Ray Tracing

Northwestern University

Nicolas Guerra
Student of Professor Mel Ulmer
Fall of 2020

Contents

1 Abstract

This report demonstrates how to simulate imperfections in a mirror and how to analyze such imperfections in order to determine if one can build a deployable mirror that works at normal incidence, after coating, in the extreme ultraviolet radiation range. To decide if such a mirror is feasible, the radius of the detector at the focal plane must be comparable to the radius of a microchannel plate. For a mirror with a diameter and focal length of one meter, tiny ripples were placed on the surface after which the radius of a detector needed to capture 90% of the rays was calculated. When the ripples were 1 μm in height from peak to trough and had an angular frequency of 2π, the radius calculated was 6.6 mm. When the ripples were 2 μm in height from peak to trough and had an angular frequency of 10π, the radius calculated was 65.7 mm. When deformations on a larger scale were implemented, the determined radius of the detector was a lot smaller to the scale of just a few microns. Because one can order a custom-made microchannel plate detector about 130 mm in diameter, its seems that manufacturing the hypothesized deployable mirror is attainable.

2 Introduction

The purpose of this report is to document how to simulate ray tracing both mathematically and in Optics Software for Layout and Optimization, or OSLO for short. It is hypothesized that one can build a deployable mirror that works at normal incidence, after coating, in the extreme ultraviolet radiation range. Since imaging is not a concern, the main interest is determining the size of a detector at the focal plane that can capture 90% of the light rays and seeing if the acquired size is similar to that of a microchannel plate detector. Through ray tracing, one can determine the Strehl ratio of the mirror, optical path difference-based root mean square wavefront error, and the corresponding radius. The ray tracing executed in OSLO will be for larger deformations in the mirror whereas the mathematical approach will be for smaller, finer imperfections.

3 Mathematical Approach Versus OSLO Approach

The difference between the mathematical approach that will utilize MATLAB and the OSLO approach is the scale of deformation. The mathematical approach focuses on much smaller, finer imperfections whereas OSLO focuses on larger scale deformations. To illustrate such a difference, the cross-section of a perfect and imperfect mirror will be shown on the same graph using Desmos Graphing Calculator.

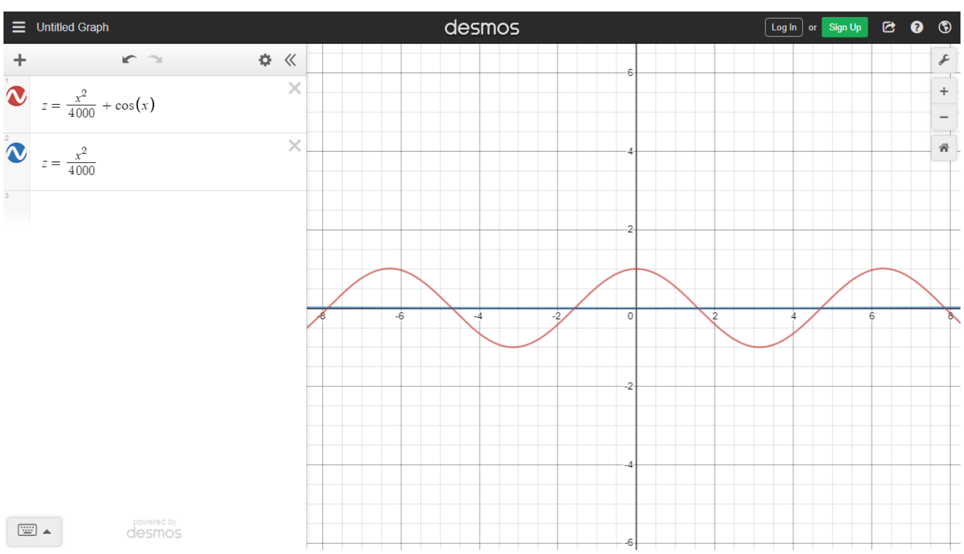

Figure 1: Perfect Mirror (Blue) Plotted with Mirror with Small-Scale Imperfections (Red)

A single unit above is one millimeter. In Figure 1, one ripple is two millimeters tall from peak to trough spaced 2π millimeters apart. The height is a huge exaggeration done for illustration purposes only. The actually ripple height will be 1000 times smaller being only one or two μm tall from peak to trough.

