



Department of Physics, IIT Delhi

**Course Code : PYL655
Semester - IV, 2022-23**

Laser Physics Assignment

Harikesh Kushwaha (2021PHS7181)

Course Coordinator: Prof. M. R. Shenoy

Assignment Topic

Ray Transfer Matrix Approach: Write a suitable algorithm and program for ray tracing in a spherical mirror resonator. See that for large number of rays, the ray diagram has the same appearance as that of a Gaussian beam.

Introduction

Ray tracing with the ray transfer matrix approach is a simple yet powerful tool to study properties of an optical system. In the first section, we'll start by giving a brief introduction on the ray transfer matrix approach, the ray transfer matrix (RTM) itself, the assumptions underlying them and how they are so powerful. In the second section, we'll calculate the RTM of some simple system and then use it to create the RTM of the spherical mirror resonator (SMR). The third section will give some details about the algorithm which has been developed to trace the rays in an SMR using [matplotlib](#) and [numpy](#). The fourth section has the results obtained by the algorithm and we will see that for large number of rays, the ray diagram has the same appearance as that of a Gaussian beam.

1 Ray Transfer Matrix Analysis

Ray transfer matrix analysis is a way to perform ray tracing. In this method, each optical element is described as a 2×2 matrix. The incoming light ray is described by a column vector. The product of the matrix and the vector gives the outgoing ray. The product of the matrices of all the optical elements gives the overall transformation of the incoming ray. This method is used to analyze the propagation of light through optical systems, such as lenses and mirrors. It is also used to analyze the propagation of light through optical fibers.[1]

This technique is derived using the paraxial approximation, which assumes that all the ray directions are at small angles θ such that the following approximation holds:

$$\begin{aligned}\sin \theta &\approx \theta \\ \tan \theta &\approx \theta\end{aligned}\tag{1}$$

1.1 Ray Transfer Matrix

The 2×2 matrix used to trace the rays is called the ray transfer matrix. It is a matrix that describes how an incident light ray is transformed as it passes through an optical system, such as a lens or a mirror. The matrix is usually denoted as M and is written as:

$$M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}\tag{2}$$

As the elements of the matrix are A , B , C and D , the matrix is also called the ABCD matrix. The elements of the matrix are determined by the optical properties of the system.

1.2 The Incoming Ray

The incoming ray is described by a column vector r :

$$r = \begin{bmatrix} y \\ \theta \end{bmatrix}\tag{3}$$

where y is the distance from the optical axis and θ is the angle of the ray with respect to the optical axis.

The RTM approach then states that for any optical system with RTM M , if the incident ray is r_i then the output ray r_f will be:

$$\begin{aligned} r_f &= Mr_i \\ \begin{bmatrix} y_f \\ \theta_f \end{bmatrix} &= \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} y_i \\ \theta_i \end{bmatrix} \end{aligned} \quad (4)$$

The elements of the matrix A , B , C and D are determined by the properties of the optical system.

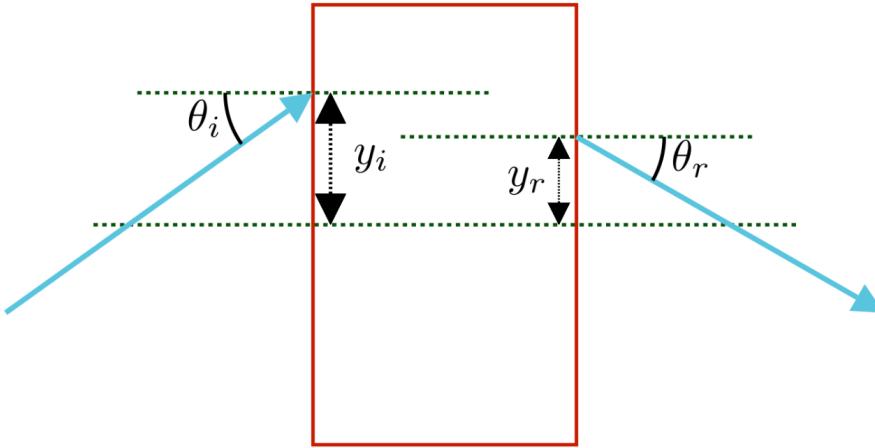


Figure 1: A picture representing the incoming and outgoing rays (In blue arrows) through an optical system (depicted as a red rectangle). The input and output rays can be related by the equation 4.
All the pictures in the paper are created by the author using [manim](#)[2].

