{4}{Lr_1^2 J_1^2(\alpha_q)} \int_0^{r_1} \int_0^L J_0\left(\frac{\alpha_q}{r_1} r\right) r f(r,z) \sin\left(\frac{p\pi z}{L}\right) dr dz \tag{5.23}$$

As we said before an appropriate choice of $f(r,z)$ reduces a_{pq} to a_{1q}, in fact if we take:

$$f(r,z) = \left(1 - \left(\frac{r}{r_1}\right)^2\right) \sin\left(\frac{\pi z}{L}\right), \tag{5.24}$$

by the orthogonality of the sine functions we get a $\delta_{1,p}$ and therefore:

$$a_{1q} = \frac{4}{Lr_1^2 J_1^2(\alpha_q)} \int_0^{r_1} \int_0^L J_0\left(\frac{\alpha_q}{r_1} r\right) r \left(1 - \left(\frac{r}{r_1}\right)^2\right) \sin^2\left(\frac{\pi z}{L}\right) dr dz \tag{5.25}$$

The Python simulation gives the following results:

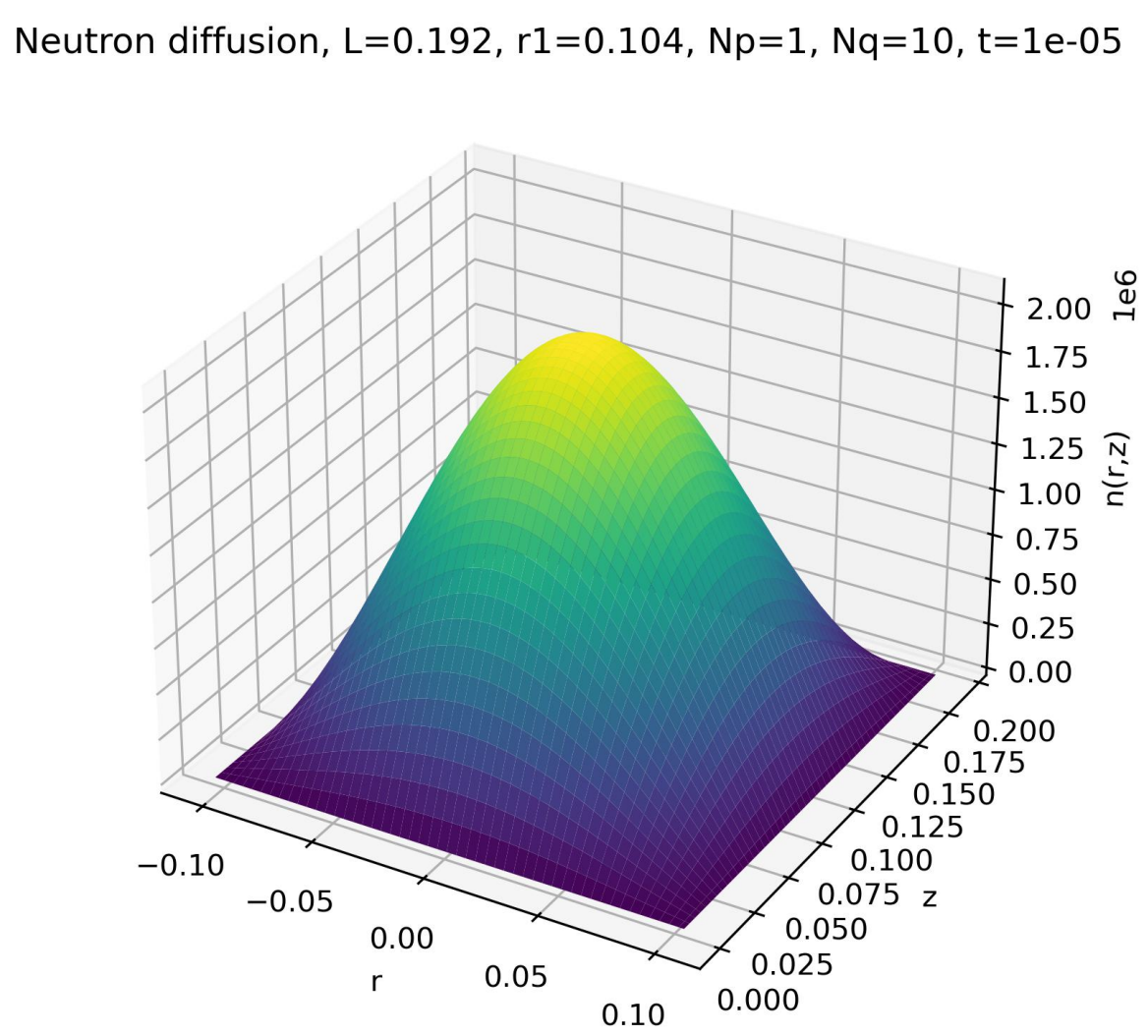

Figure 8: Plot of the neutron density for $t = 10^{-5}$. As we expected the neutron density blows up and again it's interesting to check that if we take $L < L_{crit}$ this behaviour is not present. (From 4 3D Cylindrical Coordinates.py)

