

# Final Report – Planar Concave Grating Design

---

*Student: Lukas Janavicius*

*NetId: janavic2*

*Professor: Peter Dragic*

ECE 495

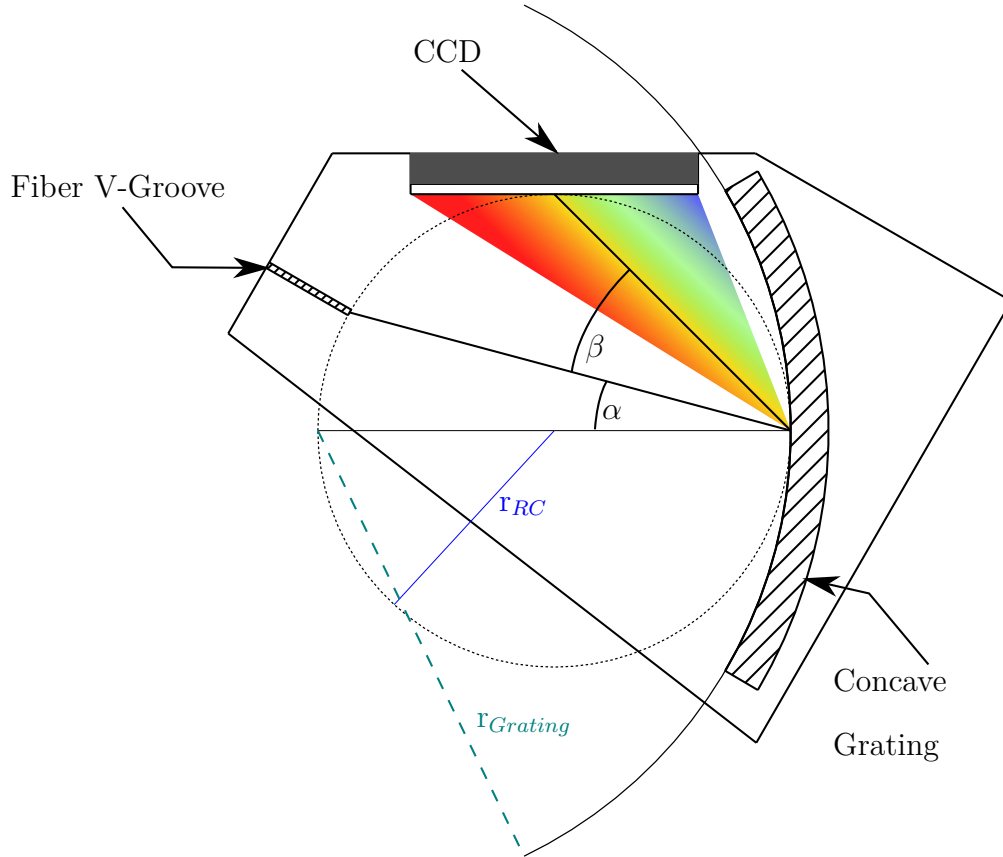
Spring 2020

# Contents

<b>1</b>	<b>Abstract</b>	<b>1</b>
<b>2</b>	<b>Introduction and Background</b>	<b>2</b>
2.1	Basis for Design . . . . .	2
2.2	Verification of Elliptical Design . . . . .	3
2.2.1	Equations . . . . .	3
<b>3</b>	<b>Implementation</b>	<b>4</b>
3.1	Algorithm . . . . .	4
3.2	Remaining Work . . . . .	7
3.3	Discussion . . . . .	7
<b>4</b>	<b>Results</b>	<b>8</b>
4.1	Discusssion . . . . .	11
	<b>References</b>	<b>12</b>

# 1 Abstract

In this work I present a novel method for programmatically generating planar concave gratings based on concentric ellipsis. Although I was unsuccessful in simulating the device with FDTD, the implemented algorithm dynamically generates and visualizes any valid grating configuration. Additionally, I propose a pure python algorithm for closed-loop generation of these gratings for a wide breadth of applications. This algorithm is fully compatible with the required file format for the silicon photonics lab, so future students can use this work as a scaffold for their own exploration of planar gratings.



## 2 Introduction and Background

### 2.1 Basis for Design

The foundational work for this design comes from P. Pottier et al. [1]. The reference design, is displayed in Figure 1, the left image is a traditional planar Rowland grating, whereas the right is a mono-order distributed Bragg reflector grating. The design presented in this report follows the latter, as it offers greater efficiency than the traditional grating, and its construction lends itself to programmatic generation.