1.3 Advantages of the RTM Approach

The beauty of the RTM approach is that it does not matter what the optical system is, as long as we know that RTM for that system, we can do the ray tracing. Furthermore, if the system consists of a number of different optical elements, with different RTM's, the RTM of the complete system is just the multiplication of the whole system. For example, if there are four optical elements (as we will see in the case of optical resonator) in the whole system with RTM's given by M_1 , M_2 , M_3 and M_4 , the RTM of the system is:

$$M = M_4 M_3 M_2 M_1$$

This makes the ray tracing very straightforward. Next, we'll consider the sign conventions used while calculating the RTM and tracing the rays.

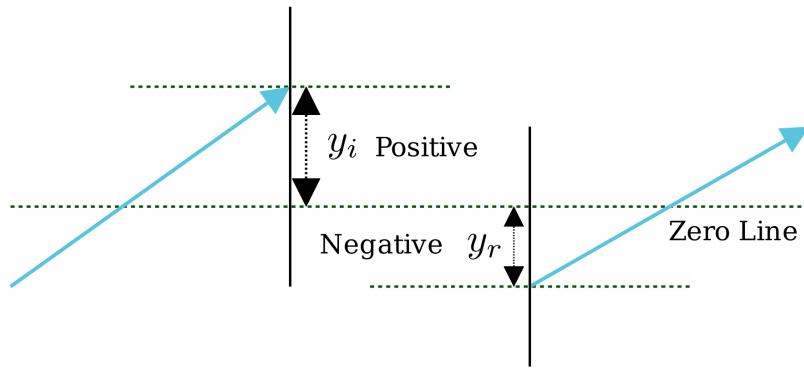


Figure 2: If the height is above the zero line (the optical axis), the sign is positive else is negative.

1.4 Sign Conventions

The ray tracing technique is based on two reference planes, called the input and output planes, each perpendicular to the optical axis of the system. The first coordinate defining a ray, the height, is measured on this plane with reference to the optical axis. The figure above shows the optical axis as the *zero line*. Using this, the sign convention is:

1. The height y is positive if the ray is above the optical axis and negative if the ray is below the optical axis.
2. The angle θ is positive if the ray is moving above from the optical axis and negative if the ray is moving below the optical axis.
3. The radius of curvature of a concave mirror is negative and that of a convex mirror is positive.

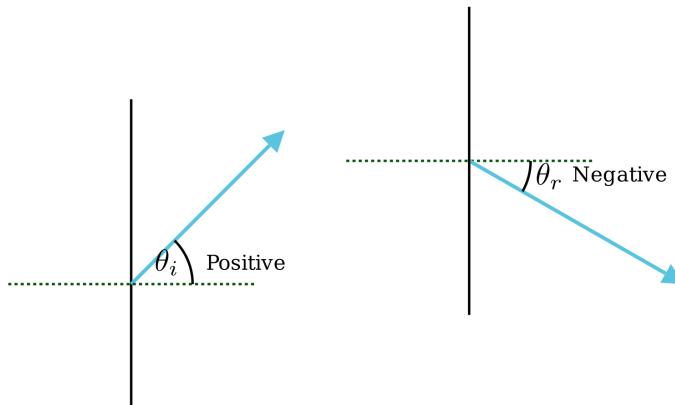


Figure 3: If the height is above the zero line (the optical axis), the sign is positive else is negative.