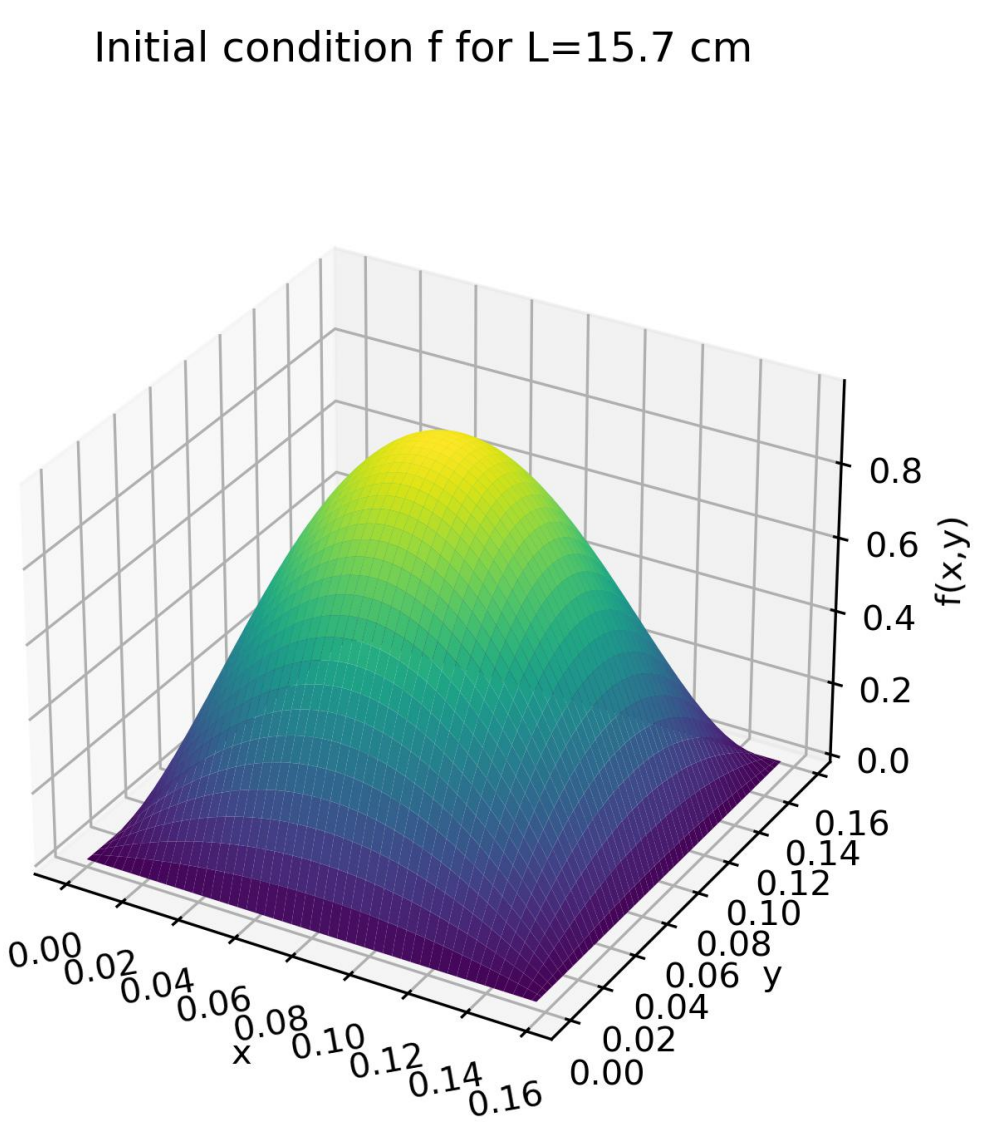

Figure 9: Plot of the initial condition $f(r, z)$. By comparing with (Figure 8) it's easy to see that the neutron density blows up very fast. (From 4 3D Cylindrical Coordinates.py)

6 3D Spherical Coordinates

We can now consider the $3D$ case in cylindrical coordinates, where (1.2) takes the form:

$$\frac{\partial n}{\partial t} = \mu \left(\frac{\partial^2}{\partial r^2} + \frac{2}{r}\frac{\partial}{\partial r} \right) n + \eta n. \tag{6.1}$$