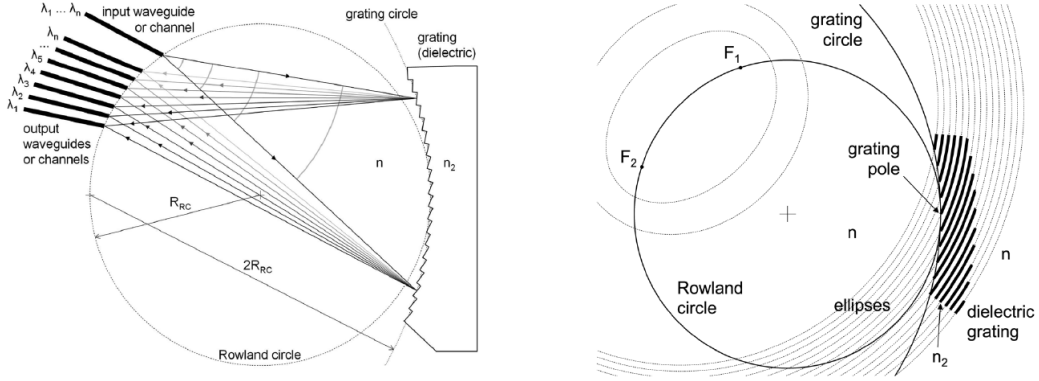


Figure 1. Reference grating geometries, reproduced from P. Pottier et al. [1].

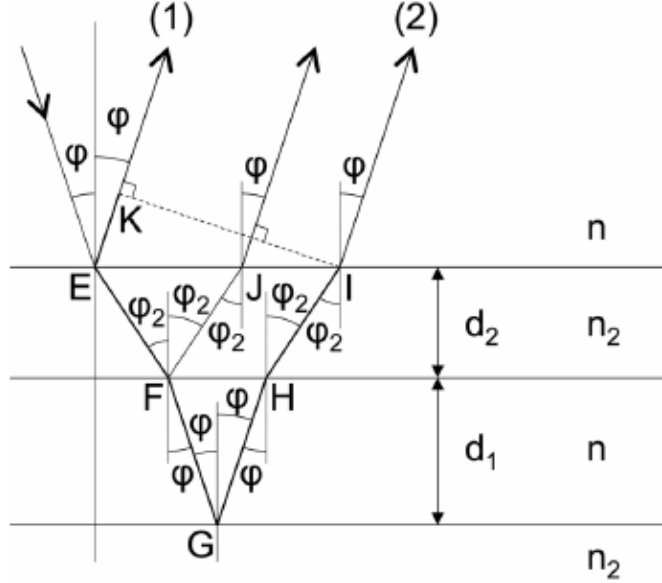


Figure 2. Bragg condition for two material stack [1].

## 2.2 Verification of Elliptical Design

The following equations describe the various geometrical and optical condition for optimal device performance [1, 2]. As discussed in the work of P. Pottier et al. [1], drawing ellipsis who's foci are the input and output points with thickness  $d_2$  offset from one another by a distance  $d$  (the center to center distance), satisfies the diffraction conditions and the blazing condition. However, as discussed below, the bragg reflector stack cannot be fabricated in an ideal manner. Despite this, if the ellipse thickness is much smaller than the spacing between the ellipsis, that is  $\frac{d_2}{d} \approx 0$ , the device's efficiency is maximized.

### 2.2.1 Equations

To improve the device's efficiency, the grating may be curved. The radius of the grating must be  $2R_{RC}$ , so that it focuses the input light into the outputs placed around the Rowland Circle. It's worth noting that the radius does not particularly matter, for the purposes of this design

$$\mathbf{M}\lambda = -na(\sin \alpha + \sin \beta) \quad (1)$$

We also note,

$$\sin \theta = \frac{d}{a}$$

The second diffraction condition comes from finding the path length difference  $\delta$ ,

$$m\lambda = 2 \left[ nd_1 \cos \phi + n_2 d_2 \sqrt{1 - \left(\frac{n}{n_2}\right)^2 \sin^2 \phi} \right]$$