2 RTM for Spherical Mirror Resonator

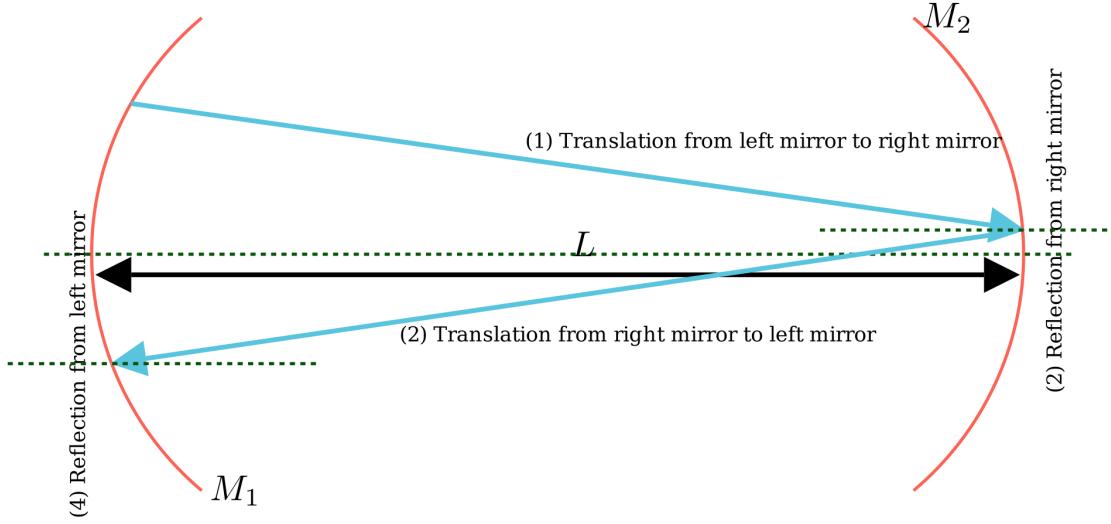


Figure 4: If the height is above the zero line (the optical axis), the sign is positive else is negative.

For ray tracing, we need to know the RTM of the SMR. In an SMR, the rays undergoes four different transformations while going from the left mirror to the right mirror and back. These transformations are:

1. Propagation through the resonator (from left to right), distance L
2. Reflection from the left mirror \$M_2\$
3. Propagation through the resonator (from right to left), distance L
4. Reflection from the right mirror \$M_1\$

So, we need calculate the RTM for each of these transformations and then multiplying them to get the RTM of the whole system.

2.1 Propagation Through a Distance of L

We can see from the figure 5 that during the propagation through a distance \$L\$, the angles of the rays remain the same but the height changes. That is:

$$\begin{aligned}\theta_r &= \theta_f \\ y_f &= L \tan \theta_r + y_i \approx L\theta_r + y_i\end{aligned}\tag{5}$$

This can be rewritten as:

$$\begin{bmatrix} y_f \\ \theta_f \end{bmatrix} = \begin{bmatrix} 1 & L \\ 0 & 1 \end{bmatrix} \begin{bmatrix} y_i \\ \theta_i \end{bmatrix}\tag{6}$$

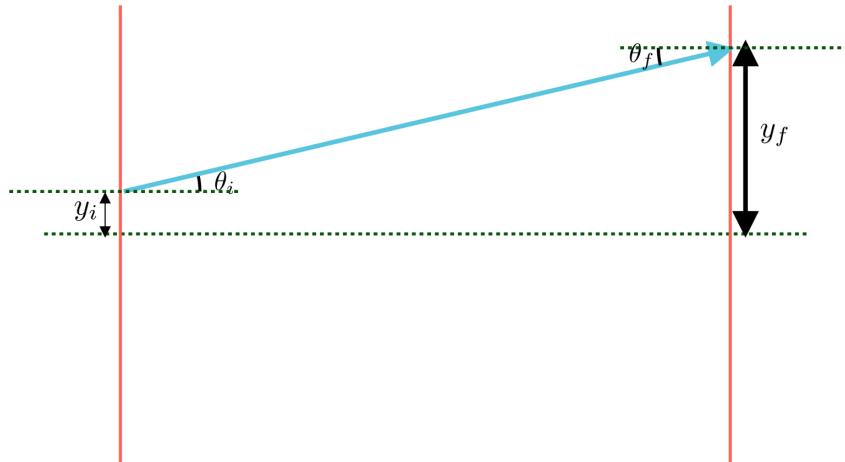


Figure 5: Propagation through a distance of L .