As in Section 5 the derivative term on the right hand side of (6.1) is the ∇^2 operator in spherical coordinates, where the terms $\frac{\partial n}{\partial \theta^2}$ and $\frac{\partial^2 n}{\partial \phi^2}$ are absent due to the radial symmetry. Thus the spatial degrees of freedom are reduced to one so we can expect results similar to the ones we find in Section 2. It also follows that we need only boundary condition:

$$\text{BC: } n(t, r_1) = 0, \tag{6.2}$$

$$\text{IC: } n(0, t) = f(r). \tag{6.3}$$

Where r_1 is the radius of the sphere.
As usual we assume that $n(t, r) = T(t)R(r)$ is a solution and plug it into (6.1) to get 2 ODEs:

$$\frac{dT}{dt} = (\eta - \alpha)\, T, \tag{6.4}$$

$$\frac{d^2R}{dr^2} + \frac{2}{r}\frac{dR}{dr} + \frac{\alpha}{\mu}R = 0. \tag{6.5}$$

Where α is the separation constant. The solution of (6.4) is the usual exponential $T(t) = Ae^{(\eta-\alpha)t}$ and the solution of (6.5) is:

$$R(r) = \frac{B}{r} \sin\left(\sqrt{\frac{\alpha}{\mu}} r \right) \tag{6.6}$$

$$= \frac{B}{r} \sin\left(\frac{p\pi r}{r_1} \right). \tag{6.7}$$

The second equality for (6.7) is obtained by imposing (6.2). The full solution would also contain a cosine term, but it vanishes because we ask $R(r)$ to be finite at $r = 0$.
Now we can use the superposition principle and write:

$$n(t, r) = \sum_{p=1}^{\infty} \frac{a_p}{r}\, e^{(\eta - \mu(\frac{p\pi}{r_1})^2)t} \sin\left(\frac{p\pi r}{r_1} \right). \tag{6.8}$$

It follows that n will increase unbounded if:

$$\eta > \mu \left(\frac{p\pi}{r} \right)^2. \tag{6.9}$$

The critical radius is obtained from (6.9) for $p = 1$ and the Python simulation gives $r_{crit} = 11.04$ cm, $V_{crit} = 5649$ cm^3 and $M_{crit} = 106$ kg.
Once again we need to compute the coefficients a_p and once again we use orthogonality:

$$\int_0^{r_1} r f(r) \sin\left(\frac{l\pi r}{r_1}\right) dr = \int_0^{r_1} \sum_{p=1}^{\infty} a_p \sin\left(\frac{p\pi r}{r_1}\right) \sin\left(\frac{l\pi r}{r_1}\right) dr \tag{6.10}$$

$$= \frac{r_1}{2} a_l \tag{6.11}$$

By exchanging sum and integral.
As initial condition we choose:

$$f(r) = 1 - \left(\frac{r}{r_1}\right)^2 \tag{6.12}$$

The Python simulation gives the following results:

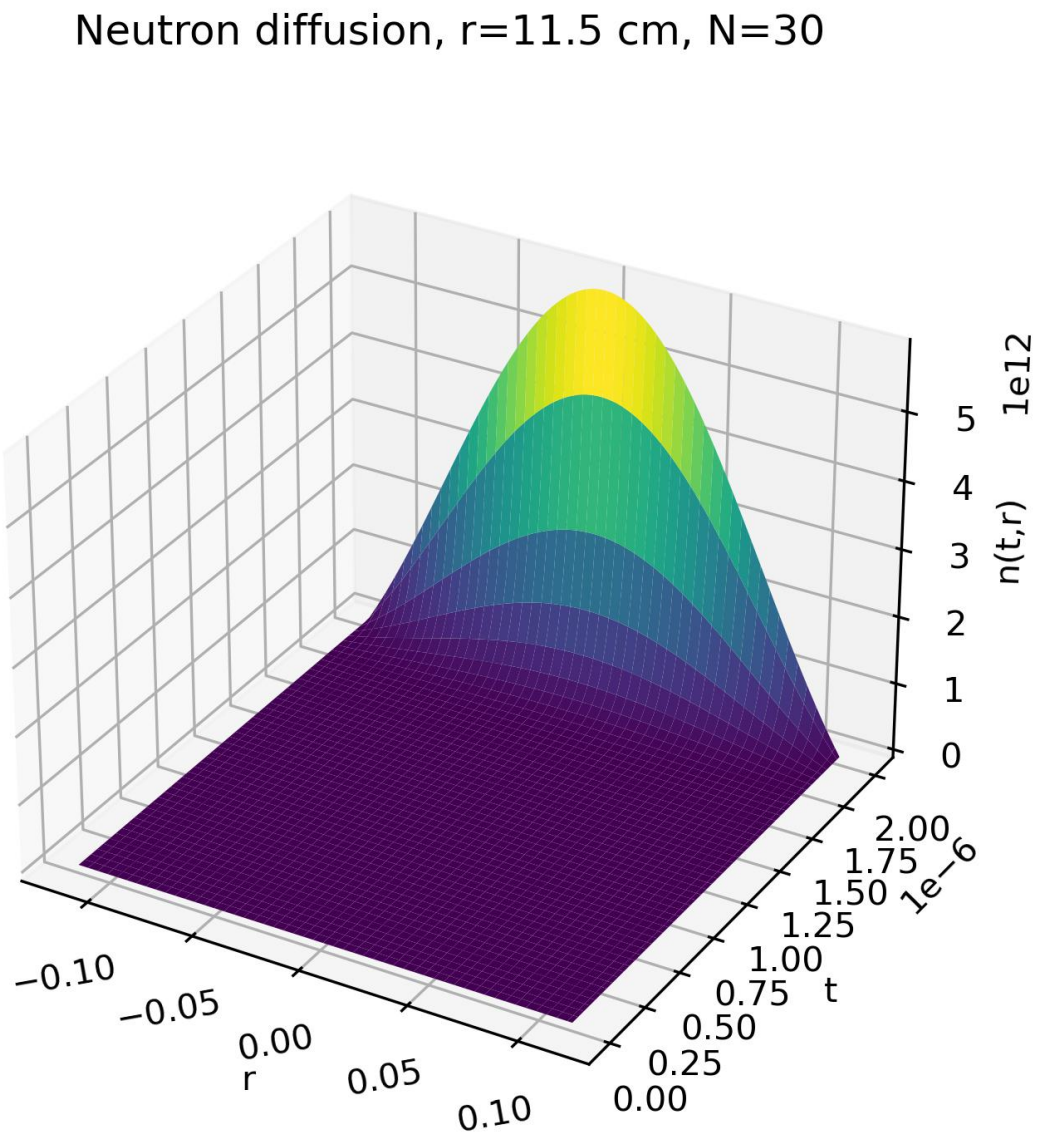

Figure 10: Plot of neutron density for $0 < t < 2 \cdot 10^{-5}$. As we expected the neutron density blows up and again it's interesting to check that if we take $r < r_{crit}$ this behaviour is not present. (From 5 3D Spherical Coordinates Dirichlet BC.py).

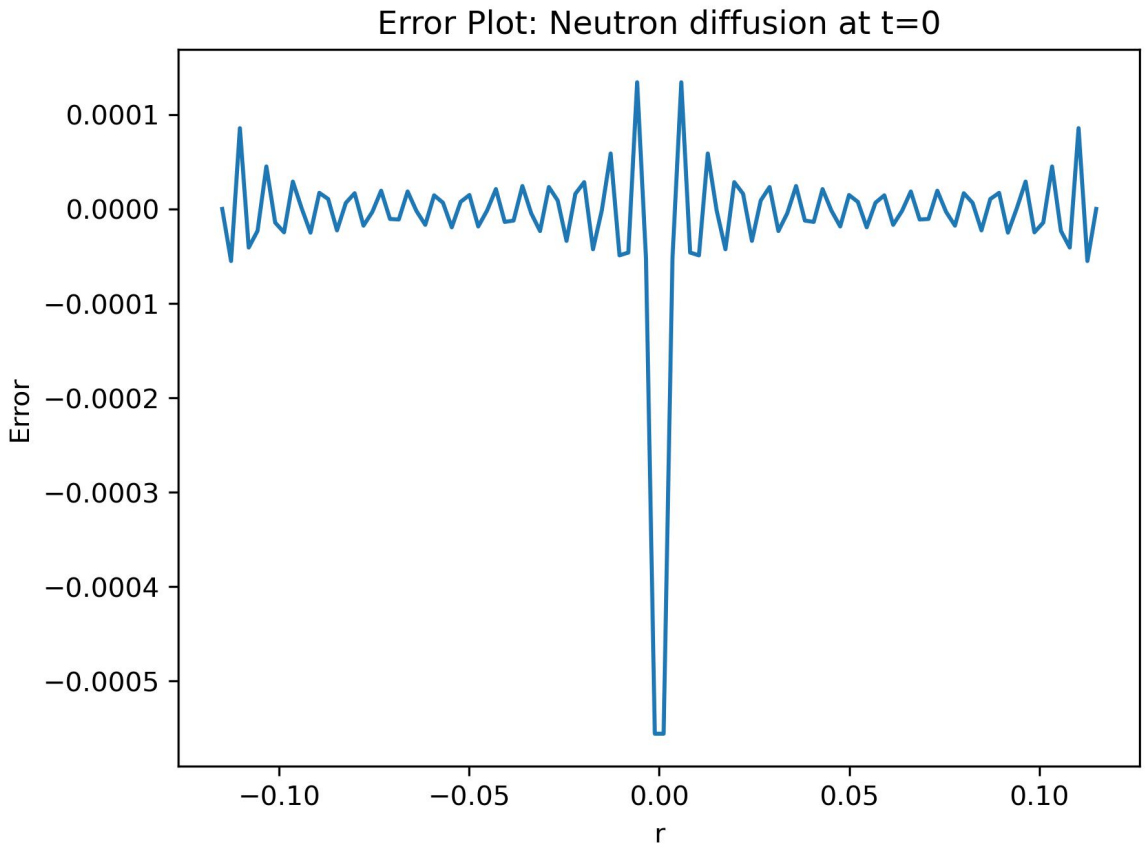

Figure 11: Plot of error: $n(0,r) - f(r)$. Our choice for $f(r)$ was a good choice since the initial neutron density is very close to it. (From 5 3D Spherical Coordinates Dirichlet BC.py).