To select our pass-band center, the angle of the grating lines with respect to the grating normal,  $\phi$ , should obey Snell's law. That is,

$$\begin{cases} \alpha - \phi + \theta = 0 \\ \alpha - \beta = 2\phi \end{cases}$$

Rearranging the above conditions yield the following critical relation, so if  $a$  is found using equation 1, the center-to-center distance  $d$  is

$$d = -a \sin \left( \frac{\alpha + \beta}{2} \right) \quad (2)$$

The optimal Bragg reflector dimensions are

$$\begin{cases} d_1 = \frac{(2k_1 + 1)\lambda}{4n \cos \phi} \\ d_2 = \frac{(2k_2 + 1)\lambda}{4n_2 \cos \phi_2} \\ k_1, k_2 \in \mathbb{Z} \end{cases}$$

It is very difficult to select dimensions such that the above conditions are met; however, if  $\frac{d_2}{d} \approx 0$  or  $d_1 = d_2$  and  $n_2 \approx n$  then the path length difference from both conditions is approximately equal [1, 2]. Therefore  $m = -M$ , which optimizes device efficiency.

### 3 Implementation

The entirety of the design work takes place in the python programming language; specifically, using the geometric package Shapely. Shapely is a BSD-licensed Python package for manipulation and analysis of planar geometric objects [3]. The key advantage of using shapely to define photonic circuit elements is the native support for gdshelpers, another package which takes shapely polygons and curves and generates photonic devices [4]. From MZIs, waveguides, couplers, and resonators the gdshelper package offers many tools for creating GDSII files, which are standard for lithography tools. Furthermore, the open source FDTD package, Meep, offers a GDSII importer for simulations [5]. Although I did not generate mask files or simulate the structure, which is discussed further below, it is possible to implement the closed-loop algorithm in Figure 5 entirely in one script.

#### 3.1 Algorithm

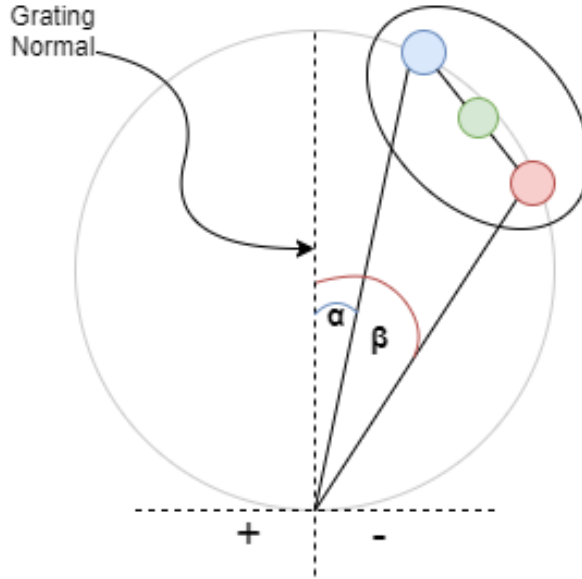


Figure 3. Grating geometric construction, the blue point is the input waveguide, the red point is the center of the output band, and the green point is the ellipse center. Also shown is the grating angle sign convention, an angle in the first quadrant is negative.

For the geometry configuration in Figure 3, after solving the ellipse offset,  $d$ , in Equation 2, the grating ellipsis are generated with the following code

Listing 1. Ellipse Class

---

```

class GratingEllipse(ParametricGeometry):
    max_radius = 0

    def __init__(self, focus_0, focus_1, offset=0, origin=None):
        super().__init__(origin)
        A, B = (focus_0.u, focus_0.v), (focus_1.u, focus_1.v)
        # x axis angle, needed for ellipse orientation
        theta = atan2(B[1]-A[1], B[0]-A[0])
        center = focus_0.midpoint(focus_1)
        self.u, self.v = center[0], center[1]

        foci_spacing = sqrt((B[1]-A[1])**2 + (B[0]-A[0])**2)
        major_axis = foci_spacing+2*offset
        minor_axis = sqrt(major_axis**2-foci_spacing**2)
        from shapely import affinity

        # create a circle of radius 1
        curve = Point(self.u, self.v).buffer(1)
        # create the ellipse by scaling the circle along x and y:
        curve = affinity.scale(curve, major_axis, minor_axis)

        # Rotate the ellipse (clockwise, x axis pointing right):
        self.curve = affinity.rotate(curve, theta, use_radians=True).boundary

```

---

The final remaining unknown variable is the Rowland radius. The actual value of the radius does not impact device performance so long as the radius is greater than a few microns and the input beam divergence is relatively small [1]. For example, the spectra in Figure 4 are produced by a device with  $R_{RC} = 50 \mu m$ . Given this freedom, and that the linear dispersion scales directly with the radius of a concave grating,  $R_{RC}$  should be scaled to fit the free spectral range of the device. However, there is no empirical equation for the FSR because the Bragg reflector limits the pass band. Figure 4 illustrates this by simulating an identical structure with varying reflector dimensions.