So, the RTM for this transformation is:

$$\mathbf{M} = \begin{bmatrix} 1 & L \\ 0 & 1 \end{bmatrix} \quad (7)$$

2.2 Reflection from a Spherical Mirror

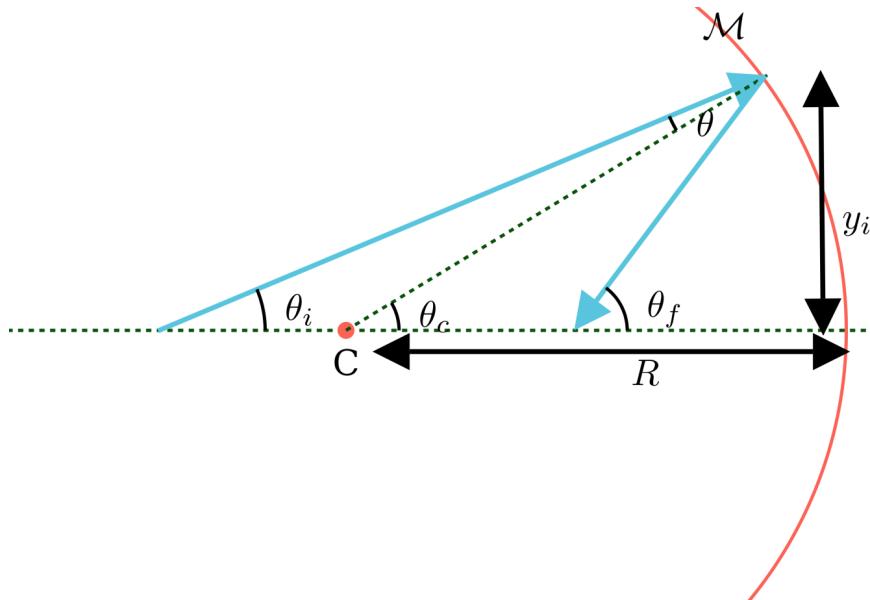


Figure 6: Reflection from a spherical mirror.

For reflection from a spherical mirror, the height remains the same while the angle changes.
So:

$$y_f = y_i$$

To determine the relation between the incoming and outgoing angles, we use the figure 6 above to

write:

$$\theta_c = \theta_i + \theta$$

and

$$\theta_f = \theta_c + \theta$$

Subtracting these two gives:

$$\theta_f + \theta_i = 2\theta_c \quad (8)$$

Using the paraxial approximation, we can write:

$$\begin{aligned} 2\theta_c &\approx 2 \tan \theta_c \\ &= -2 \frac{y_i}{R} \quad (\text{Using the sign convention}) \\ \Rightarrow -\theta_f + \theta_i &= -2 \frac{y_i}{R} \quad (\text{Again, using the sign convention}) \\ \theta_f &= \theta_i + \frac{2y_i}{R} \end{aligned} \quad (9)$$

This means that we can write:

$$\begin{bmatrix} y_f \\ \theta_f \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \frac{2}{R} & 1 \end{bmatrix} \begin{bmatrix} y_i \\ \theta_i \end{bmatrix} \quad (10)$$

So, the RTM for this transformation is:

$$\mathbf{M} = \begin{bmatrix} 1 & 0 \\ \frac{2}{R} & 1 \end{bmatrix} \quad (11)$$

