

- egn. spacing
- fig. spacing
- ✓ wrong use of italic
" " fig/eqn/sec citation
- wrong ref. formal

DESING AND SIMULATION SOFTWARE FOR RHODOTRON TYPE
ELECTRON ACCELERATORS

Correct ALL!

Düzeltilir
tekrar
by
Muhammet Furkan Er
B.S., Physics, Boğaziçi University, 2020

spiralli
gelecek

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in M.S. Physics
Boğaziçi University

2023

TABLE OF CONTENTS

ABSTRACT	iii
ÖZET	iv
LIST OF FIGURES	vi
LIST OF TABLES	vii
LIST OF SYMBOLS	viii
LIST OF ACRONYMS/ABBREVIATIONS	ix
1. INTRODUCTION AND THEORY	1
1.1. Accelerating Charged Particles	2
1.1.1. Relation between momentum and acceleration	2
1.1.2. Lorentz Force	4
1.1.3. Relativistic Lorentz Force	4
1.1.4. Acceleration caused by Lorentz force	5
1.2. Particle Accelerators	6
1.2.1. Acceleration Cavities	7
1.2.2. Bending Magnets	9
1.2.3. Key Concepts	9
1.2.3.1. Resonance Frequency	9
1.2.3.2. Bunch	9
1.2.3.3. Phase Stability	9
1.2.3.4. Phase Lag	10
1.2.3.5. Shunt Impedance	10
1.2.4. Rhodotron Accelerator <i>(JC)</i>	10
1.2.4.1. Acceleration cycle of Rhodotron	11
1.2.4.2. Cavity of a Rhodotron	11
1.3. Basic Concepts in Programming	21
1.3.1. Clock Cycle	21
1.3.2. Concurrency	22
1.3.3. Object Oriented Programming	23

2. TOOLS	24
2.1. Simulation	24
2.2. Available Tools	24
2.2.1. Poisson Superfish	24
2.2.2. CST Studio Suite	25
2.2.3. ROOT	25
2.2.4. gnuplot	25
2.3. Algorithms	26
2.3.1. Leapfrog	26
2.3.2. Runge Kutta	27
3. DESIGN	29
3.1. Cavity Design	29
3.2. Magnet Design	35
3.2.1. $n\lambda$ Technique	36
3.2.2. L_{out} Parameter Sweep	38
3.3. Initial Design at KAHVELab	39
4. SIMULATION	47
4.1. Proof of Concept	47
4.2. Intermediate Versions	49
4.2.1. L_{out} Optimization For Single e^-	49
4.2.2. ϕ_{lag} Optimization For Bunches	49
4.2.3. Simulation in 3D	51
4.2.4. Acceleration in Magnetic Field	54
4.2.4.1. RK4-1	54
4.2.4.2. RK4-2	54
4.2.5. Acceleration in Electric Field	58
4.2.6. Multithreading	63
4.3. Graphical User Interface	66
4.3.1. Simulation Frame	67
4.3.2. Configuration Frame	68
4.3.3. Render Frame	69

check (ALL egn.s)
spacing before and after egn.s

1.1. Accelerating Charged Particles

1.1.1. Relation between momentum and acceleration

In classical mechanics, Newton's second law defines force, \vec{F} , and relation

$$\vec{F} = \frac{d\vec{p}}{dt} = m \frac{d\vec{v}}{dt} = m\vec{a}, \quad (1.1)$$

where \vec{p} is the momentum, $m\vec{v}$ of the particle.

In special relativity however, relativistic momentum is defined as $\vec{p} = \gamma m_0 \vec{v}$, where

$$\gamma = \frac{1}{\sqrt{1 - v^2/c^2}} = \frac{1}{\sqrt{1 - \beta^2}} = \frac{1}{\sqrt{1 - (\vec{v} \cdot \vec{v})/c^2}} = \frac{1}{\sqrt{1 - \vec{\beta} \cdot \vec{\beta}}} \quad (1.2)$$

is the Lorentz Factor.

Considering these two statements, we can find the relation between momentum and acceleration as

$$\frac{d\vec{p}}{dt} = m_0 \frac{d(\gamma \vec{v})}{dt} = m_0 \left\{ \frac{d\gamma}{dt} \vec{v} + \gamma \frac{d\vec{v}}{dt} \right\}, \quad (1.3)$$

$$\frac{d\gamma}{dt} = \gamma^3 \vec{\beta} \cdot \frac{d\vec{\beta}}{dt} = \frac{\gamma^3}{c} \vec{\beta} \cdot \vec{a}, \quad (1.4)$$

$$\frac{d\vec{p}}{dt} = m_0 \left\{ \frac{\gamma^3}{c} (\vec{\beta} \cdot \vec{a}) \vec{v} + \gamma \frac{d\vec{v}}{dt} \right\}. \quad (1.5)$$

Therefore force can be written as

$$\vec{F} = \gamma m_0 \{ \vec{a} + \gamma^2 (\vec{\beta} \cdot \vec{a}) \vec{\beta} \}. \quad (1.6)$$

Eqn.s are
not cited
correctly
check ALL

It is clear that acceleration is not necessarily parallel to the force. To start separating the parallel and perpendicular components relative to $\vec{\beta}$, we can find $\vec{a}_{||}$ and $\vec{F}_{||}$ as

$$\vec{a}_{||} = \frac{(\vec{a} \cdot \vec{\beta})}{\beta^2} \beta, \quad \vec{F}_{||} = \frac{(\vec{F} \cdot \vec{\beta})}{\beta^2} \beta, \quad (1.7)$$

where

$$\begin{aligned} \vec{F} \cdot \vec{\beta} &= \gamma m_0 \{ \vec{a} \cdot \vec{\beta} + \gamma^2 (\vec{\beta} \cdot \vec{a}) \beta^2 \} \\ &= \gamma m_0 (\vec{a} \cdot \vec{\beta}) \{ \gamma^2 \beta^2 + 1 \}. \end{aligned} \quad (1.8)$$

Then, $\gamma^2 \beta^2 + 1 = \gamma^2$ and *Equation 1.7* can be used to achieve

$$\begin{aligned} \vec{F} \cdot \vec{\beta} &= m_0 \gamma^3 (\vec{a} \cdot \vec{\beta}), \\ \vec{F}_{||} &= \frac{(\vec{F} \cdot \vec{\beta})}{\beta^2} \beta \\ &= m_0 \gamma^3 \frac{(\vec{a} \cdot \vec{\beta})}{\beta^2} \beta \\ &= m_0 \gamma^3 \vec{a}_{||}. \end{aligned} \quad (1.9)$$

Therefore from *Equations 1.6 and 1.9*, force can be rearranged as

$$\begin{aligned} \vec{F} &= m_0 \gamma^3 \vec{a}_{||} \beta^2 + m_0 \gamma \vec{a} \\ &= m_0 \gamma^3 \vec{a}_{||} \beta^2 + m_0 \gamma \{ \vec{a}_{||} + \vec{a}_{\perp} \} \\ &= m_0 \vec{a}_{||} \gamma \{ \gamma^2 \beta^2 + 1 \} + m_0 \gamma \vec{a}_{\perp} \\ &= m_0 \vec{a}_{||} \gamma^3 + m_0 \gamma \vec{a}_{\perp} \\ &= \vec{F}_{||} + m_0 \gamma \vec{a}_{\perp}. \end{aligned} \quad (1.10)$$

We finally have two separate equations which are in similar form with *Equation 1.1*,

$$\vec{F}_{||} = \gamma^3 m_0 \vec{a}_{||}, \quad \vec{F}_{\perp} = \gamma m_0 \vec{a}_{\perp}. \quad (1.11)$$

1.1.2. Lorentz Force

Force acting on a charged particle moving in electromagnetic fields is called Lorentz Force and is given by

$$\frac{d\vec{p}}{dt} = \vec{F}_L = q(\vec{E} + \vec{v} \times \vec{B}), \quad (1.12)$$

where the q is the charge and \vec{v} is the velocity of the particle.