Figure 5 illustrates how FDTD may be leveraged to generate the optimal geometry for our device. However, because the Heidelberg maskless aligner in MRL has a minimum feature size of 600 nm, multiple exposures are required to meet the  $\frac{d_2}{d} \approx 0$  approximation. This requirement means that the GDSII files for

simulation and fabrication must be different; ultimately, I chose to simply define the elliptical curves so I could use them as a scaffold for both patterns.

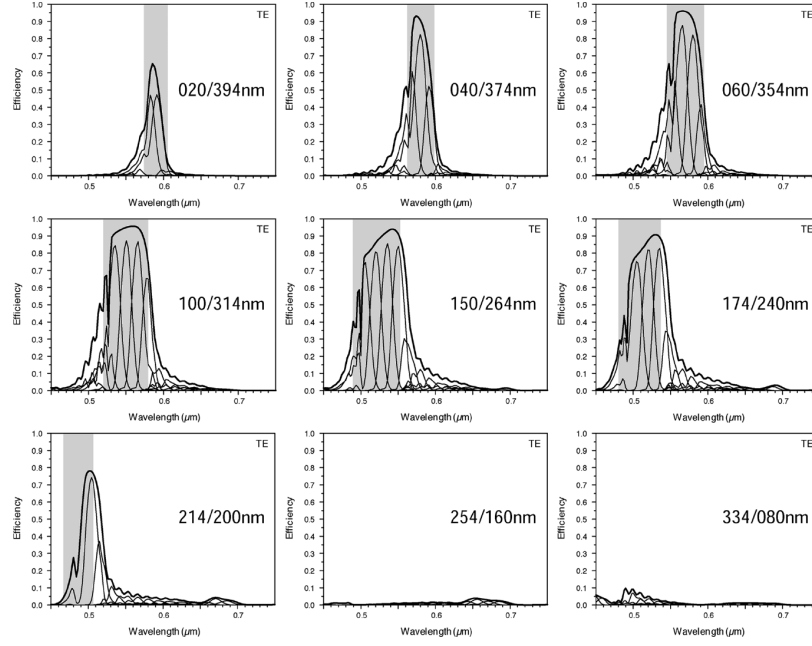


Figure 4. Grating spectrum for varying values of  $\frac{d_2}{d}$ . The bandwidth of the system is bound by the bragg reflector, not the grating itself. Reproduced from P. Pottier et al. [1].

