PH052

Optimised Numerical Methods for the Computation of Particles in Fusion Hotspots and Particle Interactions in Magnetised Liner Inertial Fusion (MagLIF)

Boyce Ang Kok Hong (Hong Guo Feng), Ng Yu Heng, Sun Zizhuo PH052

Optimised Numerical Methods for the Computation of Particles in Fusion Hotspots and Particle Interactions in Magnetised Liner Inertial Fusion (MagLIF)

ABSTRACT

Computation of particle interactions is possible. It however requires a lot of computer resources. This project aims to simulate particle interactions in fusion hotspots and Magnetised Liner Inertial Fusion (MagLIF). Using the Barnes-Hut Algorithm, C++, and python, we can create a more efficient simulation through estimations of particle interactions. The simulation was then used to visualise particle movements. Various events such as the initial velocities of particles, coulombs explosion, compression of fusion fuel and general particle interactions in fusion hotspots and MagLIF. Ongoing work is being done to reduce the prevalence of bugs and improve the accuracy of this model even further. This model can allow for the simulation of particle interactions to be more easily accessible regardless of device specifications.

INTRODUCTION

Magnetised Liner Inertial Fusion (MagLIF) introduces a fresh non-laser perspective to Magneto-inertial Fusion, involving a centimetre-scale cylindrical liner filled with fusion fuel (e.g., deuterium gas in Z-pinch experiments) and pre-magnetized axially up to 10-20 T [1]. Through the application of Helmholtz-like coils, MagLIF holds promise for achieving multi-MJ yields and substantial fuel self-heating on upcoming pulsed power facilities [2].

However, particle simulation, despite its feasibility with ample computational resources [3], introduces a challenge. Computing all interactions between pairs of particles, exhibits an $O(N^2)$ complexity, where N denotes the number of particles, imposing constraints on achievable particle counts within simulations. Instead, a more computationally efficient approach is essential to expedite particle trajectory simulations [4].

The Barnes-Hut algorithm is a tree-based solution to simulate interactions within a multitude of bodies. It adeptly addresses the general collisionless N-body problem, demonstrating a time complexity of O(N log N) through the employment of the octree data structure [5], which conceptualizes 3D objects as arrangements of cubes with varying dimensions, yielding enhanced computational efficiency [6].

Coulomb Explosion is the explosion of ions and electrons from a plasma. When an intense electric field, such as a laser, excites the electrons in a solid, the strong electrostatic forces of attraction are overcome, resulting in the explosion [7]. In this project, Coulomb Explosion will be treated as a test of the code functionality. The observations from this project will be qualitatively compared with similar past work.

The main engineering goal for this project is to develop a numerical simulation that can estimate and run particle interaction simulations with minimal computational complexity using the Barnes-Hut algorithm.

MATERIALS AND METHODS

Barnes-Hut Simulation

The Barnes-Hut Simulation runs iteratively on the octree for multiple time steps. To calculate a particle's net force, nodes are traversed from the root. When a node’s centre of mass is far from the particle, its contained bodies are treated as a single particle with “charge centres”.

If the internal node is close, this process recurs for its children nodes. A node is sufficiently far away from the body when its quotient $s/d > \theta$, where s is the width of the node and d is the change in distance of the node. θ represents an arbitrary threshold. The value of θ can be changed to affect the degree of approximation of the simulation. Ideally, the Barnes-Hut Algorithm reduces the computational complexity from $O(N^2)$ to $O(N \log N)$.

Particle Assignment via Octree

To create an octree, we first create the root node, which represents the entire space. When a new particle is added to the tree, we then check if there is an existing particle in the node that encompasses the particle's position. If there is no existing particle in the node, we add the particle to the node. Otherwise, split the root node into 8 3-dimensional octets and move the particles to the appropriate children. We then repeat step 2 until all particles are in the tree.

Initial Conditions Generation

Initial conditions for particles, such as position and velocity must be defined first. Position is determined along a normal distribution:

$$f(x) = \frac{1}{b-a} \tag{1}$$

where σ is the standard deviation and μ is the mean.