1.1.3. Relativistic Lorentz Force

Similar to non-relativistic version, relativistic Lorentz Force is given by the 4-vector equality

$$\frac{dp^\mu}{d\tau} = qF^{\mu\nu}u_\nu, \quad (1.13)$$

where $d\tau = dt/\gamma$, greek letters are Lorentz indices and p , F , u are

$$p^\mu = \begin{bmatrix} W/c \\ p_x \\ p_y \\ p_z \end{bmatrix}, \quad F^{\mu\nu} = \begin{bmatrix} 0 & -E_x/c & -E_y/c & -E_z/c \\ E_x/c & 0 & -B_z & B_y \\ E_y/c & B_z & 0 & -B_x \\ E_z/c & -B_y & B_x & 0 \end{bmatrix}, \quad u_\nu = \gamma \begin{bmatrix} c \\ -v_x \\ -v_y \\ -v_z \end{bmatrix},$$

where W is the energy of the particle and γ is the Lorentz factor mentioned in the previous section.

For $\mu = 0$, we have the time component of the equation as

$$\frac{\gamma}{c} \frac{dW}{dt} = \frac{q\gamma \vec{E} \cdot \vec{v}}{c} = \frac{q\gamma}{c} \vec{E} \cdot \frac{d\vec{r}}{dt}, \quad (1.14)$$

$$\frac{dW}{dt} = q \frac{\vec{E} \cdot d\vec{r}}{dt}. \quad (1.15)$$

This is the definition of work done by an electric field. For $\mu = 1, 2, 3$, we have the spacial components

$$\frac{d\vec{p}}{d\tau} = \gamma \frac{d\vec{p}}{dt} = q\gamma(\vec{E} + \vec{v} \times \vec{B}),$$

is missing

which simplifies to non-relativistic Lorentz Force in *Equation 1.12.*

1.1.4. Acceleration caused by Lorentz force

Due to the nature of the cross product, Lorentz force caused by a magnetic field is always perpendicular to the velocity of the particle. Therefore the acceleration of the magnetic field is straightforward,

$$\vec{F}_B = \vec{F}_\perp = \gamma m_0 \vec{a}_\perp = \gamma m_0 \vec{a}_B.$$

The same thing cannot be said about electric field however. It can create force in any direction with respect to velocity. Therefore, we have the equality

$$\vec{a}_\parallel = \frac{q}{\gamma^3 m_0} \vec{E}_\parallel, \quad \vec{a}_\perp = \frac{q}{\gamma m_0} \{\vec{E}_\perp + \vec{v} \times \vec{B}\}. \quad (1.16)$$

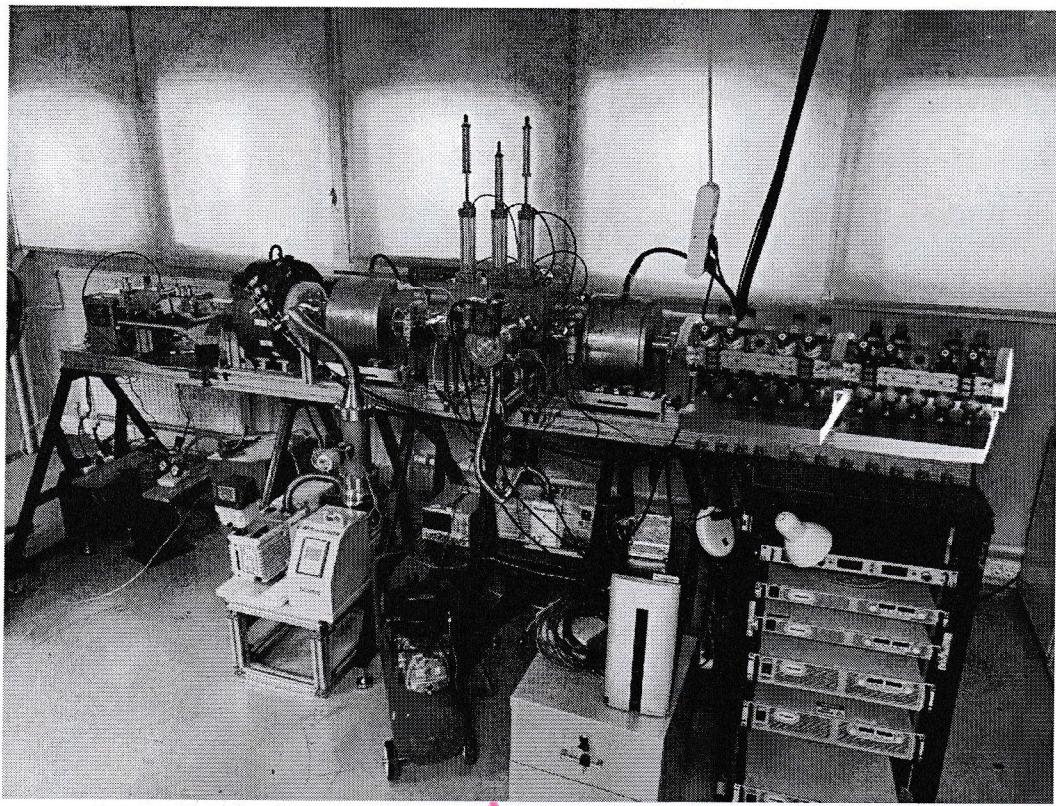
Acceleration due to electric field can be simplified as

$$\begin{aligned} \vec{a}_{\{B=0\}} = \vec{a}_E &= \vec{a}_\parallel + \vec{a}_{\perp\{B=0\}} \\ &= \frac{q}{m_0 \gamma} \left\{ \frac{\vec{E}_\parallel}{\gamma^2} + \vec{E}_\perp \right\} \\ &= \frac{q}{m_0 \gamma} \left\{ \{1 - \beta^2\} \vec{E}_\parallel + \vec{E}_\perp \right\} \\ &= \frac{q}{m_0 \gamma} \left\{ \vec{E}_\parallel + \vec{E}_\perp - \beta^2 \vec{E}_\parallel \right\} \\ &= \frac{q}{m_0 \gamma} \left\{ \vec{E} - \beta^2 \vec{E}_\parallel \right\}. \end{aligned} \quad (1.17)$$

Using the fact that $\vec{E}_\parallel = \vec{\beta}(\vec{E} \cdot \vec{\beta})/\beta^2$, we finally have

$$\vec{a}_E = \frac{q}{\gamma m_0} \left\{ \vec{E} - \vec{v} \frac{(\vec{E} \cdot \vec{v})}{c^2} \right\}, \quad \vec{a}_B = \frac{q}{\gamma m_0} (\vec{v} \times \vec{B}), \quad (1.18)$$

where \vec{a}_E consists of two components in the direction of \vec{E} and \vec{v} . The component in the direction of \vec{v} depends on the $\vec{E} \cdot \vec{v}$, resulting in $\vec{a}_E \parallel \vec{E}$ if $\vec{E} \perp \vec{v}$.



↑ Should be a single line
Figure 1.1. A linear proton accelerator in KAHVELab [2].

1.2.1. Acceleration Cavities

Check ALL figures

Radiofrequency (RF) cavities, also known as accelerating cavities or resonant cavities, are key components in particle accelerators. These cavities generate strong electromagnetic fields at specific frequencies to accelerate charged particles through clever engineering.

RF cavities are typically hollow metallic structures made of or coated with high-conductivity materials such as copper. The cavity is often cylindrical or spherical in shape, and its inner surface is polished to minimize energy losses through resistive heating. The RF cavity is designed to be resonant, meaning that it naturally amplifies the electric fields at its resonant frequency. The resonant frequency is determined by the cavity's dimensions and the speed of light in the cavity material.

To achieve efficient energy transfer to the particles, the RF cavity is driven by an external RF power source operating at the resonant frequency. The power source supplies radiofrequency energy to the cavity, which causes the electric fields inside the cavity to oscillate at the desired frequency. These oscillating fields then transfer energy to the passing particles, increasing their kinetic energy by pushing and pulling on the charged particles as they pass through the cavity.

