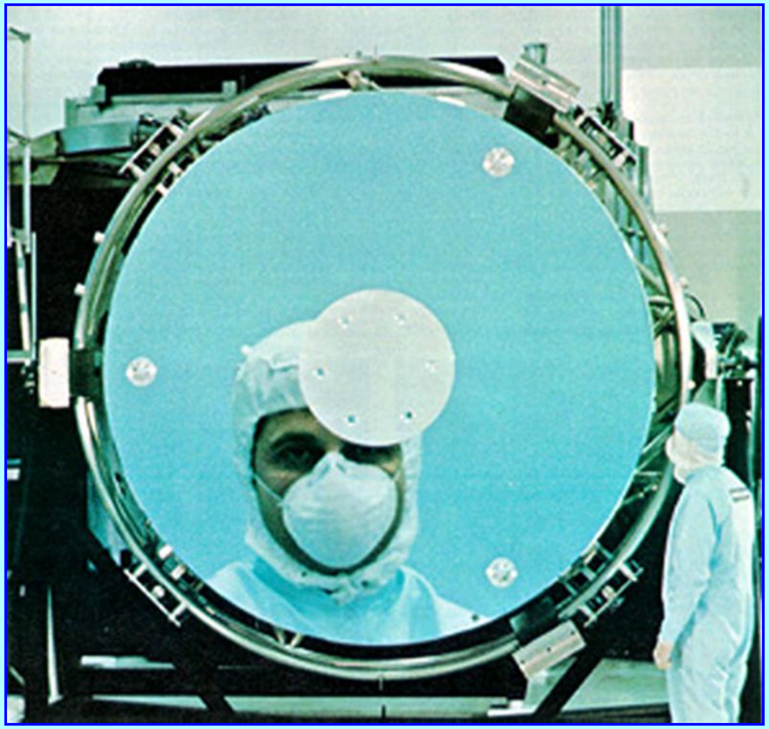
Introduction to Geometrical Optics - a 2D ray tracing Excel model for spherical mirrors - Part 4

by George Lungu

- The previous section dealt with creating a VBA custom function to calculate the coordinates where the incident rays hit the mirror and the angle of the emergent rays.
- This section charts the incident rays and derives the path of the reflected rays creating a custom VBA function for calculating an end point for each reflected rays needed to chart them.
- This is an exact model in the sense that no geometrical approximations are used, however the model does not take into consideration diffraction effects.

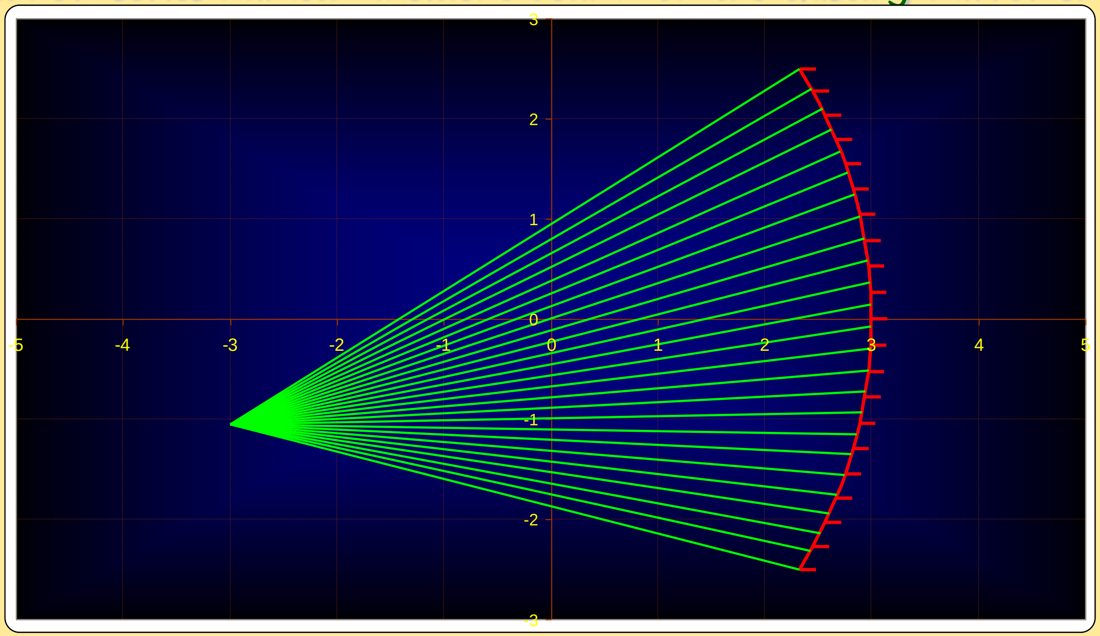
Verifying the "Reflect()" custom function:

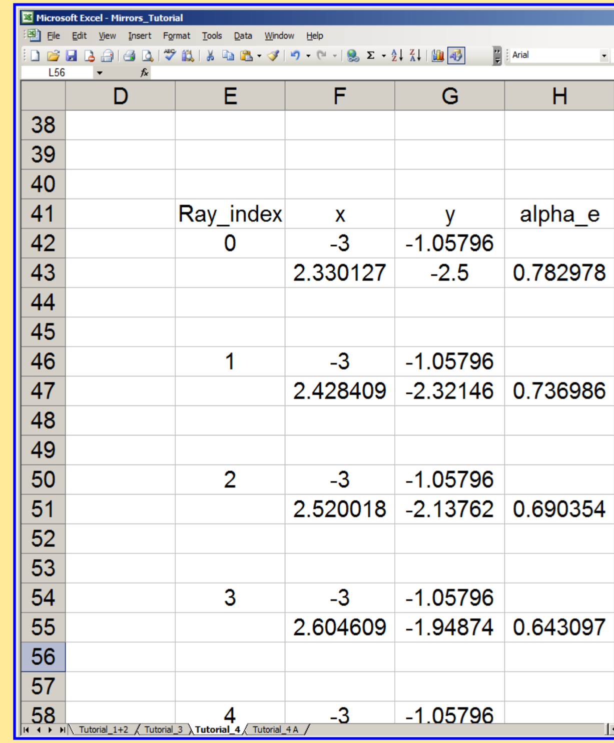
- We had the coordinates of the light source available from the start , and in the previous section we also created a function which returns the Cartesian coordinates of the points of incidence on the mirror and the slope of the reflected rays.
- Let's verify the functionality of the new custom function and for this we need to copy the last worksheet and rename the copy "Tutorial_4"
- Let's create a ray index column in range E42:E138
- E42: "=0", E45: "=E42+1" then copy E45:E47 to range E48:E116.
- The input beam will be placed in column F.



The Hubble space telescope mirror

- F42: "=xL". G42: "=yL", F43: "=Reflect(xL,yL,xM,yM,alpha_min+E42*delta_alpha, Radius)
- Select range F43:H43 then holding F2 down hit Ctrl+Shift+Enter at the same time and now we have the custom function in range F43:H43.
- Copy F42:H44 into range F45:G116 then add range F42:G115 as a new series named "Incident beam" on the existing mirror chart.



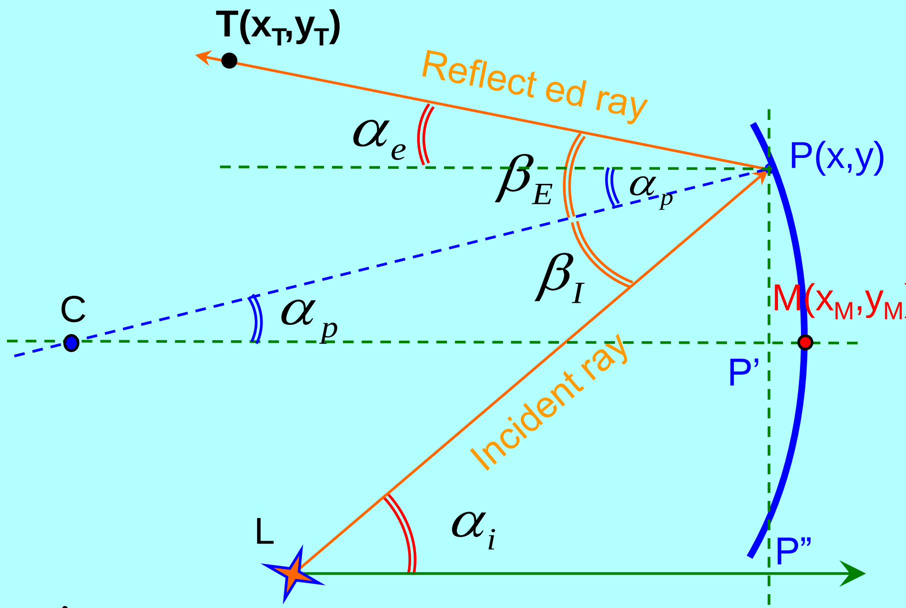


	D	E	F	G	H
38					
39					
40					
41		Ray_index	x	y	alpha_e
42		0	-3	-1.05796	
43			2.330127	-2.5	0.782978
44					
45					
46		1	-3	-1.05796	
47			2.428409	-2.32146	0.736986
48					
49					
50		2	-3	-1.05796	
51			2.520018	-2.13762	0.690354
52					
53					
54		3	-3	-1.05796	
55			2.604609	-1.94874	0.643097
56					
57					
58		4	-3	-1.05796	