Velocity of a particle is determined by the Maxwell-Boltzmann distribution:

$$v = \sqrt{\frac{K_B T}{m}} \times N(0,1) \tag{2}$$

Where K_B is the Boltzmann constant, T is the temperature in Kelvins, m is the mass of the particle and $N(0, 1)$ is a Gaussian random number with mean 0 and standard deviation 1.

Calculation of Electric (E) and Magnetic (B) Fields

The calculation of E-Fields and B-Fields are governed by the Maxwell Equations:

$$\nabla \cdot E = \frac{\rho}{\varepsilon_0} \tag{3}$$

$$\nabla \cdot B = 0 \tag{4}$$

$$\nabla \times E = -\frac{\partial B}{\partial t} \tag{5}$$

$$\nabla \times B = \mu_0 j + \frac{1}{c^2}\frac{\partial E}{\partial t} \tag{6}$$

Unfortunately, directly using the above equations would lead to unnecessary wastage of computer resources.

Optimised Numerical Methods for the Computation of Particles in Fusion Hotspots and Particle Interactions in Magnetised Liner Inertial Fusion (MagLIF)

To simplify the full Maxwell Equations, we will perform quasistatic approximation. We will assume that the changes in electric and magnetic fields are slow relative to the speed of light and ignore the effects of EM waves. In other words, we will assume that the timescales for changes in fields is much larger than the time it takes EM waves to propagate.

(3) can be rewritten as Coulomb's Law.

$$\vec{E} = \frac{1}{4\pi\varepsilon_0}\frac{q\vec{r}}{r^3} \tag{7}$$

Where $\vec{E}$ is the electric field, ε_0 is the permittivity of free space with $\frac{1}{4\pi\varepsilon_0}$ being the Coulomb constant, q is the charge of the external particle and r is the distance between the particles.

(6) can be rewritten as the Biot-Savart Law.

$$\vec{B} = \frac{\mu_0}{4\pi}\frac{I\overrightarrow{dL} \times \hat{r}}{\vec{r}^{\,3}} \tag{8}$$

Where $\vec{B}$ is the magnetic field, μ_0 is the permittivity of free space with $\frac{\mu_0}{4\pi}$ being the Biot-Savart constant, $\overrightarrow{dL}$ is the infinitesimal length of a conductor which carries an electric current *I*, and $\hat{r}$ is the unit vector of r.

From (8), we can then obtain the Biot Savart Law for a Point Charge.

$$\vec{B} = \frac{\mu_0}{4\pi}\frac{q\vec{v} \times \hat{r}}{|\vec{r}|^3} \tag{9}$$

Force, Velocity and Displacement Calculation
To update the position and velocity of the particles, we need to find the force acting on them.

Using the Lorentz Force, we can calculate the amount of force exerted by both the E-Fields and B-Fields.

$$\vec{F} = q\left(\vec{E} + \vec{v} \times \vec{B}\right) \tag{10}$$

Where $\vec{F}$ is force, q is the electric charge of the particle, $\vec{E}$ is the electric field acting on the particle, $\vec{v}$ is the velocity of the particle, and $\vec{B}$ is the magnetic field acting on the particle.

Position is then updated by using the velocity of the particle from the previous timestep:

$$P' = P + \overrightarrow{v_{\|}}(\Delta t) \tag{11}$$

Where P is the position of the particle, P' is the future position of the particle, $\overrightarrow{v_{\|}}$ is the velocity in the axis as/parallel to the axis P is in, and Δt is the difference in time.
To calculate the updated velocity, we simply need to use equations of motion to obtain velocity.

$$v = u + \frac{F}{m}(\Delta t) \tag{12}$$

Where v is the final velocity, u is the initial velocity and m is the mass of the particle.

Energy Conservation

In real life, physical equations are continuous, differential equations. Computers work with discrete data, meaning that continuous data needs to be approximated using time steps. In other words, energy between timesteps will not be conserved. There however exists a solution [8].

$$P = \frac{qB\Delta t}{m} \tag{13}$$

Where *P* refers to the cyclotron frequency times the timestep.

$$A = \begin{pmatrix} 0 & -P_y & P_y \\ -P_z & 0 & P_z \\ P_y & -P_z & 0 \end{pmatrix} \tag{14}$$

The solution for the velocity of the particle can then be written as:

$$v' = (I + A)^{-1}(I - A)v \tag{15}$$

Where v is the initial velocity and v' is the new velocity.