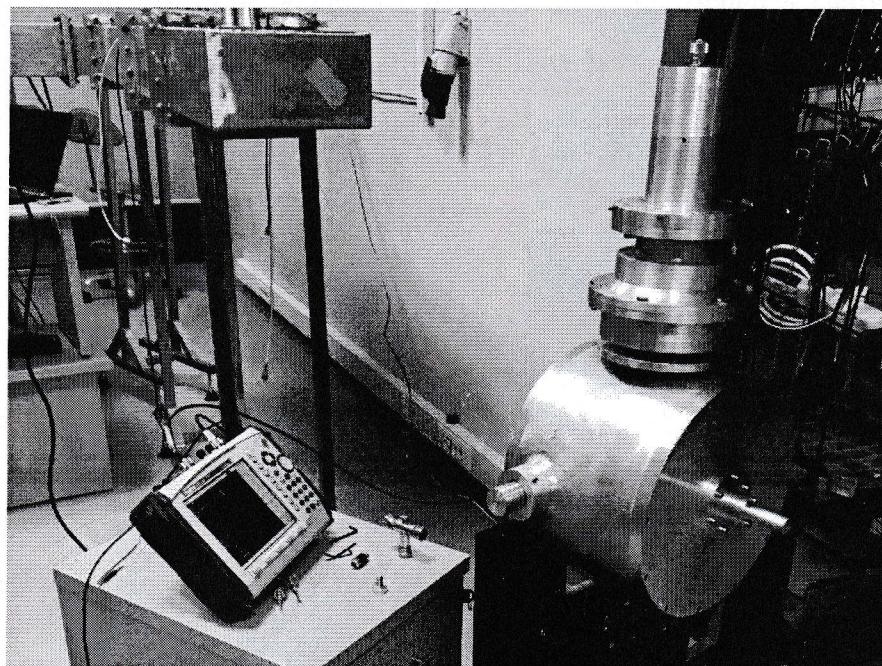


Figure 1.2. An RF cavity used in KAHVELab.

In addition to accelerating the particles, RF cavities are often designed to provide focusing forces. By carefully shaping the cavity and adjusting the electromagnetic fields, the particles can experience focusing effects as they pass through the cavity. This helps to maintain a tight and controlled beam. To ensure efficient acceleration, it is essential to maintain phase stability. This means that the particles should experience the strongest electric fields at the correct time during their passage through the cavity. Precise timing and synchronization of the RF power source with the particle beam are crucial to achieve phase stability and maximize energy transfer.

achieving efficient particle acceleration. Maintaining phase stability is essential to prevent particles from becoming out of phase as they travel through accelerator structures, ensuring that they receive the correct energy boosts and interact predictably with detectors. Deviations in phase stability can lead to particle loss and decreased beam quality.

1.2.3.4. Phase Lag. Phase lag refers to the time delay between the oscillations of two interacting waveforms or particles. It describes the difference in phase angles within their respective cycles, between two signals.

1.2.3.5. Shunt Impedance. The shunt impedance of an RF accelerator is a measure of the efficiency at which the accelerator can transform the supplied RF power into acceleration. It is defined as

$$Z_s = \frac{V_{acc}^2}{P_{diss}}, \quad (1.20)$$

where V_{acc} is the accelerating potential in which the particle is subjected to, P_{diss} is the power dissipated on the cavity walls. An example *shunt impedance* calculation can be found in *Section 1.2.4.2*.

1.2.4. Rhodotron Accelerator

not italic

Rhodotron Accelerator is a type of particle accelerator that was proposed by Jacques POTTIER in 1989 [1]. First prototype was built at CEA Saclay later in 1992 [3]. It is named after the greek word for rose, *rhodos*, due to the shape of the design [4].

The design of a rhodotron mainly consists of a coaxial cylindrical RF cavity and bending magnets surrounding it. RF cavity is fed by an external RF source, accelerating the electrons entering from an attached electron injector.

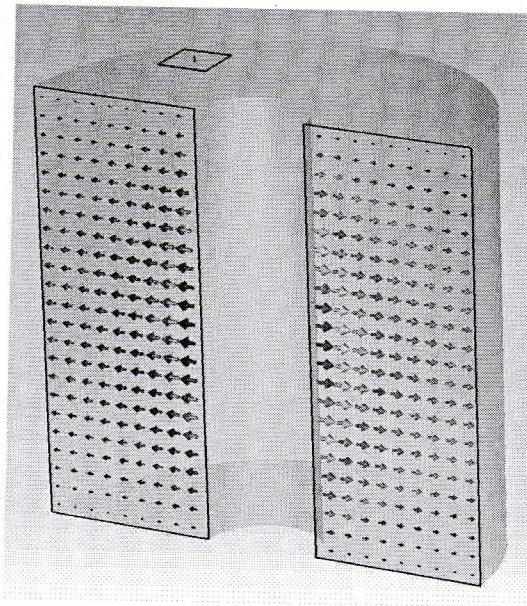
1.2.4.1. Acceleration cycle of Rhodotron. Electrons undergo four different stages inside the accelerator. They are accelerated between the cylindrical plates and are shielded from the changing RF field while inside the inner cylinder and outside the cavity. These stages are explained further below.

- *First Acceleration:* Electrons in the rhodotron cavity are accelerated by the electric field created between two coaxial cylinders, towards the inner cylinder when they are ejected into the cavity (*Figure 1.7*).
- *Inner Shielding:* While inside the inner cylinder, the cylinder acts as a faraday cage and shields the electrons inside while the electric field is being reversed (*Figure 1.8*).
- *Second Acceleration:* Once the electrons leave the inner cylinder, they accelerate towards the outer cylinder by the reversed electric field until they leave the cavity (*Figure 1.9*).
- *Recirculating Magnets:* After leaving the cavity, an electromagnet placed in their path steers the electrons back into the cavity in which time the electric field changes the direction again (*Figure 1.10*).

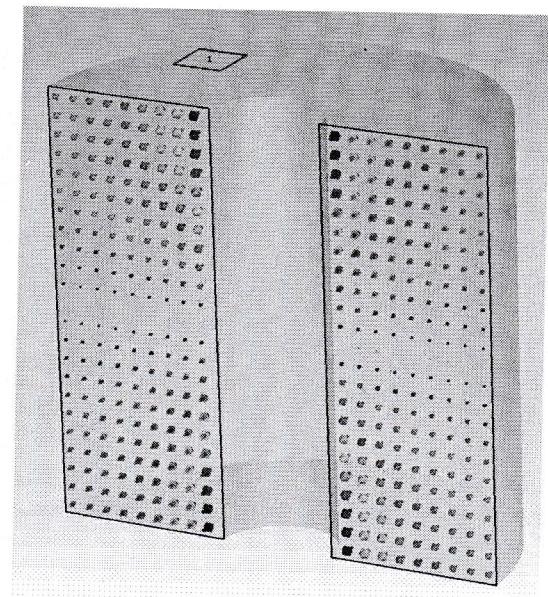
This cycle can be repeated as long as real world constraints such as; placements and dimensions of the electromagnets, power requirements due to increasing magnetic field in order for sharper turns, can be overcome. After the desired amount of cycles, also called passes, has been completed, the electrons exit the accelerator.

This process is explained further in the *Figures 1.7, 1.8, 1.9 and 1.10* where T is the period of the electric field.

1.2.4.2. Cavity of a Rhodotron. Coaxial design of the cavity concentrates the electric field, while the magnetic field diminishes in the middle of the cylinders. Therefore the electrons are injected and accelerated in the plane of zero magnetic field where the electric field is strongest (*Figure 1.3*).



(a) \vec{E} field



(b) \vec{B} field

Figure 1.3. \vec{E} and \vec{B} eigenmode field distributions inside a coaxial cavity.

For a cavity defined by the volume between two coaxial cylinders of equal lengths (h) with radii of R_1 and R_2 , where $R_1 < R_2$, located at the origin (Figure 1.4), first eigenmode solution of the E and B fields are [1]

$$E = \frac{E_0}{r} \cos\left(\frac{\pi z}{h}\right) \sin(\omega t + \phi), \quad (1.21)$$

$$B = \frac{B_0}{r} \sin\left(\frac{\pi z}{h}\right) \cos(\omega t + \phi), \quad (1.22)$$

where $\omega = 2\pi f$, f is the resonance frequency.

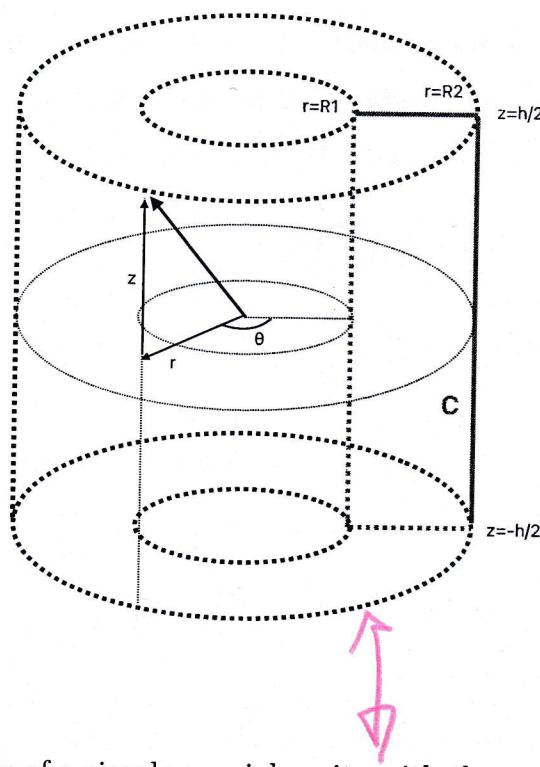


Figure 1.4. Illustration of a simple coaxial cavity with the curve \mathbf{C} in *Equation 1.24*.