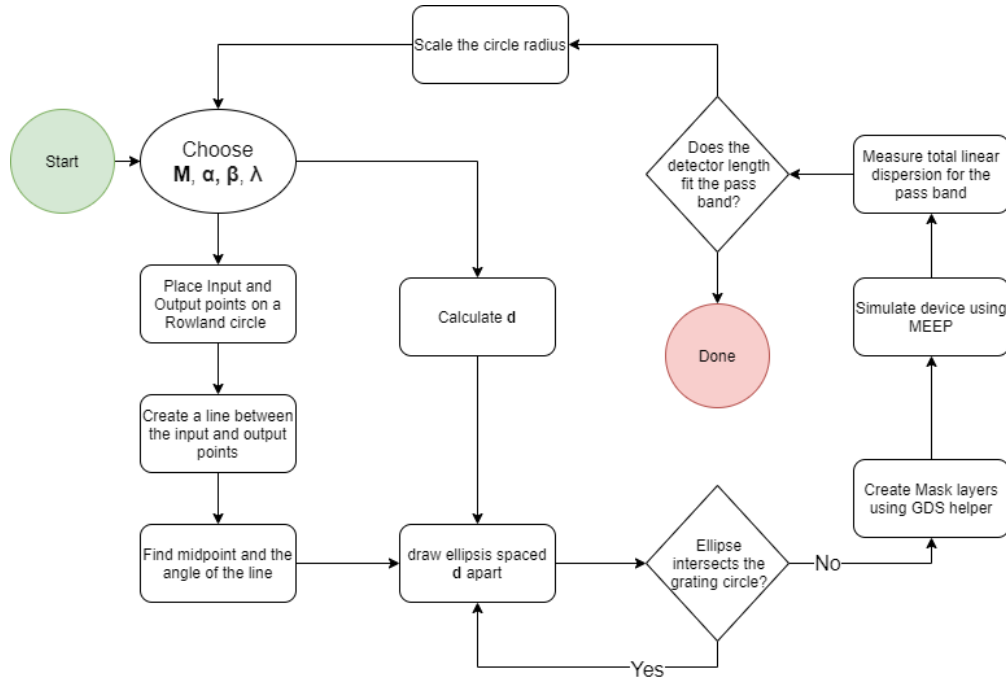


Figure 5. Grating construction algorithm, .



### 3.2 Remaining Work

Unfortunately, the only FDTD experience I have is from this class, so I wasn't able to get the simulations running in this short amount of time. As previously discussed, because the masks for fabrication and simulation must be different I was reluctant to make my script geared towards a single application. To compromise, the ellipses are defined by a center line, which shapely easily can buffer a polygon around.

Ultimately, I still plan to use my experience with microelectronics to fabricate a low-cost spectrometer for my applications. In particular, I am starting my PhD at UIUC next semester, and I would like to add several spectrometers to my reaction chamber. I have spent the last two years developing a python to python/c++ compiler for creating closed loop processing of semiconductors; however, I do not have a characterization device built on the same framework. If you are interested in the future development of my software my GitHub is <https://github.com/LukasJanavicius>, though note that the compiler has not been published for release at this time.

### 3.3 Discussion

I have elected to omit code (with the exception of the ellipsis) from this report for the sake of brevity, but I will provide it alongside my submission. Feel free to use any of the material from this report or the script for future use. Given that gdshelpers has built in tools for generating MZI structures with grating couplers [4], perhaps someone could implement their silicon photonics lab using this script and actually test the device.

I did not have time to completely generate the necessary files for the device; however, I wrote an auxiliary plotting script which accepts a parameter name and a range of values. Note that only one parameter is varied at a time, and the others are held constant at their value in Table 1.

Table 1. Standard grating configuration parameter, unless otherwise specified these parameters are held constant throughout the construction. Also, note that the unit of the Rowland circle and grating thickness is arbitrary, and may be assigned when writing to a GDSII helper.

$n$	$n_2$	$\lambda_{grating}$	M	$\alpha$	$\beta$	$R_{RC}$	$t_{grating}$
1.5	1	600 nm	-2	-15°	-45°	10	3