Temporal Discretisation

From (13), we can divide *P* by the timestep, to obtain the cyclotron frequency, *f*:

$$f = \frac{qB}{m} \tag{16}$$

From (16), we can then find the time, *t*, it takes the particle to finish a cyclotron orbit.

$$t = \frac{m}{qB} \tag{17}$$

To accurately compute the trajectory of the particle, Δt must be smaller of equal to the cyclotron orbit period, *t*. Otherwise, the numerical integration scheme would "skip" over some cyclotron orbits, resulting in a loss of energy.

To find the maximum timestep, we first find the ratio of charge to mass of each particle.

Table 1: Mass to Charge Ratio

Particle	Mass to Charge Ratio
Deuterons	2
Electrons	1/1836

Optimised Numerical Methods for the Computation of Particles in Fusion Hotspots and Particle Interactions in Magnetised Liner Inertial Fusion (MagLIF)

Beryllium	$9/2$

We then need to find the smallest charge to mass ratio, $\frac{1}{1836}$, and then multiply it by the strength of the magnetic field, B = 20T. We then obtain the maximum timestep, Δt:

$$\Delta t \leq \frac{1}{1836(20)} \tag{18}$$

$$\Delta t \lessapprox 2.72 \times 10^{-5} \tag{19}$$

This means that any timestep equal to or less than 2.72×10^{-5} s can be used in conjunction with F. Michel's numerical integration scheme. On the other hand, timesteps cannot exceed 2.72×10^{-5} s, otherwise F. Michel's method fails.

"Centre of Charge" Calculation

In order to approximate the particles, we will have to calculate and sum up all the charges, and masses of the points. These values are then to act from the centre of the node.

While there exists the formula to find the centre of mass:

$$R = \frac{1}{M}\sum_{i=1}^{n} m_i r_i \tag{20}$$

where $M = \sum_{i=1}^{n} m_i$ is the total mass of all the particles, m_i is the mass of the i-th particle and r_i is the spatial coordinates of the i-th particle, there is no equivalent formula for the centre of charge as M may equate to 0 as it could result in a division by zero when net charge is neutral. Hence, we must take an alternative approach.

Sums of charges in a non-external node are separated into positive and negative components. Separate positive and negative "centres of charges" are obtained. Due to the separated "centres of charges", even if the net charge of a particular node is 0, the difference in distance between the positive and negative "centres of charges" means that particles still interact with each other despite being in a neutrally charged node. Hence, we rewrite equation (20) as:

$$\rho(r) = \begin{cases} \frac{1}{Q}\sum_{i=1}^{n} q_i r_i, & Q \neq 0 \\ 0, & Q = 0 \end{cases} \tag{21}$$

where $Q = \sum_{i=1}^{n} q_i$ is the total positive/negative charge of all the particles, q_i is the charge of the i-th particle and r_i is the spatial coordinates of the i-th particle.

Liner Current

In order to simulate the presence of the anode and cathode in MagLIF, we will need to give each liner electron a velocity. We can calculate this by using the following equation:

$$I = nevA \tag{22}$$

Where *I* is the current, *n* is the number of electrons per unit volume, *e* is 1.6e-19 C, and *A* is the cross-sectional area.

RESULTS

Due to the nature of the simulation, what is computed is going to merely be a very close estimate of particle interactions in fusion hotspots. Despite this however, these estimates are still good enough for us to form various conclusions on particles interactions in fusion hotspots and particle interactions in MagLIF.

Energy is conserved in the simulation

Due to the usage of timesteps, the standard equations for the calculation of velocity will lead to an explosion in the internal energy of a particle. This means that energy is not conserved.

To determine if the energy of a particle is conserved, we can simply apply a magnetic field and observe a single charged particle in that magnetic field. If energy is conserved, the particle should move in a circle with constant radius. If energy is gained, the particle will spin in an increasing spiral, before leaving the simulation volume.