Because acceleration potential is located on the $z = 0$ plane and $\vec{\mathbf{E}} \parallel \hat{r}$, V_{acc} can be found by

$$\begin{aligned}
 V_{acc} &= 2 \int_{R_1}^{R_2} |E|^2 dr \\
 &= 2E_0 \int_{R_1}^{R_2} \frac{dr}{r} \\
 &= 2E_0 \ln\left(\frac{R_2}{R_1}\right).
 \end{aligned} \tag{1.23}$$

Dissipated power P_{diss} , on the other hand, can be calculated as [5]

$$\begin{aligned}
 P_{diss} &= \frac{1}{2} \int \int \rho_s |H_{||}|^2 dA \\
 &= \frac{\rho_s}{2\mu_0^2} \int_0^{2\pi} \int_C |B_{||}|^2 r ds d\theta.
 \end{aligned} \tag{1.24}$$

where ρ_s is the areal skin resistivity ($\rho_s \approx 2.51 \times 10^{-7} f^{1/2}$ for copper [1]), $B = \mu H$, μ is permeability, μ_0 is the vacuum permability. Integral curve \mathbf{C} is defined as the

Table 1
Optimized characteristics

P	R_2 (m)	R_1/R_2	Z_{se} ($M\Omega$)	Z_{sp} ($M\Omega$)
1	0.27λ	1/4	$5.77\lambda^{1/2}$	$4.9\lambda^{1/2}$
2	0.5λ	1/7	$10.4\lambda^{1/2}$	$8.83\lambda^{1/2}$



Figure 1.5. Optimized characteristics of a rhodotron cavity [1].

Here, p is the integer multiplier in the equation ($l = p\lambda$) mentioned above, R_1 is the radius of the inner cylinder, R_2 is the radius of the outer cylinder, Z_{se} is effective shunt impedance, Z_{sp} is practical shunt impedance which was taken to be $0.85Z_{se}$. Typically, phase lag ϕ is taken as 15° [1].

Considering $Z_{sp} \propto \lambda^{1/2}$, $\Delta E \propto \lambda^{1/4}$, $V \propto \lambda^3$, where V is the volume of the cavity, implementing the $p = 1$ design in Figure 1.5 is much more space efficient.

Table 2
Energy W (MeV) for $P = 100$ kW and $f = 130$ MHz

R_2 (m)	2	3	4	5	6	7	8	9	10	11	12
0.62	1.8	2.7	3.6	4.5	5.4	6.3	7.2	8.1	9	9.9	10.8
1.15	2.4	3.6	4.8	6	7.2	8.4	9.6	10.8	12	13.2	14.4



Figure 1.6. Energy of a synchronous electron after each pass for both $p = 1$ and $p = 2$ [1].

Total energy gain after n passes, ΔE_n , can then be found by Equation 1.36, taking $\phi = 15^\circ$, $p = 1$, i.e $R_2 = 0.27\lambda$, P in W , λ in m as

$$\Delta E_n \approx 2.14\lambda^{1/4}P^{1/2}n \text{ keV.} \quad (1.37)$$

1.3. Basic Concepts in Programming

1.3.1. Clock Cycle

In programming, a clock cycle refers to a fundamental unit of time measurement used in computer systems. It represents the basic rhythm or timing mechanism of a computer's central processing unit (CPU) and is typically measured in terms of the CPU's clock speed, expressed in hertz (Hz).

A clock cycle represents one complete pulse or oscillation of the CPU's clock signal. It serves as a synchronization mechanism, coordinating the execution of instructions and the timing of various operations within the CPU. Each clock cycle is associated with a specific duration, which is determined by the clock speed of the CPU.

The concept of clock cycles is often used when considering the performance and efficiency of algorithms and code. Since the execution time of instructions is influenced by the number of clock cycles required, minimizing the number of clock cycles needed for a program or algorithm can lead to faster and more efficient code execution. Table below shows the amount of clock cycles required for some mathematical calculations in various processors.

Note that *division* is by far the most time consuming basic mathematical operation between addition, subtraction and multiplication. Although not discussed in the following chapters, mathematical operations were reduced to addition, subtraction and multiplication when found to be possible while developing computationally intensive software in *Chapter 4*.

MEASURED CPU TIMES OF BASIC MATHEMATICAL OPERATIONS

OPERATION	DOUBLE PRECISION					QUADRUPLE PRECISION	
	i486	Pentium	PA-7000	μ SPARC-II	Alpha	PA-7000	μ SPARC-II
$x = y \dots$	0.46	0.24	1.00	1.00	0.41	5.1	2.8
$x = y + z \dots$	0.91	1.00	1.00	1.00	1.00	131	11.3
$x = yz \dots$	1.09	1.00	1.00	1.00	1.00	457	99.5
$x = y/z \dots$	2.81	4.49	4.92	3.32	6.59	1100	168
$x = y^{1/2} \dots$	12.0	19.6	4.95	24.4	38.5	1100	149
$x = \sin y \dots$	19.8	23.3	49.8	22.0	18.5	9510	1740
$x = \log y \dots$	20.2	23.9	47.8	31.9	22.0	5420	1240
$x = \tan^{-1} y \dots$	19.8	27.3	46.3	37.3	23.5	5930	1270
$x = \tan y \dots$	18.0	28.3	63.2	30.0	35.1	8450	2040
$x = \exp y \dots$	24.9	32.7	42.0	19.0	23.2	6560	1370

Figure 1.11. Amount of clock cycle for mathematical operations in various processors [6].

1.3.2. Concurrency

Concurrency in programming refers to the ability of a program to execute multiple tasks or processes simultaneously. It allows different parts of a program to make progress independently, potentially improving performance, responsiveness, and resource utilization.

Concurrency in single core can be achieved by implementing clever scheduling of list of operations called *threads*. This results in non-blocking execution of multiple threads, but only one thread would be executed at any given time. True concurrency on the other hand, can only be achieved by using multiple cores. Each core would be able to execute one thread at any time.

Programs that implement concurrency using multiple cores executes more operations per clock cycle. However, this does not directly lead to an improved performance due to the heavy burden of scheduling and managing multiple threads.

Threads utilize *mutexes*, short for "mutual exclusion," to synchronize and manage

2. TOOLS

For the purpose of designing, enhancing, and optimizing particle accelerators, certain tools can be employed. The following sections will explore several of the software and algorithms that were put to use.

2.1. Simulation

A simulation software is a computer program or tool that enables the creation and execution of simulations to model and analyze real-world systems or processes. It allows users to replicate the behavior, interactions, and outcomes of the system or process under study, providing insights and predictions that can be valuable for decision-making, optimization, or understanding complex phenomena.

Simulation software provides a virtual environment where users can define the parameters, variables, and rules of the system being simulated. The software then uses mathematical and physical models, algorithms, and computational techniques to simulate the behavior of the system over time.

2.2. Available Tools

2.2.1. Poisson Superfish

Poisson Superfish is a software package developed by the *Los Alamos National Laboratory* for the design and analysis of electromagnetic fields, particularly those related to particle accelerators and high-energy physics experiments [7]. It is commonly used in the field of accelerator physics to simulate and optimize the behavior of charged particle beams as they pass through various electromagnetic structures, such as cavities and magnets.