## 4 Results

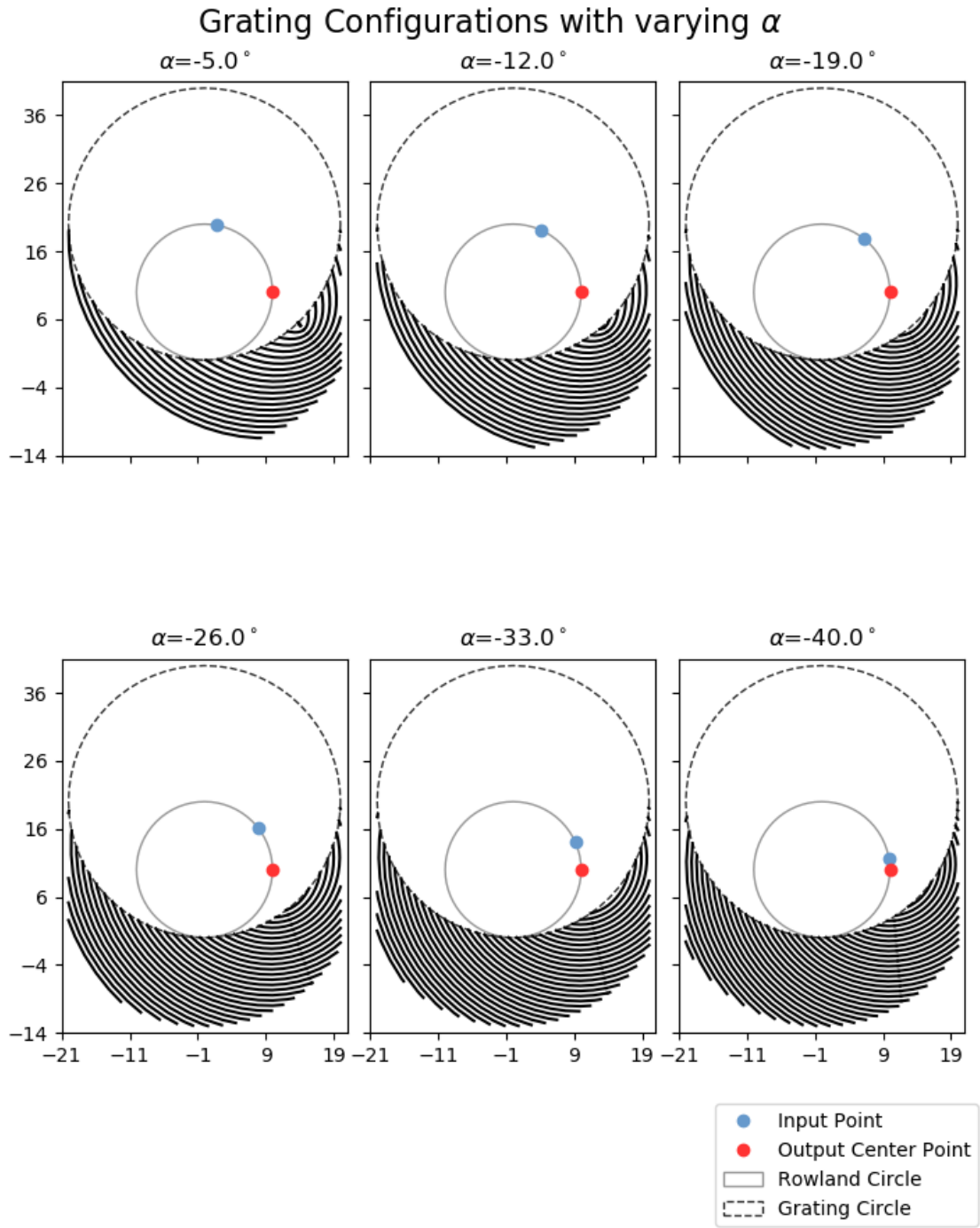


Figure 6. Grating geometries with varying input angle. Note the eccentricity of the ellipsis at large angle differences, and the circularity when approaching the Littrow condition  $\alpha = \beta$ .