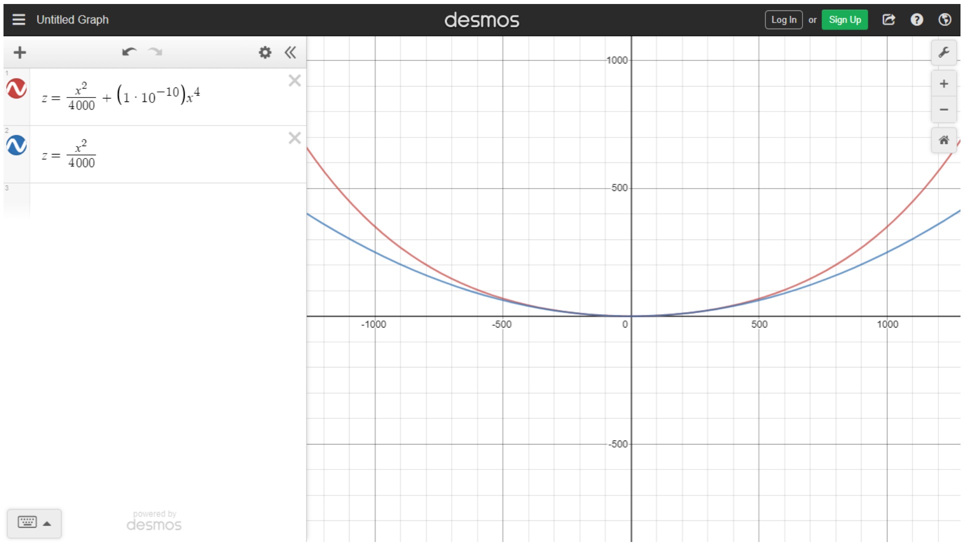

Figure 2: Perfect Mirror (Blue) Plotted with Mirror with Large-Scale Imperfections (Red)

In Figure 2, the mirror would be in full view, and the deformation is easily visible to the naked eye. Of course, this is an exaggeration as well just for illustration purposes, but both figures show the differences between the two approaches. One could technically use the mathematical approach for the bigger deformations, but since OSLO EDU is free and readily available, that is the route chosen for large-scale imperfections.

4 Mathematical Approach

4.1 Singular Ray Trace

The first step is to figure out an equation that models the mirror. This report will simulate ray tracing with a specific parabolic mirror with a certain focal length, but the procedure can be applied to any other mirror. The following is the formula for a parabolic surface.

$$z = \frac{x^2}{a^2} + \frac{y^2}{b^2} \tag{1}$$

The mirror of interest has a focal length of 1000 millimeters and a diameter of 1000 millimeters. Because the mirror will be rotationally symmetric, one can focus on a slice of the mirror at $y = 0$. The following work is to account for the focal length.

$$vertex = (h, k) = (0, 0)$$
focus = (h,k+p)=(0,1000)
p=1000
z=x$^2/a^2$
x$^2 = a^2 * z$
a$^2 = 4 * p$
a$^2 = 4000$

Taking the focus into account, the parabolic equation so far is the following.

$$z = \frac{x^2}{4000} + \frac{y^2}{4000} \tag{2}$$

The above equation however is for a perfect mirror. In order to have a parabolic mirror with radially outward ripples, the cosine of the radius should be added to it.

$$z = \frac{x^2}{4000} + \frac{y^2}{4000} + A * cos(w * \sqrt{x^2 + y^2}) \tag{3}$$

The above is the final equation that will be used to represent the parabolic mirror where A is the amplitude of the ripples and w is the angular frequency of the ripples. Note that if the frequency is $6 * \pi$, this means that there are three ripples over one millimeter. Because the diameter of the mirror is 1000 millimeters, the z axis is bounded in between 0 and 62.5 millimeters after plugging in let's say $x = 500\,mm$ and $y = 0\,mm$. Now that an equation has been reached to represent the surface, a singular ray can be traced. The process for this singular ray can be used for multiple rays at once using MATLAB or another programming language. An example of such code can be found in the Appendix.

Using vector calculus, a ray can be mimicked as a vector passing perpendicularly through a point on the XY plane where $x = 0$, $y = 0$, and $z = 0$ is the mirror's vertex. To explain through example, a ray's vector will be $< 0, 0, -q >$ and will pass through an arbitrary point (100,100,0). This can be stated as a 3D line in the following format. The value of q will be calculated later.

$$r =< a, b, c > t + (x, y, z) \tag{4}$$

$$r =< 0, 0, -q > t + (100, 100, 0)$$

To find the reflected ray after coming in contact with the surface, the initial ray must be reflected across the normal of the tangent plane of the particular point in the mirror. In doing so, the angle of incidence will equal the angle of reflection. Furthermore, the reflected ray must have the same magnitude as the original ray in order to get correct results.