2.3. Algorithms

2.3.1. Leapfrog

The Leapfrog method is a numerical method commonly used to solve ordinary differential equations (ODEs) that involve second-order time derivatives. Such an ODE can be written as

$$\ddot{x} = \frac{d^2x}{dt^2} = f(x). \quad (2.1)$$

The Leapfrog method is a variant of the finite difference method, and it approximates the solution of an ODE by discretizing both time and space. The method gets its name from the way it calculates the values of the solution at each time step, which resembles a "leapfrogging" motion. It is a simple and efficient algorithm that is often used in simulations of physical systems, such as celestial mechanics or molecular dynamics. The idea is straight forward; in the time interval Δt ,

$$a(t_0) = f(x_0), \quad (2.2)$$

$$x(t_0 + \Delta t) = x(t_0) + v(t_0)\Delta t + a(t_0) \frac{\Delta t^2}{2}, \quad (2.3)$$

$$v(t_0 + \Delta t) = v(t_0) + \{a(t_0) + a(t_0 + \Delta t)\} \frac{\Delta t}{2}. \quad (2.4)$$

For more stability, this version can be rearranged to what is called 'kick-drift-kick' form as

$$v(t_0 + \Delta t/2) = v(t_0) + a(t_0) \frac{\Delta t}{2}, \quad (2.5)$$

$$x(t_0 + \Delta t) = x(t_0) + v(t_0 + \Delta t/2)\Delta t, \quad (2.6)$$

$$v(t_0 + \Delta t) = v(t_0 + \Delta t/2) + a(t_0 + \Delta t) \frac{\Delta t}{2}. \quad (2.7)$$

This version provides more time resolution to our calculation; however, it increases the number of calculations needed by about 50%.

2.3.2. Runge Kutta

The Runge-Kutta methods, named after the German mathematicians Carl Runge and Martin Wilhelm Kutta [10] [11], family of numerical methods used to solve ordinary differential equations (ODEs) that are in the form as

$$\frac{dy}{dx} = f(x, y). \quad (2.8)$$

The basic idea behind the Runge-Kutta method is to approximate the solution of an ODE by taking small steps and using a weighted average of function evaluations at different points within each step as

$$y_{n+1} = y_n + \delta x \sum_{i=1}^m b_i k_i, \quad x_{n+1} = x_n + \delta x, \quad (2.9)$$

where

$$k_i = f\left(x_n + c_i \delta x, y_n + \delta x \sum_{j=1}^{i-1} a_{ij} k_j\right). \quad (2.10)$$

These Equations 2.9 and 2.10 define the family of methods. To specify a particular method, order m , coefficients a_{ij} , b_i and c_i should be provided. The coefficients of any Runge-Kutta method can be visualized by a tableau called *Butcher Tableau* which can be observed in Figure 2.1.

	0				
c_1		a_{21}			
c_2		a_{31}	a_{32}		
\vdots			\dots		
c_m	a_{m1}	a_{m2}	\dots	$a_{m,m-1}$	
	b_1	b_2	\dots	b_m	

Figure 2.1. Butcher Tableau.

The simplest Runge-Kutta method is the *Euler's method* with the following *Butcher tableau*,

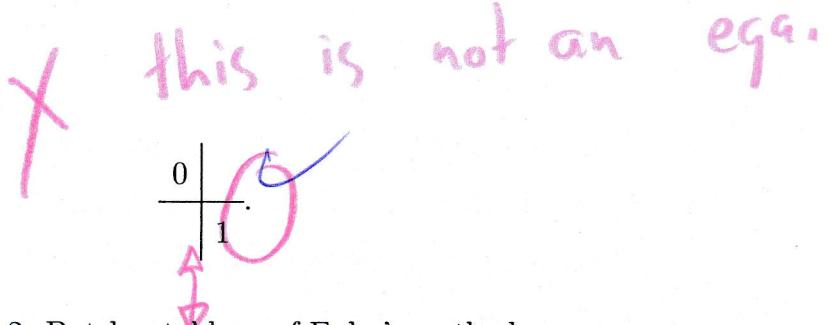


Figure 2.2. Butcher tableau of Euler's method.

The most commonly used version of the Runge-Kutta method is the fourth-order Runge-Kutta method, also known as RK4. The RK4 method involves four function evaluations per step and has an error term that is proportional to the step size raised to the fifth power. Two of the most used *Butcher tableaus* for RK4 given in *Figure 2.3*. The second tableau in *Figure 2.3* is called the "*3/8 rule*". Its main advantage is that its error coefficients are smaller than the other. But it costs more floating point operations per step. Resulting in slower calculations.

0				0
1/2	1/2			
1/2	0	1/2		
1	0	0	1	
1/6	1/3	1/3	1/6	1/8
0	1/3			
1/3	1/3			
2/3	-1/3	1		
1	1	-1	1	
1/8	3/8	3/8	1/8	

Figure 2.3. Butcher Tableau for RK4.

3. DESIGN

In this chapter, the primary design factors of rhodotron-type accelerators are examined, along with an exploration of the design created within the *KAHVELab*. Rhodotron-type accelerators are composed of two principal components: a coaxial acceleration cavity and recirculating magnets. The cavity's design is of utmost significance for attaining resonance and improving beam-RF interactions, while the magnet design focuses on preserving the beam's phase stability.

3.1. Cavity Design

The first and the most important design parameter for a cavity is the operating RF frequency. After an operating RF frequency is set and the desired $R1/R2$ relation in *Figure 1.5* is selected, design parameters of acceleration plane of the cavity is fully determined.

By following the cylindrical design mentioned in *Section 1.2.4*, the only main design parameter remaining is the height of the cylindrical cavity. This parameter can be found using the constraint mentioned in *Section 1.2.1*; the fact that operating RF frequency must be equal to resonant frequency of the cavity. For simple coaxial cavity, the height should be $\lambda/2$, where λ is the wavelength of the external RF supply [1]. Simulation tools such as *CST Studio*, *Poisson SUPERFISH* can be used to confirm this condition.

In the following examples, $p = 1$ from *Figure 1.5* was used, and it will be the focus of all further calculations. Using $f_{RF} = 107.5$ MHz & $f_{RF} = 180$ MHz for comparison:

$$\begin{aligned}
 f_{107.5} &= 107.5 \text{ MHz}, & f_{180} &= 180 \text{ MHz}, \\
 \lambda_{107.5} &= \frac{c}{f_{107.5}} = 2.789m, & \lambda_{180} &= \frac{c}{f_{180}} = 1.666m, \\
 R_2 &= 0.27 \times \lambda_{107.5} = 0.753m, & R_2 &= 0.27 \times \lambda_{180} = 0.450m, \\
 R_1 &= \frac{R_2}{4} = 0.188m, & R_1 &= \frac{R_2}{4} = 0.113m, \\
 h &= \frac{\lambda}{2} = 1.394m. & h &= \frac{\lambda}{2} = 0.833m.
 \end{aligned} \tag{3.1}$$

In the *figures 3.1, 3.2 and 3.3*, Poisson Superfish and CST simulations of two cavities defined by *Equation 3.1* can be observed.

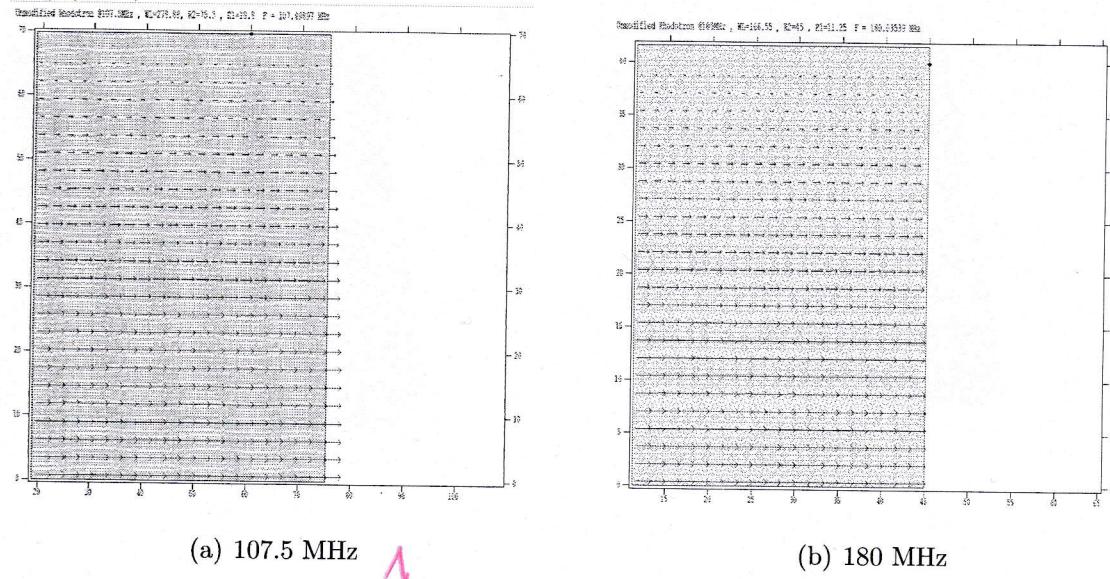


Figure 3.1. Poission Superfish results for *Equation 3.1*.