7 3D Spherical Coordinates with Neumann BC

The whole analysis we carried out in the preceding sections is based on Dirichlet boundary conditions, which are equivalent to assuming that neutrons can't escape. A more physical choice is represented by Neumann boundary conditions that, as we shall see, will also lead us to a much smaller value of M_{crit}. For this reason we now want to repeat the calculations of Section 6 but with these new boundary conditions.
The diffusion equation is still 6.1:

$$\frac{\partial n}{\partial t} = \mu \left(\frac{\partial^2}{\partial r^2} + \frac{2}{r} \frac{\partial}{\partial r} \right) n + \eta n. \tag{7.1}$$

This goes together with:

$$\text{BC: } \left.\frac{dn}{dt}\right|_{r=r_1} = -\left.\frac{3n}{2\lambda_t}\right|_{r=r_1} \tag{7.2}$$

$$\text{IC: } n(0,t) = f(r). \tag{7.3}$$

Where r_1 is still the radius of the sphere and λ_t is the transport free path.
Now we can solve (7.1) with the variable separation method, as we did in Section 6. For this time, however, we prefer to keed the separation constant α in the equation for $R(r)$, so the solutions turn out to be:

$$T(t) = Ae^{(-\alpha t)}, \tag{7.4}$$

$$R(r) = \frac{B}{r}\sin(kr). \tag{7.5}$$

Now we can implement (7.2):

$$\frac{dR}{dr} = -\frac{3R}{2\lambda_t}. \tag{7.6}$$

After taking the derivative and some algebraone finds:

$$-1 + kr\cot(kr) + \frac{3r}{2\lambda_t} = 0. \tag{7.7}$$

If we fix $r = r_1$ in (7.7) we can find the corresponding α and viceversa if we fix α we can find the corresponding r. As usual we're interested in r_{crit}, which is obtained for $\alpha = 0$. The Python simulation gives $r_{crit} = 8.363$ cm, $V_{crit} = 2450$ cm^3 and $M_{crit} = 45.8$ kg. As we can see the more physical Nuemann boundary conditions give a much smaller critical mass.
Now we can assume to have a sphere with radius larger than the critical one, e.g. $R = 8.5$ cm, and compute the corresponding α and k, which, according to Python, are $\alpha = -4.068 \cdot 10^{-6}$ s and $k = 0.2808$ cm^{-1} .