To find the normal of the tangent plane, the gradient of the surface must be calculated.

$$z = \tfrac{x^2}{4000} + \tfrac{y^2}{4000} + A * cos(w * \sqrt{x^2 + y^2})$$
$$\text{f}=\text{x}^2/4000 + y^2/4000 + A * cos(w * \sqrt{x^2 + y^2}) - z$$
$$\text{n}=\nabla f =< \tfrac{\delta f}{\delta x}, \tfrac{\delta f}{\delta y}, \tfrac{\delta f}{\delta z} >$$

$$n =< \tfrac{x}{2000} - \tfrac{A*w*x*sin(w*\sqrt{x^2+y^2})}{\sqrt{x^2+y^2}}, \tfrac{y}{2000} - \tfrac{A*w*y*sin(w*\sqrt{x^2+y^2})}{\sqrt{x^2+y^2}}, -1 >$$

After plugging in the particular point of interest, $x = 100\,mm$ and $y = 100\,mm$, and letting $A = .0005\,mm$ and $w = 2 * \pi$, one gets the following vector for the normal of the tangent plane: $n =< 0.04895\,mm, 0.04895\,mm, -1\,mm >$ Using the vector with the exact opposite signs,
$n =< -0.04895\,mm, -0.04895\,mm, 1\,mm >$, yields correct results too. It is important to note that the z component of the normal vector will always be 1 for a paraboloid in this situation.

To find the reflected ray, one can think of the normal as the sum of the initial ray and the reflected ray with equal magnitude.

$$incident\,ray\ +\ reflected\,ray\ =\ normal\,vector\ \ < 0, 0, q > + < x, y, z >=< a, b, 1 >$$
$$\text{solve}\ for\ reflected\,ray\ \ < x, y, z >=< a, b, 1 - q >$$

Since the magnitudes of both the reflected ray and incident ray are equal, one can solve for q from the incident ray.

$$\sqrt{0^2 + 0^2 + q^2} + \sqrt{a^2 + b^2 + (1 - q)^2} = 2q$$
$$\text{a}^2 + b^2 + 1 - 2q + q^2 = q^2$$

One can use the following general equation to solve for the z component of the incident ray where a and b are the x and y components of the normal vector, respectively.

$$q = \frac{a^2 + b^2 + 1}{2} \tag{5}$$

Continuing the previous example where the incident ray hits the point on the surface that has a normal of $n =< -0.04895\,mm, -0.04895\,mm, 1\,mm >$, the following gives the value of the incident ray's vector.

$$q = \frac{(-0.04895)^2+(-0.04895)^2+1}{2} = 0.5024\,mm$$
$$\text{incident } ray \ = < 0, 0, 0.5024 >$$

It is important to note one would get the same results if one used the exact of opposite signs of the incident ray vector while using the original results of the normal vector.
Given the incident ray vector and normal vector, the reflected ray vector is the following.

$$< x, y, z >=< a, b, 1 - q >=< -0.04895, -0.04895, 0.4976 >$$

This reflected ray vector will be one component to find the equation of a 3D line along with the fact that the line passed through the point $x = 100\,mm$ and $y = 100\,mm$ on the surface. To find the z coordinate, just plug in $x = 100\,mm$ and $y = 100\,mm$ into the surface equation. Using this 3D line, one can calculate at which point the reflected ray hits the focal plane, $z = 1000\,mm$. Below is the 3D line representing the reflected ray.

$$r =< -0.04895, -0.04895, 0.4976 > t + (100, 100, 4.9996)$$

Using this line, t is solved for when $z = 1000\,mm$.

$$1000 = 0.4976t + 4.9996$$
$$\text{t=1999.6}$$

Plugging in $t = 1999.6$, one can see where the reflected ray will hit the focal plane.

$$x = -0.04895t + 100$$
$$\text{x=-0.04895(1999.6)+100}$$
$$\text{x=2.12}\,mm$$

$$y = -0.04895t + 100$$
$$\text{y=-0.04895(1999.6)+100}$$
$$\text{y=2.12}\,mm$$

4.2 Encircled Energy

Using the procedure defined in the subsection above, one can use a programming language such as MATLAB to do such a procedure thousands of times and obtain results as to how big a detector at the focal point must be to capture, let's say, 90% of the rays. To do so in code, start off with a detector size with a radius of zero and tally the number of reflected rays that falls in between such a radius. Find the proportion of rays that hit the detector over the number of total rays and see if the percentage is at least 90%. If it is not, slowly increment the radius by let's say 0.0001 mm. The smaller the increment, the more accurate are the results, but also it will take a lot longer. Also, the more rays in the calculation, the more accurate are the results too, but it will also take longer. Finally, once the proportion of reflected rays that have hit the detector is at least 90%, one has obtained the necessary radius.

4.3 Point Spread Function

A point spread function shows the intensity of light hitting the focal plane. The best way to come up with such a graph in MATLAB is just simply using a 3D histogram. Every time a reflected ray hits a certain spot on the focal plane, the histogram column will increase in height at the specific area the spot is in. The best way to normalize the point spread function, or PSF for short, is the have the peak equal the Strehl ratio.