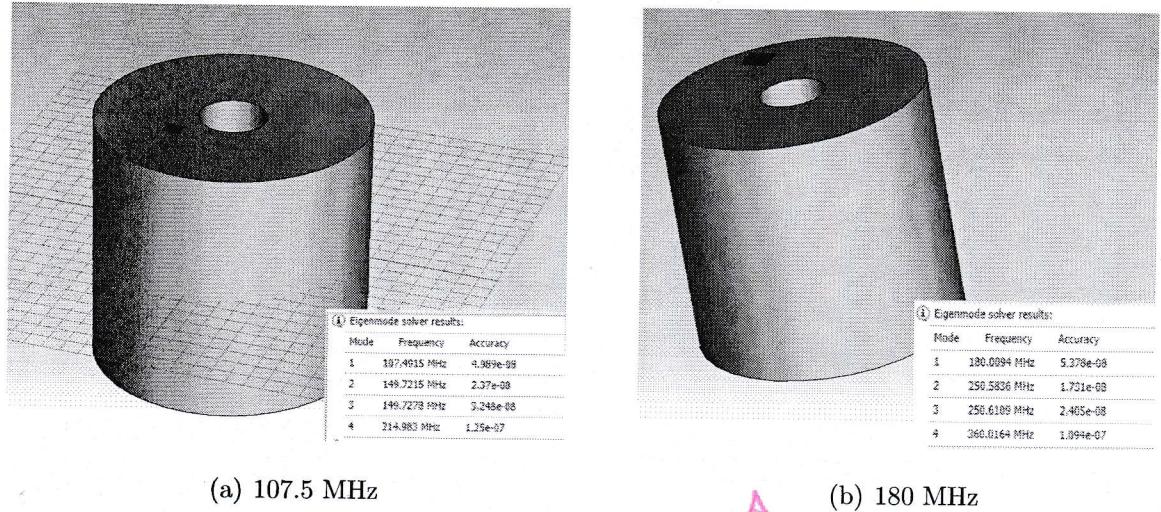


Figure 3.2. CST Eigenmode results for *Equation 3.1*.

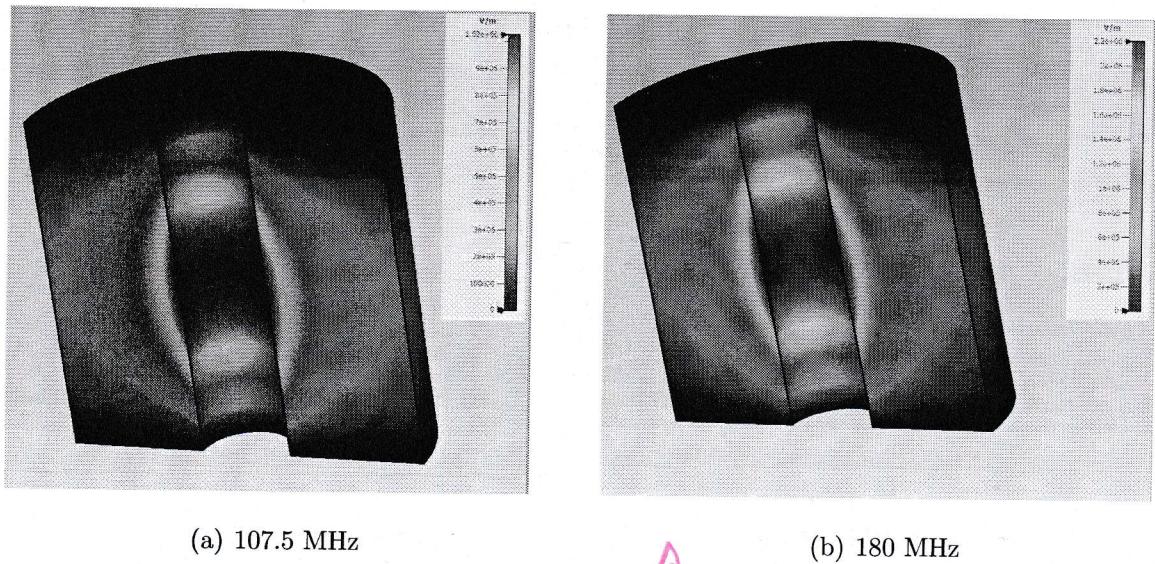


Figure 3.3. CST Electric field solutions for *Equation 3.1*.

As mentioned by *Pottier*, truncated cone terminations in inner cylinder can improve the shunt impedance Z of the cavity [1]. In the *Figure 3.4*, Poission Superfish results of such a modification with matching height increase to keep resonant frequency can be found.

All calculated values below refer to the mesh geometry only.							
Field normalization (NORM = 8):	EZERO =	1.00000	MV/m				
Frequency	=	107.50055	MHz				
Normalization factor for EB =	1.000	MV/m		107.50055	MHz		
Stokes parameter	=	0.00027432	Joules/cm				
Using standard room-temperature copper:							
Surface resistance	=	2.70499	milliohms				
Normal-conductor resistivity	=	1.77410	microohm-cm				
Operating temperature	=	20.0000	C				
Power dissipation	=	117.3612	MW				
Q = 3045.6	Shunt impedance	=	4810.014	ohms			
r/Q = 50212.880	Ohm	Wake loss parameter	=	9.12258	V/mC		
Average magnetic field on the outer wall	=	2655.04	A/m, 0.75530	Wb/cm ²			
Maximum H (at X,Y = 73.55,71)	=	2797.08	A/m, 0.81587	Wb/cm ²			
Maximum E (at X,Y = 75.3,8.8)	=	1.06482	MV/m, 6.485182	Kilop			
Ratio of peak fields: Emax/Eax	=	3.3890	MV/m/(MV/m)				
Peak-to-average ratio Emax/EB	=	1.0602					
Wall segments:							
Segment	Xend (cm)	Yend (cm)	Emax (MV/m)	Pmax (A)	P/A (W/cm ²)	dE/dX (MHz/mm)	dE/dY (MHz/mm)
1	18.399	0.0000					
1	18.399	51.496	0.9956	16.24	317.9	-4.2238E-02	0.4906
2	26.399	71.000	0.3137	18.47	372.4	-4.3908E-02	-1.4744E-02
3	75.399	71.000	4.82766E-02	47.31	975.5	0.0666	0.1700
4	75.399	0.9988	1.894	35.25	496.4	-3.5050E-03	0.1700
Total				117.4			

(a) 107.5 MHz truncated

All calculated values below refer to the mesh geometry only.							
Field normalization (NORM = 8):	EZERO =	1.00000	MV/m				
Frequency	=	180.09245	MHz				
Normalization factor for EB =	1.000	MV/m		10888.517	MHz		
Stokes parameter	=	0.0001145	Joules/cm				
Using standard room-temperature copper:							
Surface resistance	=	3.50020	milliohms				
Normal-conductor resistivity	=	1.77410	microohm-cm				
Operating temperature	=	20.0000	C				
Power dissipation	=	128.3570	MW				
Q = 38905.8	Shunt impedance	=	3270.000	ohms			
r/Q = 32377.188	Ohm	Wake loss parameter	=	9.14325	V/mC		
Average magnetic field on the outer wall	=	2665.56	A/m, 1.24531	Wb/cm ²			
Maximum H (at X,Y = 43.5,42)	=	2678.17	A/m, 1.2553	Wb/cm ²			
Maximum E (at X,Y = 45.0,0)	=	1.00154	MV/m, 6.020642	Kilop			
Ratio of peak fields: Emax/Eax	=	1.3603	MV/m/(MV/m)				
Peak-to-average ratio Emax/EB	=	1.0015					
Wall segments:							
Segment	Xend (cm)	Yend (cm)	Emax (MV/m)	Pmax (A)	P/A (W/cm ²)	dE/dX (MHz/mm)	dE/dY (MHz/mm)
1	11.250	0.0000					
1	11.250	25.250	0.9981	17.74	0.583	7.6459E-02	0.000
2	15.000	43.000	0.1700	9.122	1.181	-8.1176E-02	-4.5898E-02
3	45.000	42.000	1.1924E-02	37.29	1.242	0.000	-0.3870
4	45.000	0.9988	1.982	26.38	0.6282	-4.6971E-03	0.000
Total				90.54			

(b) 180 MHz truncated

Figure 3.8. Poisson Superfish calculations with *truncated* cavities.