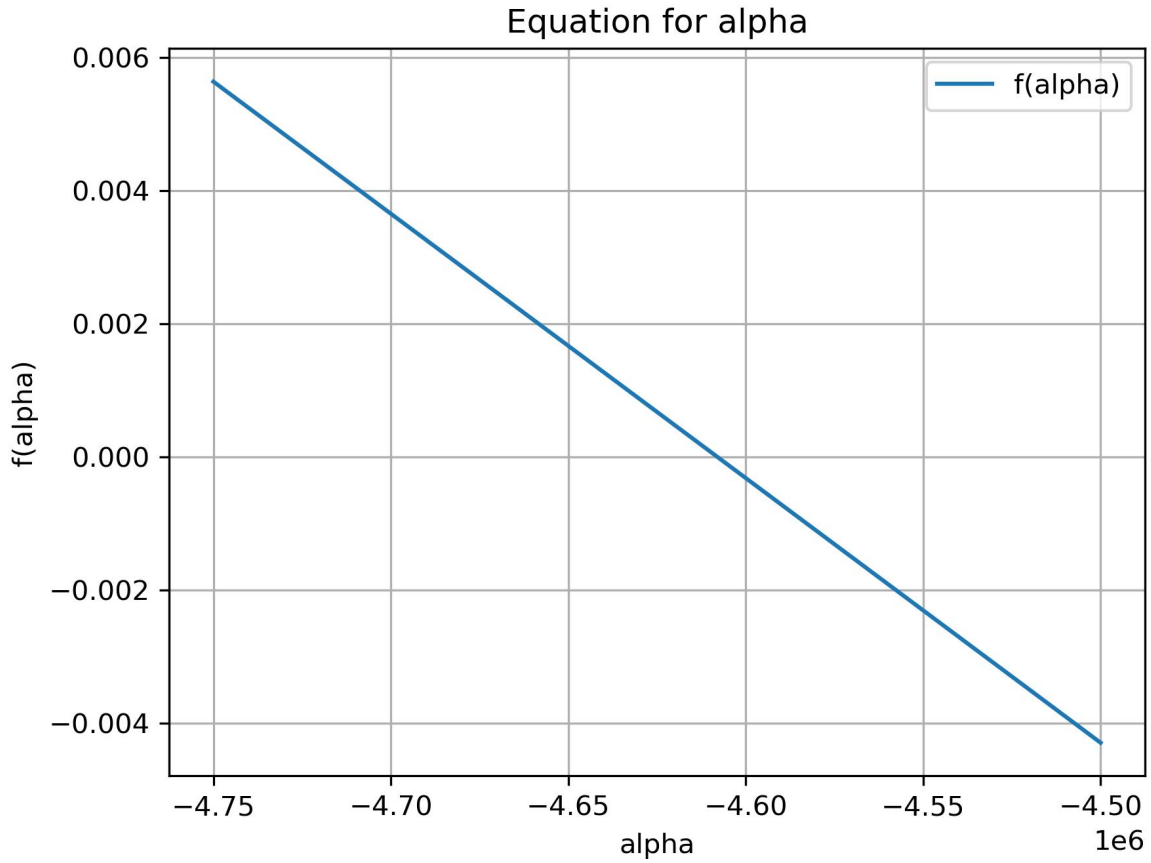

Figure 12: Plot of 7.7 for $r = R = 8.5$ cm in the range where its zero is found. (From 6 3D Spherical Coordinates Neumann BC.py).

We can now write down the neutron density:

$$n(t, r) = \frac{A}{r} e^{(-\alpha t)} \sin(kr) \,. \tag{7.8}$$

We only have to determine the constant $A \cdot B \equiv A$ and we do it by setting $n(0, R) = 1$, which gives $A = 12.415$ cm^{-2}.
The Python simulation gives the following results:

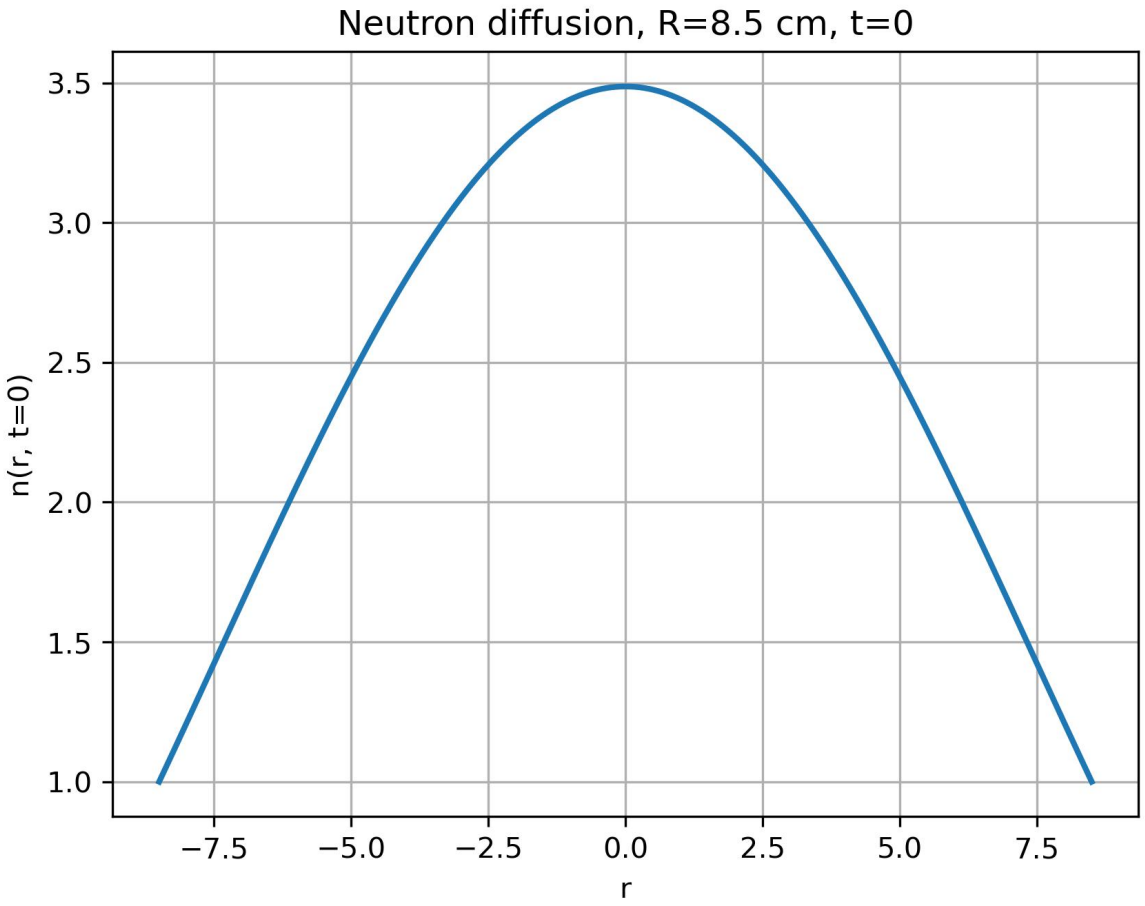

Figure 13: Plot of the neutron density at $t = 0$ for $r = R = 8.5$ cm. (From 6 3D Spherical Coordinates Neumann BC.py).