4.4 Strehl Ratio

The Strehl ratio is the ratio between the peak focal spot irradiance of the mirror of study and the diffraction-limited peak irradiance. The most practical way of calculating the Strehl ratio is by using the **optical path difference-based root mean square wavefront error**, or **RMS OPD** for short. The RMS OPD is the average optical path difference between a perfect mirror and the actual mirror in units of wavelengths. To calculate it, find the distance between a point an incidence ray hit and the focal point and the distance between the same point the incident ray hit and the point on the focal plane that the reflected ray hit. Do this for virtually every spot on the mirror. Using the following equation, the RMS OPD, in units of wavelengths, can be calculated where λ is the wavelength of light and N is the number of reflected rays. Make sure all units are the same.

$$RMS\,OPD = \sqrt{\frac{(\Sigma((D_{perfect} - D_{actual})^2))}{N}} * \frac{1}{\lambda} \tag{6}$$

Using the RMS OPD, one can find a very close approximation for the Strehl ratio.

$$Strehl\ Ratio \approx e^{-(2*\pi*RMS\,OPD)^2} \tag{7}$$

5 OSLO Approach

Using OSLO, the deformations are on a larger scale and not as fine as the previous section above. However, both have their place in research. This will not be a tutorial on how to make a mirror in OSLO, but rather how to perform simulations on the given mirror such as finding the encircled energy function, point spread function, and Zernike polynomials after varying temperature, pressure, and coefficients of deformation. If one were looking for an OSLO tutorial, see the footnote[1].
The following is the spreadsheet of the mirror that will be worked on throughout this paper. The parameters of the mirror are the same as the previous section with a focal length of 1000 millimeters and a diameter of 1000 millimeters.

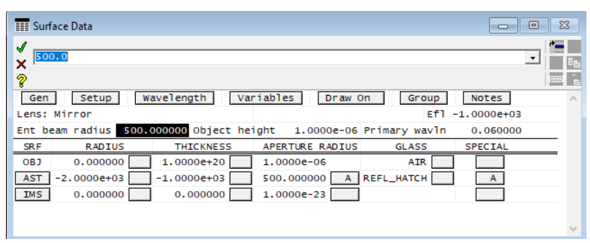

Figure 3: Spreadsheet of Example Mirror

This particular mirror was built in a temperature of 26° Celsius and in a pressure of 0 atmospheres. One can set up their mirror in a certain temperature and pressure by clicking on "Gen" in the spreadsheet and editing those parameters.

5.1 Ray Tracing

The wavelength of light that will be used is 60 nm. This can be edited in the spreadsheet in the area that reads "Primary waveln" and also by clicking the "wavelength" button and adding the desired wavelength.

[1] OMLC https://omlc.org/classroom/oslotut/index.html

5.1.1 Point Spread Function, Encircled Energy, and Zernike Polynomials

In order to evaluate the point spread function of a mirror, select Evaluate> Spread Function> Plot PSF Map/Contour. Another set of options should pop up where one should select "Monochromatic" making sure it is set to the desired wavelength and "Direct Integration". Below is how the spreadsheet should look like.

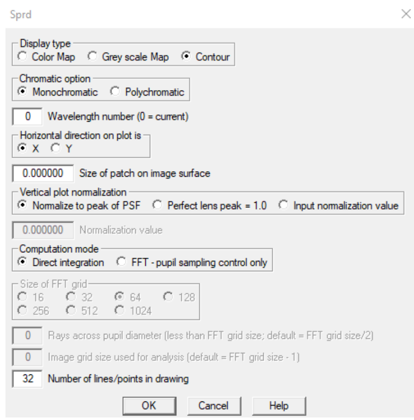

Figure 4: Spreadsheet of PSF Evaluation

After pressing "OK", a graph like the following should appear. The most important aspect of this graph is most likely the Strehl ratio in the top-left hand corner where it says "Peak".