2.3 The RTM of the SMR

Using the equations 7, 11 and the fact that the rays undergoes four transformations, we can write the RTM of the SMR by multiplying these matrices. Assuming the radii of curvature of the left and right mirrors are R_1 and R_2 respectively, and the distance between the mirrors is L , the RTM of the SMR is:

$$\mathbf{M} = \begin{bmatrix} 1 & 0 \\ \frac{2}{R_1} & 1 \end{bmatrix} \begin{bmatrix} 1 & L \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \frac{2}{R_2} & 1 \end{bmatrix} \begin{bmatrix} 1 & L \\ 0 & 1 \end{bmatrix} \quad (12)$$

These are all the information needed to trace rays in the SMR.

3 Ray Tracing

The program to trace the rays in the SMR is written in Python. I've used [matplotlib](#)[3] for the visualization and [numpy](#)[4] for the matrix operations. The code can be found in the [Github Repository](#)[5].

3.1 The Algorithm for Ray Tracing

The algorithm for ray tracing is very straightforward and is as follows:

1. Define the radius of curvature (ROC) of both the mirrors $R_1 \& R_2$ and the distance between the mirrors L .
2. Use the equations 7 and 11 to find the RTM for propagation M_P , for the reflection from the left mirror M_{R_L} and for the reflection from the right mirror M_{R_R} .
3. Choose a starting point for the ray and the angle of incidence, say $\mathbf{r}_0 = [y_0, \theta_0]^T$. The ray will start from the left mirror.
4. To determine the y-position of the ray when it reaches the right mirror, multiply \mathbf{r}_0 with M_P , call it \mathbf{r}_1 . The y-coordinate of \mathbf{r}_0 and the y-coordinate of \mathbf{r}_1 can be used to draw a line which will give the path at which the ray travels from left to right mirror. (Assuming that the x-coordinate of the mirror remains the same, which is indeed the case.)
5. Now, multiply \mathbf{r}_1 with M_{R_R} (this will result in change in the angle) and then again with M_P . Call it \mathbf{r}_2 . This will give the y-position of the rays when it reaches the left mirror.
6. Again, a line can be drawn using the y-positions of \mathbf{r}_1 and \mathbf{r}_2 . This will depict the ray coming to the left mirror from the right mirror.
7. Continue steps 4-6 by using the last position vector as starting point. This traces the rays.

3.2 Some Detail About the Program

This section gives some more detail about the program. The program is written using [object oriented programming](#) where we start by modeling the real world objects in code. This makes the code very intuitive and easy to implement. The program uses two objects: [Mirror](#) and [Resonator](#). Physically, a resonator is nothing but two mirrors separated by some distance. The [Resonator](#) object is implemented likewise. A [Resonator](#) can be instantiated by giving the radii of the two mirrors and the separation between them. Then the [propogate\(\)](#) method can be called to trace the rays passing it the starting position and the number of rays to plot (This method uses the same algorithm outlined above to trace the rays.). Please refer to the Github Repository for the source code and for more detail on how the algorithm works.

4 Results

This section shows some ray diagrams of both stable and unstable resonators with ray tracing done by the program described above. We will see that for large number of rays, the ray diagram has the same appearance as that of a Gaussian beam.

Here are ray diagrams for some resonators. The first few diagrams are for stable resonators. Then we give one ray diagram for the unstable resonator. The resonator details and ray tracing details are given in the caption.