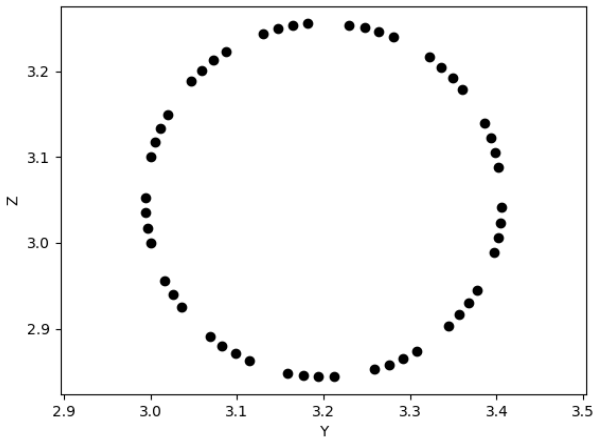

Figure 1: Position of Deuteron in Magnetic Field Across Multiple Timesteps
1 unit represents 10 μm.

Figure 1 shows the particle taking the path of a constant circle. Therefore, we can conclude that our revised computations conserve energy.

Initial velocity of electrons will be 60 times slower than the initial velocity of deuterons

From the simulation, we can obtain the velocities of every simulated particle by averaging it:

$$\bar{v} = \frac{1}{n}\sum_{i=1}^{n} v_i \tag{23}$$

Where n is the number of particles, and v is the velocity of the particles.

Table 2: Average Initial Velocity of Particles

Particle	**Average Velocity at 11000 K (3 s.f.)**

Deuterons	1.10
Electrons	65.9

We observe that the initial velocity of the electrons is 60 times higher than the initial velocity of deuterons. Assuming that the initial kinetic energy of the particles is the same, this difference in initial velocity can be predicted theoretically.

Given that the relative atomic mass is $\frac{1}{1836}$ for an electron and 2 for a deuteron, the initial velocity, *v*, of an electron relative to a deuteron can be calculated using the kinetic energy equation.

$$K.E. = \frac{1}{2}mv^2 \tag{24}$$

$$\frac{1}{2}\left(\frac{1}{1836}\right){v_{electron}}^2 = \frac{1}{2}(2){v_{deuteron}}^2 \tag{25}$$

$$\frac{1}{60}v_{electron} \approx v_{deuteron} \tag{26}$$

Thus, we should expect to see electrons escaping the simulation volume first, followed by the deuterons.

As temperature increases, initial particle velocity increases as well

Table 3: Average Velocity of Particles at Temperature

Particle	Average Velocity at 293 K (3 s.f.)	Average Velocity at 11000 K (3 s.f.)
Deuterons	0.180	1.10
Electrons	10.8	65.9

This can also be predicted theoretically by using the Maxwell-Boltzmann distribution, eq. (2). Thus, as temperature increases, initial particle velocity increases as well.

Electrons rapidly leave the plasma, followed slowly by deuterons

In a Coulomb explosion, lighter particles (e.g., electrons) travel outwards faster than heavier particles (e.g., protons/neutrons).

Optimised Numerical Methods for the Computation of Particles in Fusion Hotspots and Particle Interactions in Magnetised Liner Inertial Fusion (MagLIF)

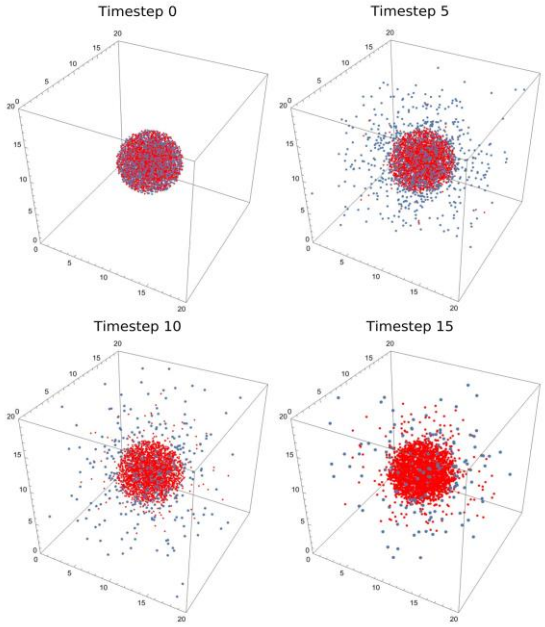

Figure 2: Position of Particles at Various Timesteps
Red represents deuterons and blue represents electrons. Each timestep is 10 ns and each axis unit is 10 μm.