## Grating Configurations with Varying $M$

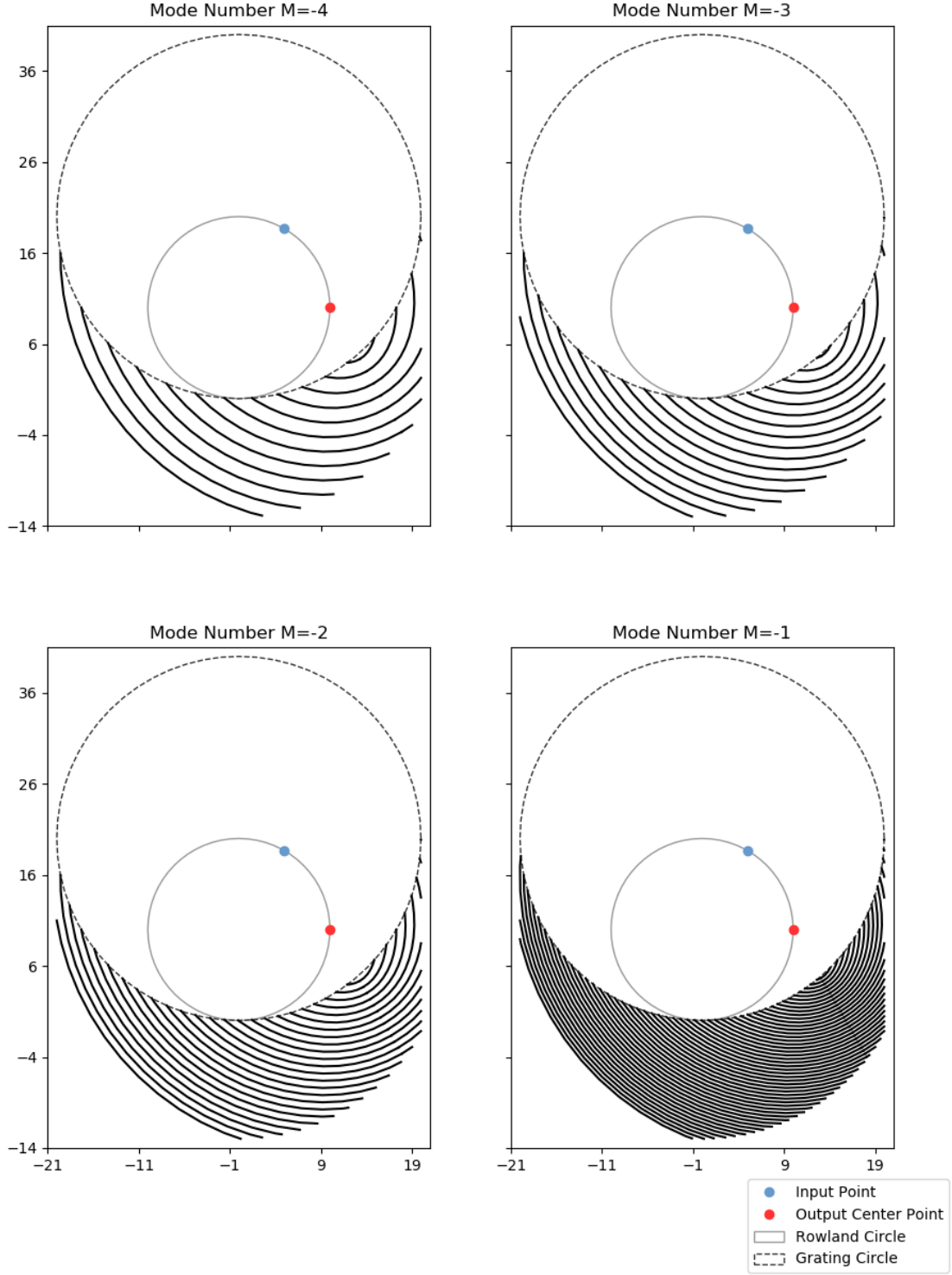


Figure 7. Grating geometries with varying mode number  $M$ . As  $M$  decreases so does the grating spacing.