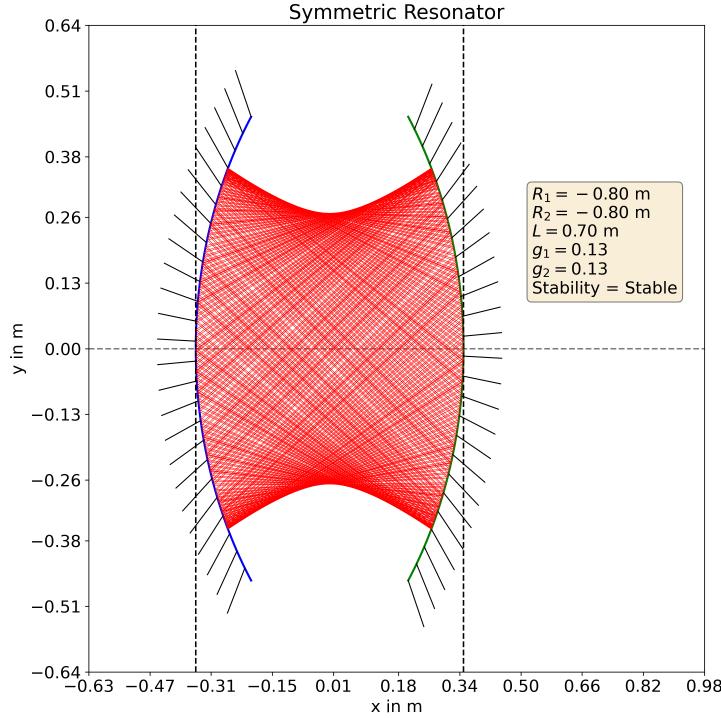


Figure 7: Ray diagram of a symmetric resonator. The metadata for the resonator is shown in the figure. The initial position is $[0.3, -0.1]^T$. 200 rays are traced. The Gaussian shape is readily observed.

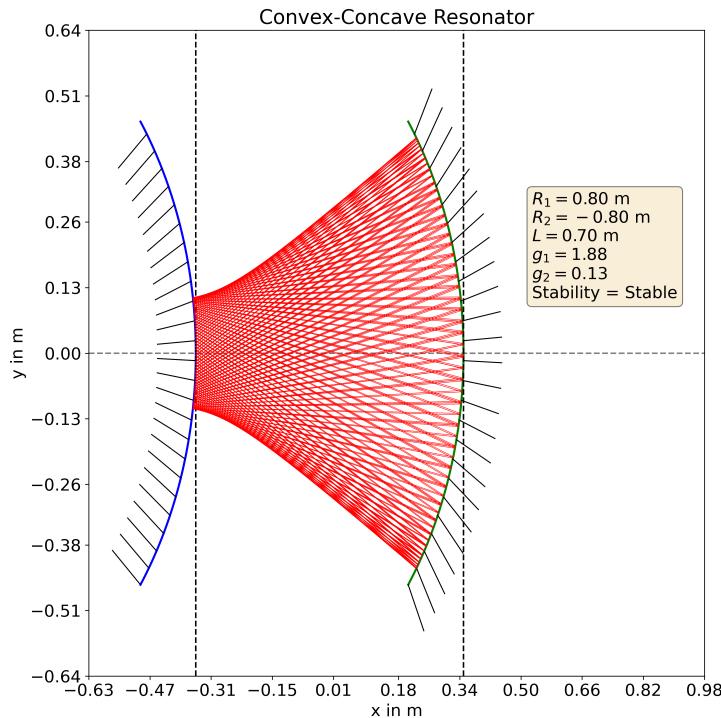


Figure 8: Ray diagram of a resonator with one convex and once concave mirror. The metadata for the resonator is shown in the figure. The initial position is $[-0.1, 0.1]^T$. 200 rays are traced. Once again, the Gaussian shape can be seen.