After the electrons have escaped the plasma, Deuterons gain energy due to electrostatic forces of repulsion, allowing the deuterons to approach each other, overcome electrostatic forces of repulsion and undergo fusion.

The beryllium liner collapses

Due to a strong magnetic field which confines the particles, we should expect to see the beryllium liner compress. As the beryllium liner compresses, the electrostatic forces of repulsion between the ions will become stronger. Eventually, this causes the beryllium liner to collapse.

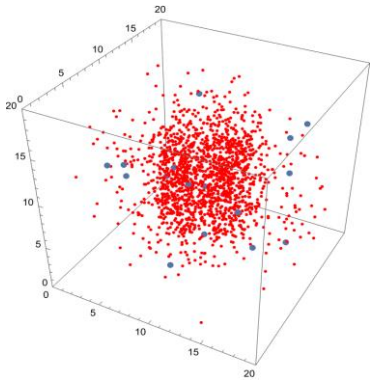

Figure 3: Collapse of Beryllium Liner
Red represents Beryllium ions and blue represents electrons. Each timestep is 10 ns and each axis unit is 10 μm.

Compression of the liner was not reliably observed. This could potentially be caused by computation errors. However, we do observe the collapse of the beryllium liner.

The MagLIF particles are confined.

Optimised Numerical Methods for the Computation of Particles in Fusion Hotspots and Particle Interactions in Magnetised Liner Inertial Fusion (MagLIF)

Due to pressure exerted by the magnetic and electric fields, as well as the beryllium liner on the fusion fuel (deuterons), the fuel should compress. Like the beryllium liner, as the fusion fuel compresses, the electrostatic forces of repulsion between the ions will become stronger. If overcome, this allows for fusion to take place. However, if not enough pressure is exerted on the fuel, the fuel rod will collapse.

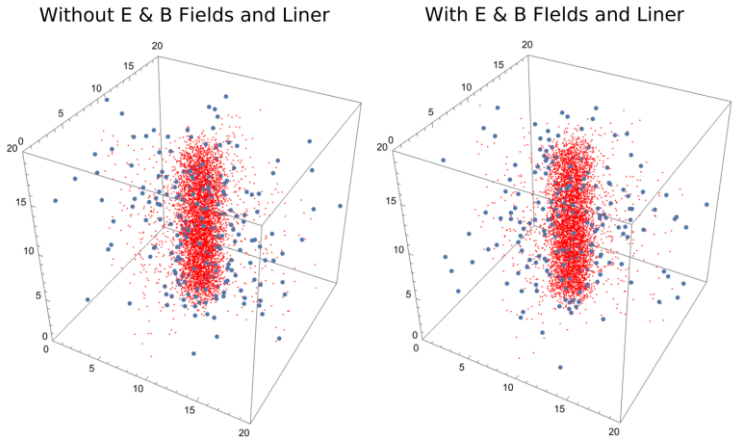

Figure 4: Confinement of Fuel Rod after 70 timesteps
Red represents deuterons and blue represents electrons. Each timestep is 10 ns and each axis unit is 10 μm.

When the liner, E fields and B fields are not present, we observe that the liner is less confined compared to when E fields, B fields and the liner are present. In other words, the fuel rod was confined in the simulation. Due to the lack of reliable compression of the beryllium liner, we were not able to reliably observe compression of the fuel rod.

While the simulation is presently unable to show this, we would like to propose that:
Temperature increases after compression
As the fuel is compressed, it gains kinetic energy in the form of increased motion. Since we are studying plasmas, we will treat kinetic energy and temperature as the same thing. In that case, as kinetic energy increases, the temperature of the plasma increases as well.

Additionally, as deuterons undergo nuclear fusion, they release large amounts of energy in the form of high-energy particles and photons. These particles or photons then deposit their heat into the surrounding plasma, causing the plasma to heat up further.

DISCUSSION

The initial particle distribution and velocity aligned closely with theoretical predictions. The occurrence of Coulomb's explosion indicated a functional simulation. Energy conservation was evident, and although the compression and collapse of the beryllium liner were observed, their reliability was limited. The presence of magnetic and electric fields facilitated particle confinement, a prerequisite for potential fusion.