One can observe the shunt impedance gain between truncated and unmodified cavities in *Figures 3.7 and 3.8* as

$$\Delta Z_{107.5} = 2.5\%, \quad \Delta Z_{180} = 2.7\%. \quad (3.2)$$

To start, an optimization criteria for the following pass such as

- Ensure the phase stability of the synchronous electron
- Maximize the energy gain of the synchronous electron
- Minimize the energy spread of the beam
- Minimize the phase lag spread of the beam
- All of the above with decided weights

needs be selected. Using a simulation software such as *CST Studio Particle Module*, the optimum value of K for the criteria can be found by simulating for different values of K (*sweeping*) and analyzing the results. This process can be repeated for each magnet until the desired beam characteristics in the end of the accelerator are achieved.

However, one caviat of this technique is that, available simulation softwares are not well suited for this kind of process. As mentioned above, *CST Studio Particle Module* has a *parameter sweep* functionality. But the calculation times are too slow to be useful in this particular problem. A *custom built software offering magnet optimizing sweep functionality will be discussed in the following chapter.*

3.3. Initial Design at KAHVELab

A rhodotron with operation frequency of 107.5 MHz was decided to be built at KAHVELab. Frequency was selected to benefit from the RF power supply units at hand. After considering the earlier simulations mentioned in *Sections 3.1 and 3.2*, an initial design was created.

Figure 3.14 shows that this design provides relatively uniform magnetic field inside, although magnetic field gradient in the openings can be improved. After adding two more magnets and using $\gamma = 15^\circ$, the second prototype design was created (*Figure 3.15*). 

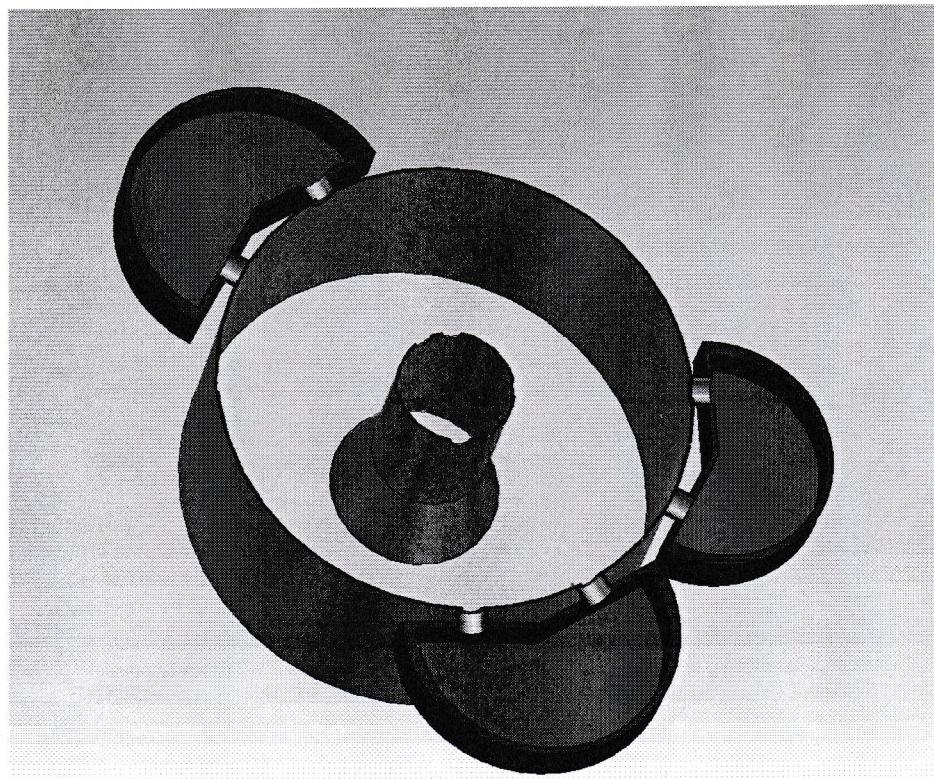


Figure 3.15. Initial cavity with three magnets as shown in *Figure 3.13*, $\gamma = 15^\circ$. 

These designs, together with 40 keV e^- gun injecting with phase lag of 15° for 1 ns, was simulated at RF power of 12 kW using *CST Studio Particle Module PIC Solver*. Results of these simulations can be found in *Figure 3.16*. 

The behavior of the beam in the *Figure 3.16* after the first pass indicates that phase stability is not maintained. When the simulation results are examined further, it was observed that the beam started the second pass at $t \approx 12$ ns after the injection. Considering the starting phase lag $\phi_{lag} = 15^\circ$, this would mean phase lag of the second pass was $\phi_{lag2} = 120^\circ$. This result can be observed in *Figure 3.16*, as the beam decelerates after a short acceleration in the beginning of the second pass.

After various similar observations in *CST* simulations, underlying cause of this problem was thought to be the $L_{pass} = n\lambda$ approach on magnet design failing in low energies, as anticipated and discussed earlier in the *Section 3.2.2*. *The necessity for an alternative tool, capable of aiding researchers in improving beam behavior, was deemed evident.*

Several improvements were made to KAHVELab Rhodotron initial designs [13]. These include curved edges on cavity which reduce the power loss on cavity walls, and redesigned iron casing on magnets for ease of production and maintenance. Improved designs can be seen in the figures below.

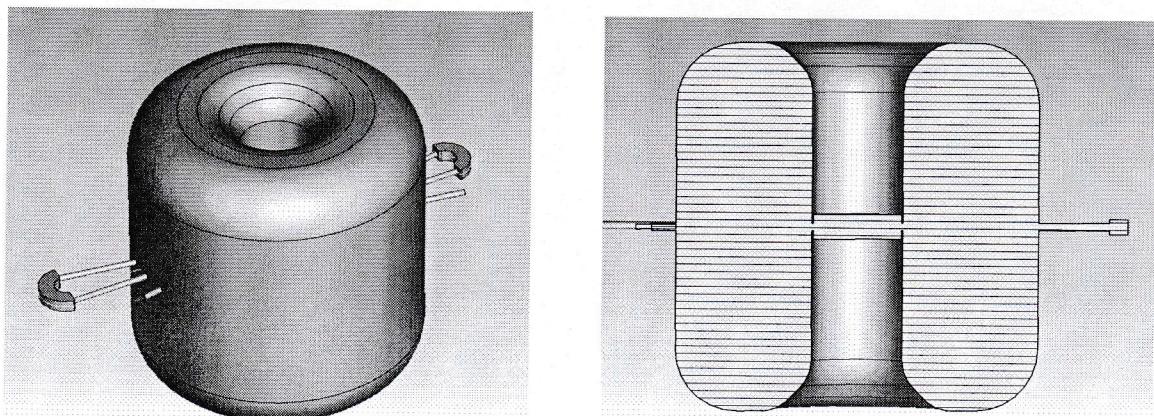


Figure 3.17. Improved cavity design.

```

1  for (double t=0; t<SimuTime; t+=dT){
2      double RelBeta = v/c;
3      double RelGamma = 1.0 / sqrt(1.0-RelBeta*RelBeta);
4
5      double ef=Eradiation(r_pos*1000,t,0); // convert position to mm
6      double acc=ef*1E6*eQMratio/(RelGamma*RelGamma*RelGamma);
7
8      r_pos = r_pos + v * dT*ns + 1/2*acc*(dT*ns)*(dT*ns);
9      v = v + acc*dT*ns;
10
11     RelBeta = v/c;
12     RelGamma = 1.0 / sqrt(1.0-RelBeta*RelBeta);
13     Et=RelGamma*E0;
14 }
```

OK



Figure 4.1. Core logic loop of the POC.