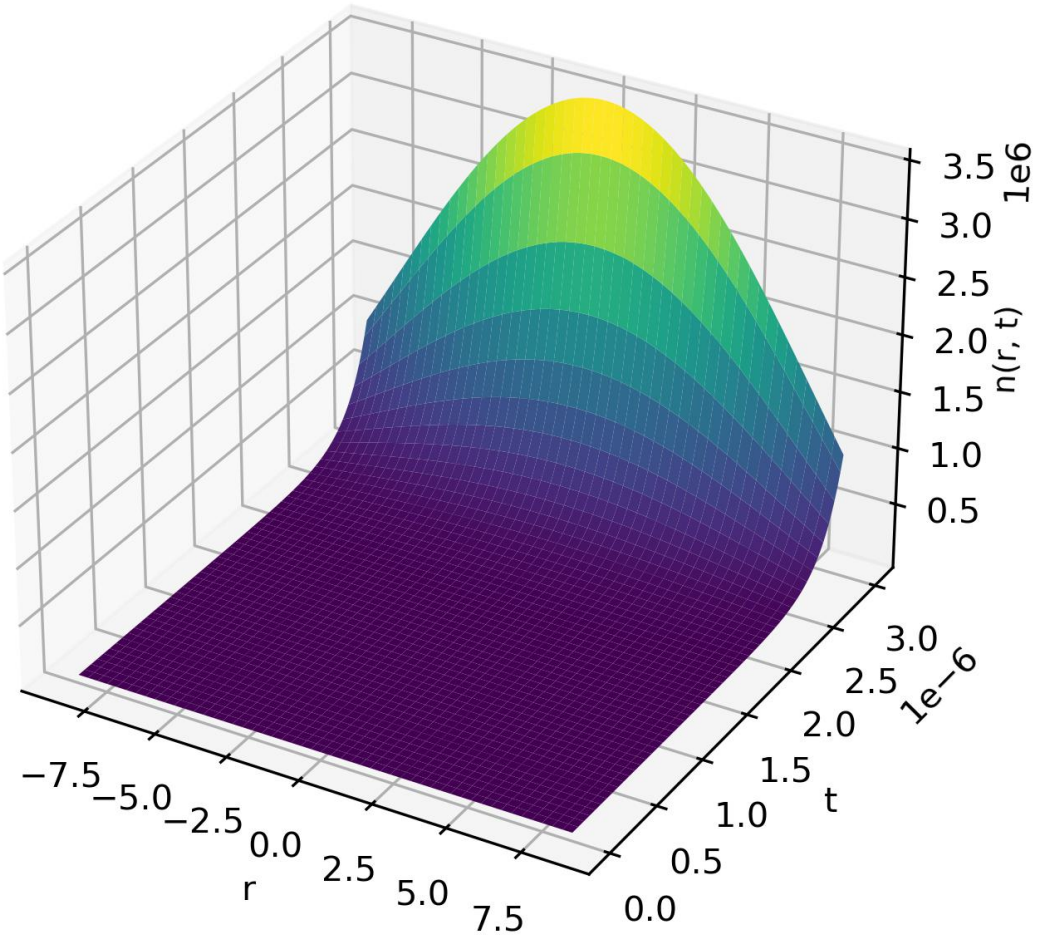

Figure 14: Plot of neutron density for $0 < t < 3 \cdot 10^{-6}$. As we expected the neutron density blows up and again it's interesting to check that if we take $r < r_{crit}$ this behaviour is not present. (From 6 3D Spherical Coordinates Neumann BC.py).

8 Neutron Mean Free Path

To conclude we want to find the mean free path, that is the average distance travelled without any interaction for a neutron. To do that we need to find the probability

density of interaction of a single neutron and compute the expectation value of the position for such probability density. Consider a slab of thickness s and area Σ which is bombarded by neutron at a rate R_0. If n is the density if nuclei in the slab and σ is their cross-section to neutrons, then the effective area for the interaction will be $\Sigma\sigma ns$ and the rate of reaction will be $R = R_0\Sigma\sigma ns$. From the rate we can find the probability of interaction just by dividing it by the flux:

$$P_{reac} = \sigma sn. \tag{8.1}$$

Neutrons that do not interact escape from the slab and since probabilities are normalized to 1:

$$P_{esc} = 1 - P_{reac} \tag{8.2}$$

$$= 1 - \sigma sn. \tag{8.3}$$

Consider now a block of thickness x which is made out of m slabs, all of thickens s, so $x = ms$. If we have N_0 incoming neutrons that hit the block, those that escape are $N_0 P_{esc}$ for the first slab, $N_0 P_{esc}^2$ for the second one and so until we reach the last slab, where we find a $N_0 P_{esc}^m$ escaped neutrons.