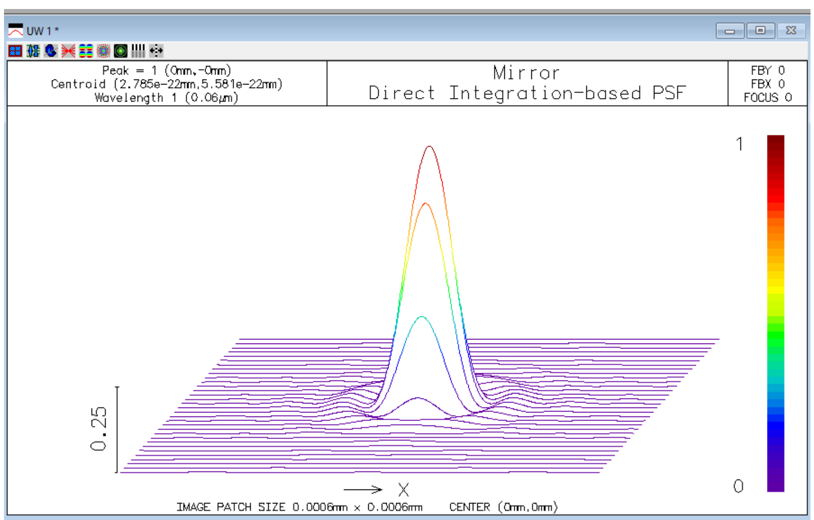

Figure 5: Point Spread Function Example

In order to calculate the encircled energy on the focal plane, one must click on Evaluate> Energy Distribution> Diffraction. Once again, another spreadsheet will pop up, and the user will need to press "Monochromatic" and confirm the desired wavelength is being used. The information can either be outputted on a plot or printed out numerically. This option is up to the user. Leave the rest of the spreadsheet alone. Then, press "OK" to retrieve the results.

In order to find the Zernike polynomials that best describes a surface, click Evaluate> Other Aberrations> Zernike Wavefront. Select "Zernike analysis of wavefront", "Minimum RMS point on surface", and "Monochromatic" and confirm the desired wavelength is being used. Leave the rest of the spreadsheet alone. After pressing "OK", this should give the Zernike polynomials along with an RMS OPD in units of wavelengths. It's important to note that the Zernike polynomials given, especially if given a little amount of polynomials, is only an approximation of the surface and not an exact replica. Therefore, do not be worried if the RMS OPD does not match exactly to another source. Another way of obtaining the RMS OPD is to do the same thing done previously in this paragraph, but instead of selecting "Zernike analysis of wavefront", select "Wavefront statistics". This will also give you the Strehl ratio.

5.1.2 Varying Temperature, Pressure, and Coefficients of Deformation

In order to see effects of temperature and pressure on the surface, make sure to have the default temperature and pressure set before even building the surface. Also, set a thermal coefficient of expansion for the surface by clicking the button in the Special column and the AST row and selecting Surface Control (F)> General. Once the temperature and pressure are set, one can proceed to make a mirror. There are several ways to see the effects of temperature and pressure on the surface. One way can be by taking note of the PSF before and after the temperature and pressure change or another way can be by taking note of the encircled energy before and after the temperature and pressure change. Let's say one wanted to compare the Strehl ratio before and after a temperature change, take note of the Strehl ratio in the default state, then select "Gen" in the spreadsheet of the mirror and change the temperature. Proceed to calculate the Strehl ratio once more. If one wanted to instead compare the effects of a pressure change, it is the same procedure as above but, instead of varying temperature, pressure is varied.
Varying the coefficients of deformation is a little different. Before getting into how to vary the coefficient of deformation, one should understand where these coefficients come from. The following is a snapshot from page 130 from the OSLO optics reference which can be found in the previous footnote.

Polynomial aspherics

There are a variety of forms of aspheric surface (more complicated than conic sections) used in optical design. Most of these are written as polynomials, and are described in the OSLO Help system. The most common is an aspheric surface of the form

$$z = \frac{\mathbf{cv}r^2}{1+\sqrt{1-\ \mathbf{cv}(\mathbf{cc}+1)r^2}} + \mathbf{ad}r^4 + \mathbf{ae}r^6 + \mathbf{af}r^8 + \mathbf{ag}r^{10} \tag{6.5}$$

which OSLO calls a standard asphere.

In a standard asphere the aspheric constants give the sag of the surface in terms of its departure from a conic, not necessarily a sphere. If the conic constant is –1, then

$$z = \frac{1}{2}\mathbf{cv}r^2 + \mathbf{ad}r^4 + \mathbf{ae}r^6 + \mathbf{af}r^8 + \mathbf{ag}r^{10} \tag{6.6}$$

Figure 6: Snapshot from OSLO Optics Reference

The coefficients of deformation affect the mirror on a larger scale than on a finer, smaller scale. One can see visually the effects of the coefficients of deformation by playing with graphing websites or calculators. In order to see the effects of the change in the coefficients of deformation, begin with taking note of either the PSF, encircled energy, or other calculations beforehand. Then, click on the button in the Special column and the AST row, select Polynomial Asphere (A)> Standard Asphere. Here, you can manipulate the mirror on a larger scale. After editing the coefficients, calculate the previous evaluation and compare with the result from before.

6 Data for Mirror with Focal Length and Diameter of 1 Meter

Using the procedures outlined above, below is the data for a mirror with a focal length and diameter of 1000 millimeters using light with wavelength of 60 nm.