Although the exact simulation of plasma hotspots could not be achieved, the results that were

produced were of an acceptable accuracy in terms of particle movement and energy distribution. However, owing to limitations of the simulation, conditions were insufficient to initiate fusion.

Experimental observations excluded liner electron velocity due to computation issues. Energy conservation was assumed based solely on charged particle trajectories in a constant magnetic field. The challenge of temporal discretization arose from converting continuous physical equations into computer-compatible discrete data, resulting in omitted particle interactions between time steps. Notably, a time step of 1e8 was chosen as reasonable as seen in (19). Additionally, the simulation did not incorporate Bremsstrahlung, thus overlooking energy loss from particle deflection. The project relied on assumed outcomes due to the absence of empirical results and made assumptions about initial velocities and temperatures (0 m/s, 293 K for liner; 11,000 K for plasma), further restricting result interpretation.

In future work, the lack of observed liner compression, restricted to collapse phenomena, warrants investigation to discern the underlying causes. Efforts should be taken to rectify issues with the computation of liner electron velocity. Additionally, calculating total energy across all time steps would enhance analysis. Advancements in temporal discretization techniques, including the derivation of an ideal time step through incorporation of factors like plasma, cyclotron, and Debye periods, along with particle transit time due to electric fields, should also be taken. Furthermore, potential improvements involve incorporating Bremsstrahlung to account for energy loss due to particle deflection.

ACKNOWLEDGEMENTS

We would like to thank Associate Professor Dr Paul Lee (NTU/NIE) for his guidance, teachings, and weekly Sunday online meetings. We would also like to thank Mr Kong Chiak Wu for his constant support and help throughout the project work cycle.

REFERENCES

[1] UK IF Consortium (n.d.) Laser fusion: UK direct drive collaboration. https://www.inertial-fusion.co.uk/maglif. Last retrieved, 10 August 2023.

[2] D.A. Yager-Elorriaga, M.R. Gomez, D.E. Ruiz, S.A. Slutz, A.J. Harvey-Thompson, C.A. Jennings, P.F. Knapp, P.F. Schmit, M.R. Weis, T.J. Awe, G.A. Chandler, M. Mangan, C.E. Myers, J.R. Fein, B.R. Galloway, M. Geissel, M.E. Glinsky, S.B. Hansen, E.C. Harding, D.C. Lamppa, W.E. Lewis, P.K. Rambo, G.K. Robertson, M.E. Savage, G.A. Shipley, I.C. Smith, J. Schwarz, D.J. Ampleford, K. Beckwith, K.J. Peterson, J.L. Porter, G.A. Rochau and D.B Sinars (2022). An overview of magneto-inertial fusion on the Z machine at Sandia National Laboratories. *Nuclear Fusion*, *62*(4).

[3] L. Greengard and V. Rokhlin (1987). A fast algorithm for particle simulations. *Journal of Computational Physics*, *73*(2), 325–348.

[4] Y. Lin and S. B. Andersson (2021). Computationally efficient application of sequential Monte Carlo expectation maximization to confined single particle tracking. *2021 European Control Conference (ECC)*.

[5] J. Barnes and P. Hut (1986). A hierarchical O(n log n) force-calculation algorithm. *Nature*, *324*(6096), 446–449.

[6] P. Sojan Lal, A. Unnikrishnan, K. Poulose Jacob (1998). Parallel implementation of octtree generation algorithm. *Proceedings 1998 International Conference on Image Processing. ICIP98 (Cat. No.98CB36269)*.

[7] F. Légaré, Kevin F. Lee, I. V. Litvinyuk, P. W. Dooley, S. S. Wesolowski, P. R. Bunker, P. Dombi, F. Krausz, A. D. Bandrauk, D. M. Villeneuve, and P. B. Corkum (2005). Laser Coulomb-explosion imaging of small molecules. *Physical Review A*, *71*(1).

[8] F. Michel (n.d.) Numerical Integration of trajectories in static magnetic fields. *OpenStax CNX*. https://web.archive.org/web/20081220143112/http://cnx.org/content/m12765/1.2/source. Last retrieved, 10 August 2023.