$$N_{esc} = N_0 \left(1 - \sigma sn\right)^{\frac{x}{s}} \tag{8.4}$$

$$= N_0 \left(1 + z\right)^{\frac{-\sigma nx}{z}} \tag{8.5}$$

$$= N_0 \left[\left(1 + z\right)^{\frac{1}{z}}\right]^{-\sigma nx}. \tag{8.6}$$

Where $z = -\sigma ns$.
We can assume $s \to 0$ so that also $z \to 0$ and we get and exponential by definition:

$$N_{esc} = N_0 e^{-\sigma nx}. \tag{8.7}$$

Again neutron that do not escape, interact, so:

$$N_{reac} = N_0 \left(1 - e^{-\sigma nx}\right). \tag{8.8}$$

From (8.8) we can find the probability of interaction just dividing by N_0 and from the probability the probability density by taking a derivative with respect to x:

$$p_{reac} = \frac{dP_{reac}}{dx} = \sigma n e^{-\sigma nx} \tag{8.9}$$

As we said the mean free path is the expectation value of the position for the probability density (8.9):

$$\langle x \rangle = \sigma n \int_0^\infty x e^{-\sigma nx} dx \tag{8.10}$$

$$= \frac{1}{\sigma n}. \tag{8.11}$$

The Python simulation gives $\langle x \rangle = 16.89$ cm and the following plots:

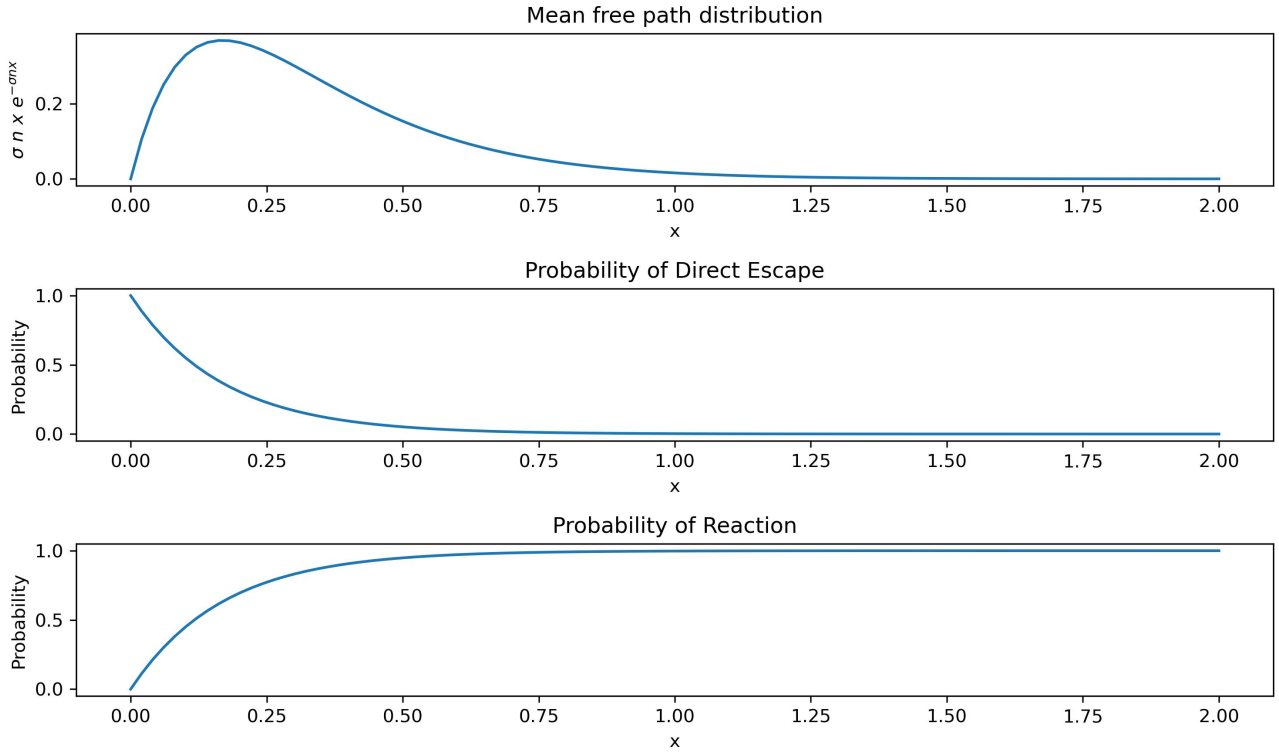

Figure 15: In this image we see the distribution of the mean free path, that has a peak for $0 < x < 0.2$ cm as well as the plots of the probabilities of escape and of interaction, that have a complementary behaviour as expected. (From 7 Neutron Mean Free Path.py).

References

[1] G.W. Griffiths, "Neutron Diffusion" (2018) [University of London]

[2] B.C. Reed, "The Physics of the Manhattan Project" (2015) [Springer]

[3] N.H. Asmar; "Partial Differential Equations With Fourier Series and Boundary Value Problems" (2005) [Dover]