6.1 Finer Imperfections

Table 1: Data after varying the angular frequency of ripples 1 μm in height peak to trough; Note: a ω of $N\ \frac{rad}{mm}$ means N ripples per one millimeter.

ω (rad/mm)	RMS OPD (wavelengths)	Strehl	Radius for 90% Capture of Rays at Focal Plane (mm)
$\pi/20$	681	0	.17
$\pi/10$	1284	0	.33
$\pi/2$	6599	0	1.65
2π	26984	0	6.6
6π	80760	0	19.8
10π	133713	0	32.9

Table 2: Data after varying the angular frequency of ripples 2 μm in height peak to trough; Note: a ω of $N\ \frac{rad}{mm}$ means N ripples per one millimeter.

ω (rad/mm)	RMS OPD (wavelengths)	Strehl	Radius for 90% Capture of Rays at Focal Plane (mm)
$\pi/20$	1362	0	.34
$\pi/10$	2569	0	.66
$\pi/2$	13198	0	3.29
2π	53945	0	13.2
6π	161735	0	39.6
10π	267167	0	65.7

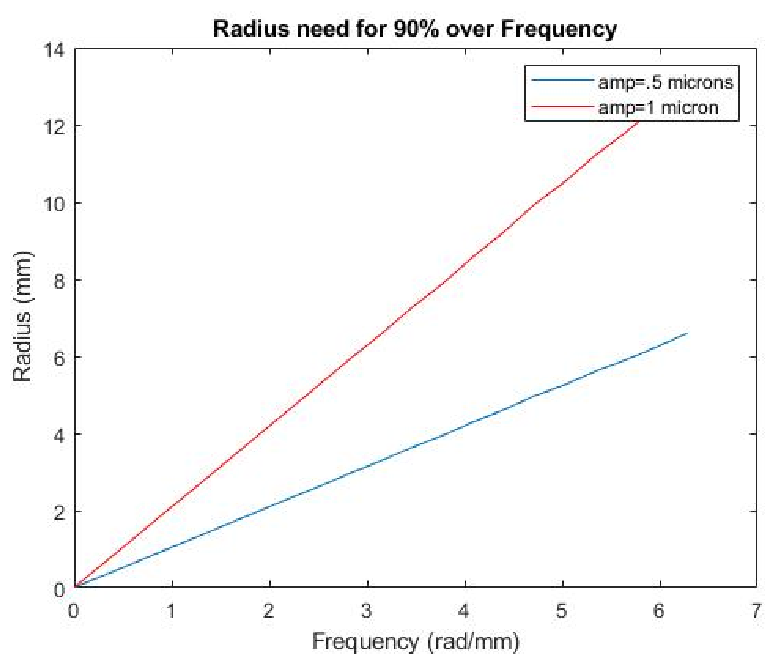

Figure 7: Radius over Frequency

6.2 Larger Deformations

The parameters of the mirror are the same as above, but one aspect that should be noted is that the thermal coefficient of expansion is $6.6 * 10^{-6}/Celsius$. However, when the temperature varied $\pm 2^{\circ}$ $Celsius$, there is no noticeable effect on the surface.

Table 3: Data after varying the 4th Order Coefficient of Deformation

Coefficient of Deformation	RMS OPD (wavelengths)	Radius for 90% Capture of Rays at Focal Plane (mm)
1e-18	0.000614	0.000125
1e-17	0.00614	0.000126
1e-16	0.0614	0.000166
2e-16	0.123	0.000225
3e-16	0.184	0.000291
4e-16	0.246	0.000361
5e-16	0.307	0.000432
1e-15	0.614	0.000791

Only the coefficient of deformation is being varied directly, so therefore Figure 10 below just shows the correlation between the RMS OPD and the radius needed for a detector to capture 90% of the rays at the focal plane.