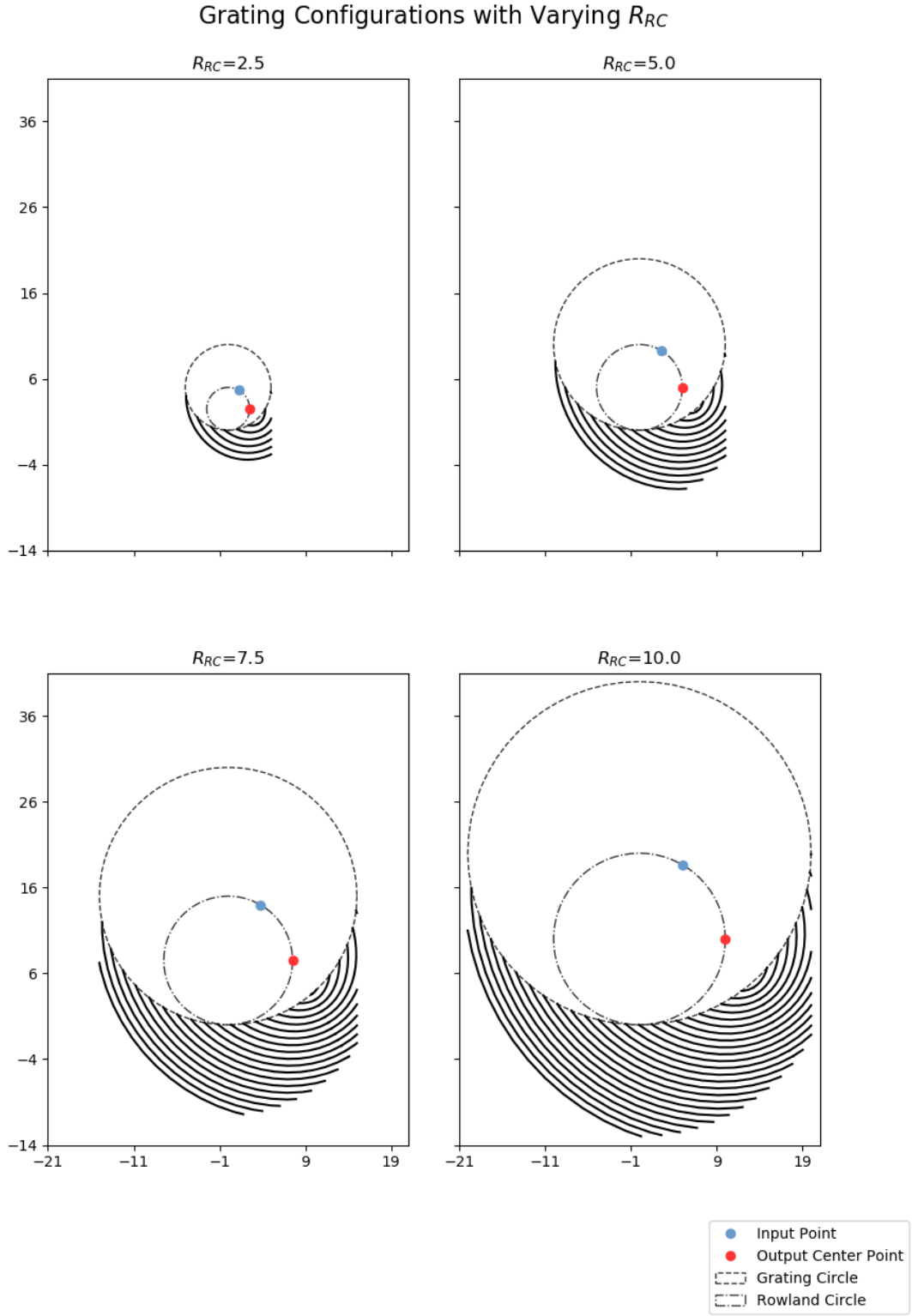


Figure 8. Grating geometries with varying Rowland radius. Grating spacing is independent of  $R_{RC}$ .

Table 2. Grating spacing as a function of the input angle.

$\alpha$	d (nm)
-5.0	389.568
-12.0	386.191
-19.0	382.462
-26.0	378.487
-33.0	374.392
-40.0	370.314

Table 3. Grating spacing as a function of M.

M	d (nm)
-4	769.26
-3	576.945
-2	384.63
-1	192.315

## 4.1 Discussion

Although my script can vary the grating geometry for any parameter in Table 1, Figures 6 through 8 are meant to illustrate the geometry's dependence on parameters which may be freely chosen. The indices of refraction and target wavelength are bound by the material system and application, so are held constant for my purposes.

Tables 2 and 3 show how the design may be tweaked to fit lithographic constraints either by angle or mode number. It is worth noting that, as with a traditional spectrometer, with increasing  $|M|$  the device's resolving power increases, but the free spectral range decreases. For my applications I would like the greatest FSR possible, but as Figure 7 illustrates, low mode numbers require extremely tight patterns. Using the school's photolithography tools I would expect to achieve a minimum  $|M| = 4$ , and with a dedicated mask it should be possible to achieve  $|M| = 2$ . Without simulations or empirical equations for the device's resolving power or FSR I cannot comment on their expected values. However, given the side wall roughness produced when fabricating photonics I fully confident my device's efficiency will be much lower than that reported by P. Pottier et al. [1].

## References

- [1] M. Packirisamy and P. Pottier, “Mono-Order High-Efficiency Dielectric Concave Diffraction Grating,” *Journal of Lightwave Technology*, Vol. 30, Issue 17, pp. 2922-2928, vol. 30, no. 17, pp. 2922–2928, 9 2012.
- [2] P. Pottier, M. J. Strain, and M. Packirisamy, “Integrated Microspectrometer with Elliptical Bragg Mirror Enhanced Diffraction Grating on Silicon on Insulator,” *ACS Photonics*, vol. 1, no. 5, pp. 430–436, 5 2014. [Online]. Available: <https://pubs.acs.org/doi/10.1021/ph400165j>
- [3] “Shapely — Shapely 1.8dev documentation.” [Online]. Available: <https://shapely.readthedocs.io/en/latest/>
- [4] “Welcome to the gdshelpers’ documentation! — gdshelpers 1.1.1 documentation.” [Online]. Available: <https://gdshelpers.readthedocs.io/en/latest/index.html>
- [5] “Manual - MEEP Documentation.” [Online]. Available: <https://meep.readthedocs.io/en/latest/>