Optimised Numerical Methods for the Computation of Particles in Fusion Hotspots and Particle Interactions in Magnetised Liner Inertial Fusion (MagLIF)

APPENDIX

Appendix A: MAGLIF diagram

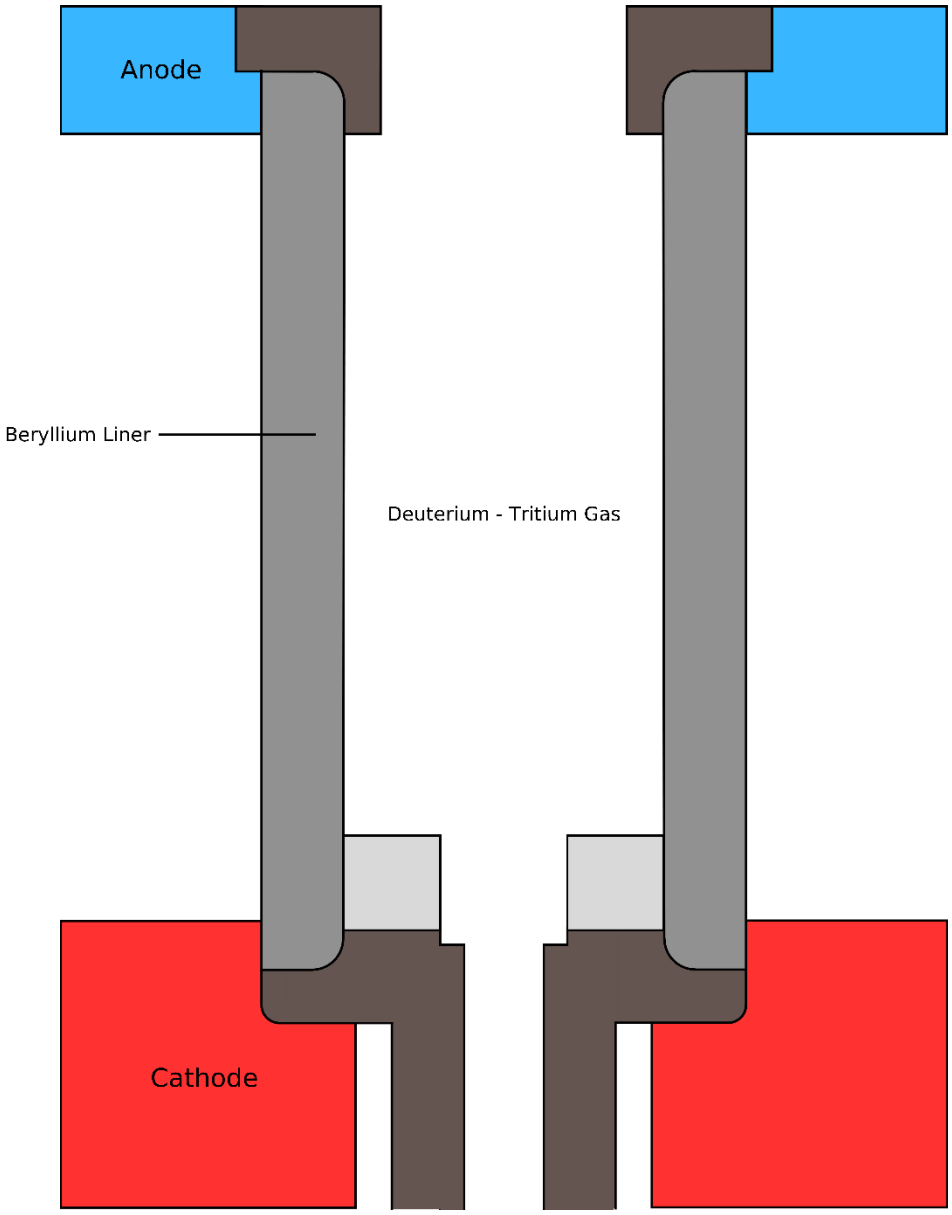

Figure 5: Diagram of MagLIF

Appendix B: Code

https://github.com/23HCI03SMP/Barnes-Hut-Simulation