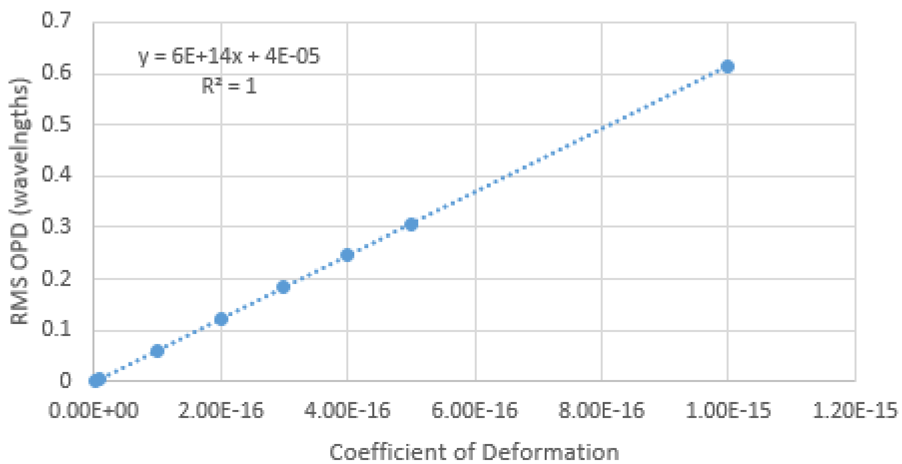

Figure 8: RMS OPD vs Coefficient of Deformation

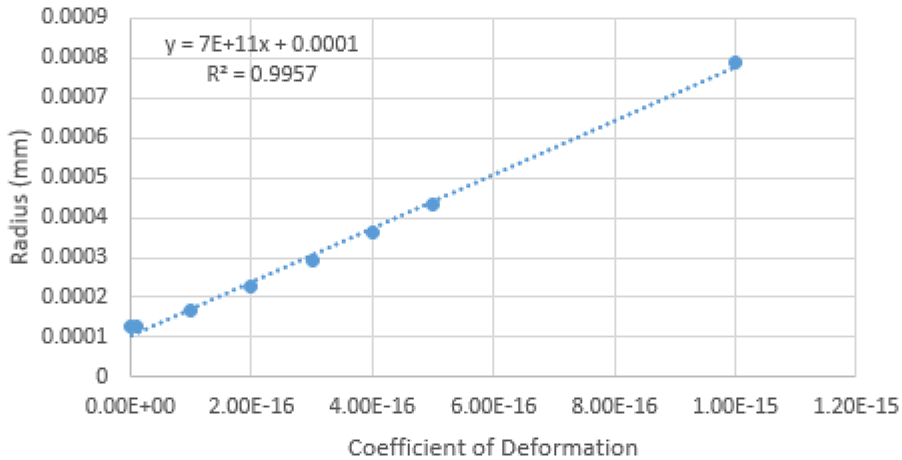

Figure 9: Radius vs Coefficient of Deformation

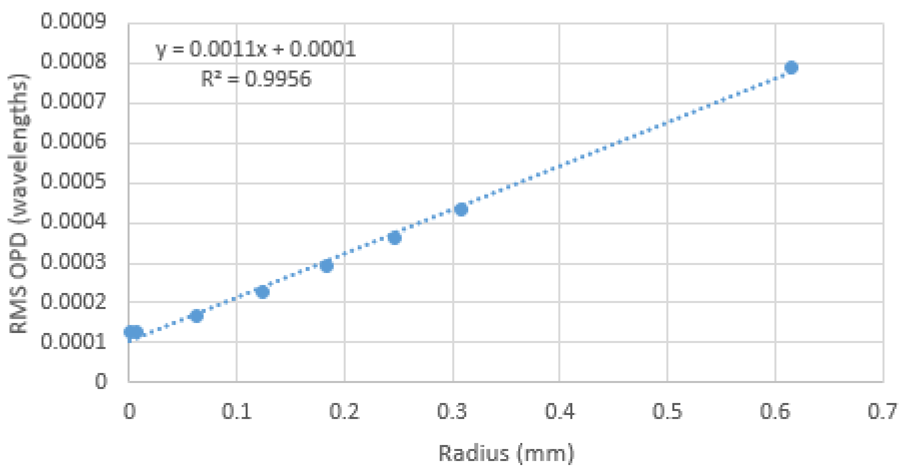

Figure 10: RMS OPD vs Radius

7 Conclusion

When the ripples were 1 μm in height from peak to trough and had an angular frequency of 2π, the radius calculated was 6.6 mm. When the ripples were 2 μm in height from peak to trough and had an angular frequency of 10π, the radius calculated was 65.7 mm. When deformations on a larger scale were implemented, the determined radius of the detector was a lot smaller to the scale of just a few microns. Such data suggests that the radius of the detector at the focal plane depends largely on the small-scale surface roughness than it does on the larger deformations. The radius of a detector at the focal plane of a deployable mirror only needs to be a few centimeters at most depending on the surface roughness to capture 90% of ultraviolet rays which seems like a feasible task. Because one can order a custom-made microchannel plate detector with a diameter of about 130 mm, it seems possible that one can build a deployable mirror that works at normal incidence, after coating, in the extreme ultraviolet radiation range.