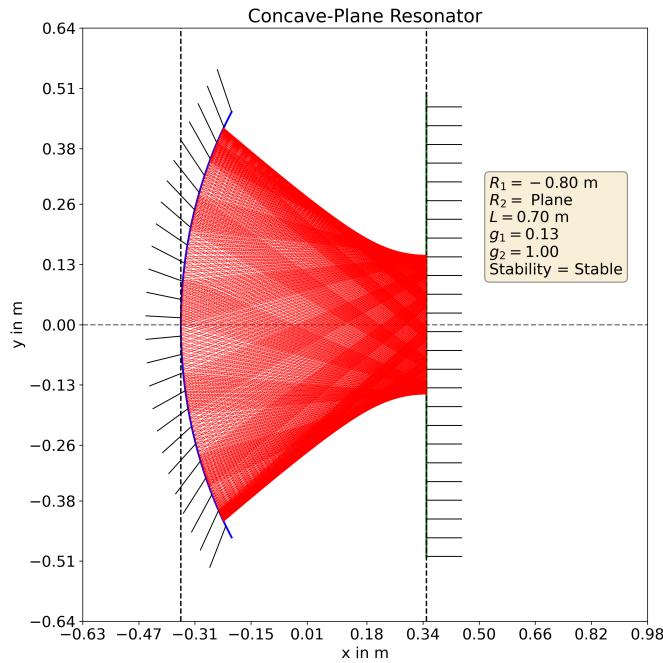


Figure 9: A stable resonator can be created with plane mirror too, given that the second mirror is a concave one. The diagram shows such a resonator. The intial position is $[-0.15, 0]^T$. 300 rays are traced. The Gaussian shape can be see here too.

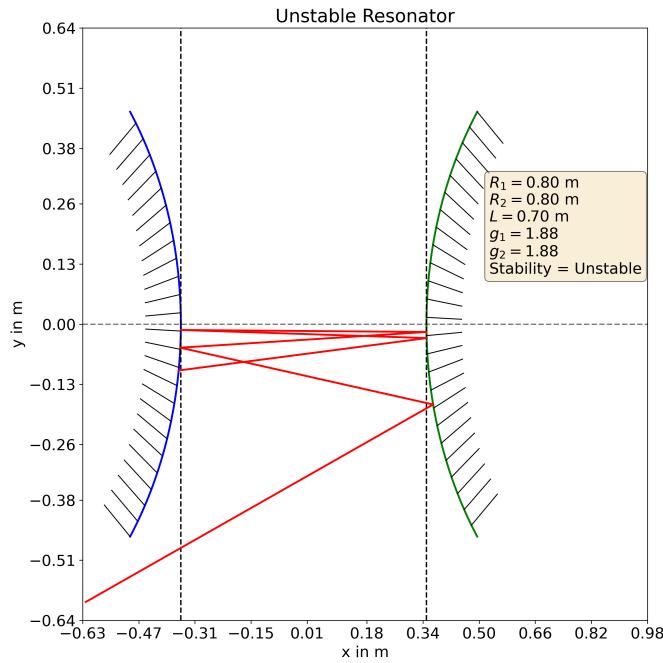


Figure 10: Two convex mirrors can never form a stable resonator. The diagram shows such an unstable. The intial position is $[-0.1, 0.1]^T$. Less than 10 rays are inside the mirrors given in the figure.

5 Conclusion

In this paper, we discussed how RTM can be used to trace rays in a SMR. We derived the RTM for the SMR and used it with `matplotlib` and `numpy` to write a program to trace the rays inside the given resonator. We plotted such a ray diagram for a number of stable resonators and found out that their shape resembles that of a Gaussian beam. We also saw ray diagram of an unstable resonator.

References

- [1] *Ray Transfer Matrix Analysis*. Apr. 8, 2023. URL: https://en.wikipedia.org/wiki/Ray_transfer_matrix_analysis.
- [2] The Manim Community Developers. *Manim - Mathematical Animation Framework*. Version v0.17.3. Apr. 2023. URL: <https://www.manim.community/>.
- [3] J. D. Hunter. “Matplotlib: A 2D graphics environment”. In: *Computing in Science & Engineering* 9.3 (2007), pp. 90–95. DOI: 10.1109/MCSE.2007.55.
- [4] Charles R. Harris et al. “Array programming with NumPy”. In: *Nature* 585.7825 (Sept. 2020), pp. 357–362. DOI: 10.1038/s41586-020-2649-2. URL: <https://doi.org/10.1038/s41586-020-2649-2>.
- [5] Harikesh Kushwaha. *Github Repository for the Term Paper*. Apr. 2023. URL: <https://github.com/Hari31416/Laser-Term-Paper>.