OK

After simulating the *synchronous electron* mentioned in *Section 1.2.4.2* with $\phi_{lag} = 15^\circ$, $P_{diss} = 40\text{kW}$, $R_1 = 0.188\text{m}$, $R_2 = 0.753\text{m}$, $f = 107.5\text{MHz}$ for one pass, the results from *POC*, *CST* and *Equation 1.36* were found to be

OK $E_{Pottier} = 0.593\text{MeV}, \quad (4.1)$

$E_{CST} = 0.521\text{MeV}, \quad (4.2)$

OK $E_{POC} = 0.532\text{MeV}. \quad (4.3)$

As can be observed in *Equation 4.2*, *POC software* and *CST* produced similar results; $\Delta E_{(POC-CST)}/E_{CST} \approx 2\%$. while *Equation 1.36* was noticeably different; $\Delta E_{(Pottier-CST)}/E_{CST} \approx 14\%$. This inconsistency between *Equation 1.36* and *CST* simulation was noticed in the earlier simulations as well, which supported the idea of using another simulation software to reduce reliance on *CST*. Since using a single software would have made the process error prone.

Addition of bunches would immediately proven useful when finding optimal phase lag for bunch entry. ϕ_{lag} for a bunch was defined as the RF phase when the first e^- of the bunch has entered the cavity and it defines the starting time of the current pass. To use the *parameter sweep* method, as used in L_{out} optimization, relevant bunch characteristics are defined as

- μE : Average energy,
- E_{rms} : Root mean square of energy,
- R_{rms} : Root mean square of e^- positions.

Optimal ϕ_{lag} would produce maximum μE while minimizing E_{rms} & R_{rms} . For the first pass, E_{rms} and R_{rms} would be vaguely dependent of each other; therefore, initial implementation of ϕ_{lag} sweep was based on minimizing E_{rms} during simulation (see Figure 4.2). For μE considerations, data from the software would be analyzed either manually or by using external tools such as *ROOT framework*.

```

1 int phase_opt(int phase_sweep_range){
2     double minrms = 1;
3     int opt_phase;
4     for(int RFphase = -phase_sweep_range; RFphase <= phase_sweep_range; RFphase++){
5         Bunch bunch1(RFphase);
6         double t1 = 0;
7         bunch1.bunch_gecis_t(t1);
8         bunch1.reset_pos();
9
10        if( bunch1.E_rms() < minrms ){
11            minrms = bunch1.E_rms();
12            opt_phase = RFphase;
13        }
14    }
15    return opt_phase;
16 }
```

Figure 4.2. ϕ_{lag} Optimization For Initial Bunch Design.

```

1 vector3d CoaxialRFField::actOn(Electron& e){
2     vector3d Efield = getField(e.pos);           // Calculate E vector
3     vector3d F_m = Efield*1E6*e.QMratio;        // Calculate F/m vector
4     vector3d acc = (F_m - e.vel*(e.vel*F_m)/(c*c))/e.gamma(); // Calculate a vector
5     return acc;
6 }
```

(Handwritten note: A circled '4' is written above the caption.)

Figure 4.3. e^- - \vec{E} interaction logic from *Equation 1.18.*

```

1 vector3d MagneticField::actOn(Electron& e){
2     if (isInside(e.pos) == -1)
3         return vector3d(0,0,0);
4     vector3d Bfield = getField(e.pos);           // Calculate B vector
5     vector3d F_m = (e.vel % Bfield)*e.QMratio; // Calculate F/m vector
6     vector3d acc = (F_m)/e.gamma();             // Calculate a vector
7     return acc;
8 }
```

(Handwritten note: A circled '4' is written above the caption.)

Figure 4.4. e^- - \vec{B} interaction logic from *Equation 1.18.*

Simulating in 3D had one other benefit; it was now possible to visualize the results by rendering the interaction data. For this purpose, *gnuplot* was integrated into *Rhodotron Simulation* to produce 2D visualization of acceleration plane. Rendered results could be stored as *gif* animations. Two of such renders can be seen in *Figure 4.5.*

single column

- Electric and magnetic field import
- Field generator module to directly produce field files from specified cavity
- Synchrotron radiation calculations to determine radiation and energy loss
- $e^- - e^-$ interactions
- Redesign of the GUI
- 3D render in GUI
- Refactoring of the $e^- - \mathbf{EM}$ interaction and logging to improve performance
- Refactoring of the GUI Render Frame to improve render speed
- Refactoring the magnet class to introduce field leaks
- Extension of analysis tools in Analyze Frame
- L_{out} sweep in Sweep Frame

After the cavity manufacturing is concluded, energy gain predictions of this software will be tested on this cavity. Complementary bending magnet design will be the first challenge of *Rhodotron Simulation*, which will provide tremendous testing and improvement opportunities.

6.2. Discussion

In conclusion, a new computational $e^- - \mathbf{EM}$ interaction simulation and analysis tool that focuses on designing and improving *Rhodotron-type* accelerators has been successfully developed; a *Rhodotron-type* accelerator with an operating frequency of 107.5 MHz and a target energy of 1 – 5 MeV has been designed and is currently being manufactured.

The software features a robust GUI that is capable of rendering and analyzing simulation results. Leap-frog and Runge-Kutta numerical methods have been implemented for initial speed and accuracy tests. The real-world performance and accuracy of this new tool, on the other hand, cannot be tested before the cavity production is completed and the performance is compared with the predictions of the tool; However, the capabilities it presents, coupled with the analogous results obtained from other extensively utilized simulation tools, lead to the inference that it holds promise.

Should be after references

APPENDIX A: Intermediate Codes

```

1  for(double i = 2; i < 9; i += dT_out){
2      t_dum += i;
3      double enow = gecis(r_pos, vel, Et, t_dum);
4      if( enow > maxE ){
5          maxE = enow;
6          t_opt = i;
7      }
8      t_dum = t;
9  }

1  double gecis(double r_pos, double vel, double Et, double &t){
2      for(; r_pos >= -R2 && r_pos <= R2 ; t+=dT){
3          vel = c*sqrt(Et*Et-E0*E0)/Et;
4          double RelBeta = vel/c;
5          double RelGamma = 1.0 / sqrt(1.0-RelBeta*RelBeta);
6
7          double ef=Eradiat(r_pos*1000,t,RFphase*deg_to_rad);
8
9          double acc=ef*1E6*eQMratio/(RelGamma*RelGamma*RelGamma);
10
11         r_pos = r_pos + vel * dT*ns + 1/2*acc*(dT*ns)*(dT*ns);
12         vel=vel+acc*dT*ns;
13         RelBeta = vel/c;
14         RelGamma = 1.0 / sqrt(1.0-RelBeta*RelBeta);
15         Et=RelGamma*E0;
16     }
17     return Et;
18 }
```

Figure A.1. L_{out} Optimization For Single e^- .

REFERENCES

- Pottier J.W.)
"new"*
- wrong
formal!
Correct all*
1. Jacques Pottier, *A new type of rf electron accelerator: The rhodotron*, Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms, Vol. 40–41, No 2, pp 943-945, 1989.
 2. Adığüzel, A., S. Açıksöz, A. Çağlar, H. Çetinkaya, Ş. Esen, D. Halis, A. Hamparsunoglu, T.B. İlhan, A. Kılıçgedik, O. Koçer, S. Oğur, S. Öz, A. Özbey, V.E. Özcan and N.G. Ünel *Ion source and LEBT of KAHVELab proton beamline*, Journal of Instrumentation, Vol. 18, 2023
 3. Bassaler, J.M., J.M. Capdevila, O. Gal, F. Lainé, A. Nguyen, J.P. Nicolaï, K. Umiastowski, *Rhodotron: an accelerator for industrial irradiation*, Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms, Volume 68, No. 1–4, Pages 92-95, 1992.
 4. Jongen, Y., *Manufacturing of Electron Accelerators*, Ion Beam Applications (IBA) Chemin du Cyclotron 3, B-1348 Louvain-la-Neuve, Belgium. 2001.
 5. Jensen E., in Proceedings of the CAS-CERN Accelerator School: Advanced Accelerator Physics CERN-2014-009, edited by W. Herr, Trondheim, Norway, pp. 405–430, 2013.
 6. Fukushima, T., *Reduction of Round-off Errors in Symplectic Integrators*, The Astronomical Journal, Vol. 121, pp. 1768-1775, 2001.
 7. User's Guide for the POISSON / SUPERFISH group of codes, Los Alamos National Laboratory, New Mexico, USA, 1987. Accessed on Aug 10, 2023.
 8. The European Organization for Nuclear Research, "ROOT - An Object Oriented Data Analysis Framework Release v6.20/04, 01/04/2020", 2020, <https://zenodo.org/record/3895855>, accessed on Aug 10, 2023.