8 Appendix: MATLAB Code

```
%%%%%%%%AMPLITUDE and FREQUENCY of ripples in mm and rad/mm%%%%%
three_d_amp = .0005;
freq = 2*pi; %angular frequency
lambda = 60; %in nm
%%%%%%%%%%%%%%%%%INITIAL LIGHT RAYS%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
three_d_rays = [];
for i = -500:10:500
    for j = -sqrt(500^2-i^2):10:sqrt(500^2-i^2)
        three_d_rays = [three_d_rays; i j];
    end
end
radius = sqrt(three_d_rays(:,1).^2+three_d_rays(:,2).^2);
%%%%%%%%%%%%%%FIND NORMAL OF TANGENT PLANES%%%%%%%%%%%%%%%%%%%%%
n = [];
for i = 1:size(three_d_rays,1)
    n = [n;-1*(1/2000)*three_d_rays(i,1)+three_d_amp*sin(freq*radius(i))*freq*three_d_rays(i,1)/radius(i), -1*(1/2000)*three_d_rays(i,2)+three_d_amp*sin(freq*radius(i))*freq*three_d_rays(i,2)/radius(i), 1];
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%FIND RAY Z COMPONENT%%%%%%%%%%%%%%
z = [];
for i = 1:size(three_d_rays,1)
    z = [z; (n(i,1)^2+n(i,2)^2+1)/2];
end
%%%%%%%%%%%%%%%%%%%FIND REF VECTOR%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%subtracting normal and ray vector
ref_vec = [];
for i = 1:size(n,1)
    ref_vec = [ref_vec; n(i,1), n(i,2), n(i,3)-z(i)];
end
%%%%%%%%%%%%%%%%%%%%%%%%%FIND T WHEN Z=1000%%%%%%%%%%%%%%%%%%%%%%%%
%3D LINE%
t = [];
for i = 1:size(ref_vec,1)
    t = [t; (1000-((1/4000)*three_d_rays(i,1)^2+(1/4000)*three_d_rays(i,2)^2+three_d_amp*cos(freq*radius(i))))/ref_vec(i,3)];
end
%%%%%%%%%%%%%%%%%%%%%%%%%%PLOT AT T%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
coords = [];
for i = 1:length(t)
    coords = [coords;three_d_rays(i,1)+t(i)*ref_vec(i,1) three_d_rays(i,2)+t(i)*ref_vec(i,2)];
end
figure(4)
plot(coords(:,1),coords(:,2), '*')
xlabel('mm')
ylabel('mm')
title('Focal Plane Scatterplot')
```

```
figure(5)
hist3([coords(:,1),coords(:,2)],'CDataMode','auto','FaceColor','interp','nbins',[25 25]);
colorbar
xlabel('mm')
ylabel('mm')
title('Point Spread Function')

percentage = 0;
max_rad = 0;
while percentage < .9
    %YOU CAN CHANGE RADIUS INCREMENT SIZE HERE
    max_rad = max_rad + 1e-2;
    fit_coords = [];
    for i = 1:size(coords,1)
        if sqrt(coords(i,1)^2+coords(i,2)^2)< max_rad
            fit_coords = [fit_coords; coords(i,1),coords(i,2)];
        end
    end
    percentage = size(fit_coords,1)/size(coords,1);
end
percentage = percentage * 100;
fprintf("A radius of %f mm will absorb %f percent of the rays.\n", max_rad, percentage)

figure(6)
plot(fit_coords(:,1),fit_coords(:,2), '*')
xlabel('mm')
ylabel('mm')
title('90% Fit Focal Plane Scatter Plot')

%%%%%%%%%%%%%%%%%%%%%%%%%RMS OPD and STREHL%%%%%%%%%%%%%%%%%%%%%
surf = [];
for i = 1:size(three_d_rays,1)
    surf = [surf; three_d_rays(i,1) three_d_rays(i,2) (three_d_rays(i,1)^2+three_d_rays(i,2)^2)/4000 + three_d_amp*cos(radius(i))];
end

%Perfect Distance
p_dist = [];
for i = 1:size(surf,1)
    p_dist = [p_dist; sqrt((0-surf(i,1))^2+(0-surf(i,2))^2+(1000-surf(i,3))^2)];
end

%Actual Distance
a_dist = [];
for i = 1:size(surf,1)
    a_dist = [a_dist; sqrt((coords(i,1)-surf(i,1))^2+(coords(i,2)-surf(i,2))^2+(1000-surf(i,3))^2)];
end
```

```
%path displacement
dis = [];
for i = 1:length(a_dist)
    %convert distance to nm from mm
    dis = [dis; abs(p_dist(i)*1000000-a_dist(i)*1000000)];
end


%RMSE
sub = (dis).^2;
for i = 1:length(sub)
    if isnan(sub(i))
        sub(i) = 0;
    end
end

%in units of nm (average path diff)
rms = sqrt(sum(sub)/length(a_dist));

%average path displacement in wavelengths
rms_wavelengths = rms/lambda;
strehl = exp(-(2*pi*rms_wavelengths)^2);
fprintf("This mirror has a Strehl ratio of %f and an RMS OPD of %f wavelengths
    .\n", strehl, rms_wavelengths)
```