Calculating the reflected path:

- We had the coordinates of the light source available from the start , and in the previous section we also created a function which returns the Cartesian coordinates of the points of incidence on the mirror and the slope of the reflected rays.
- From this data we have already charted the incident rays since we have two points for each ray, the source L and the end point since after that, any incident ray turns into a reflected ray.
- For each of the reflected rays we already have the starting point which is the incidence with the mirror point and we need to calculate a second point using the data from this first point and the slope of the ray (returned by the custom function Reflect()).

Finding the end point of the reflected ray:



- For now let's use a the following insight to obtain a second point for the reflected ray:
- Let's assume the charted has the scale for the x axis ranging between x_scale_min and x_scale_max. Also let's assume that the y axis ranges between y_scale_min and y_scale_max.
- If the reflected ray travels left, therefore angle α_E is valued between $-\pi/4$ and $\pi/4$, we choose x_T = x_scale_min and calculate y_T from the reflected ray equation:

$$\begin{cases} x_T = x\_scale\_\min \\ \dfrac{y_T - y}{x_T - x} = -\tan(\alpha_e) \end{cases} \Rightarrow \begin{cases} x_T = x\_scale\_\min \\ y_T = y - (x\_scale\_\min - x)\cdot\tan(\alpha_e) \end{cases}$$

If the reflected ray travels up, therefore angle α_E is valued between $\pi/4$ and $3\pi/4$, we choose y_T = y_scale_max and calculate x_T from the reflected ray equation:

$$\begin{cases} y_T = y\_scale\_\max \\ x_T = x - \dfrac{(y\_scale\_\max - y)}{\tan(\alpha_e)} \end{cases}$$

If the reflected ray travels down, therefore angle α_E is valued between $-3\pi/4$ and $-\pi/4$, we choose y_T = y_scale_min and calculate x_T from the reflected ray equation:

$$\begin{cases} y_T = y\_scale\_\min \\ x_T = x - \dfrac{(y\_scale\_\min - y)}{\tan(\alpha_e)} \end{cases}$$

If the reflected ray travels right, therefore angle α_E is valued between $3\pi/4$ and π or between $-\pi$ and $-3\pi/4$, we choose x_T = x_scale_max and calculate y_T from the reflected ray equation:

$$\begin{cases} x_T = x\_scale\_\max \\ y_T = y - (x\_scale\_\max - x)\cdot\tan(\alpha_e) \end{cases}$$

The "Chart_Reflect()" custom VBA function:

-This function is based on the formulas in the previous page and it will return the coordinate of the end point of a reflected ray, which point will be near the visible range of the chart

- The function will take as arguments: the chart scaling factors, the coordinates of the incident point and the angle of the reflected ray

- Copy the current worksheet and rename the copy "Tutorial_4"

- In the VBA editor, in a module, write the following code:

- These are just the formulas from the previous page set up as a custom VBA function.

-The output of this function is a 1D vector array and we will need to be careful when we type it in (using F2 + Ctrl + Shift + Enter).

- The numbers used in the function are fractions of π.

```
Function Chart_Reflect(x_max, x_min, y_max, y_min, x, y, alpha_reflected)
    Dim xT, yT, pi As Double
    pi = 3.1415
    If alpha_reflected > -pi / 4 And alpha_reflected <= pi / 4 Then
        xT = x_min
        yT = y - (x_min - x) * Tan(alpha_reflected)
    End If
    If alpha_reflected > pi / 4 And alpha_reflected <= 3 * pi / 4 Then
        yT = y_max
        xT = x - (y_max - y) / Tan(alpha_reflected)
    End If
    If alpha_reflected <= -pi / 4 And alpha_reflected > -3 * pi / 4 Then
        yT = y_min
        xT = x - (y_min - y) / Tan(alpha_reflected)
    End If
    If alpha_reflected > 3 * pi / 4 Or alpha_reflected <= -3 * pi / 4 Then
        xT = x_max
        yT = y - (x_max - x) * Tan(alpha_reflected)
    End If
    Chart_Reflect = Array(xT, yT)
End Function
```

to be continued...