

Rochester Institute of Technology

**RIT Digital Institutional Repository**

---

Theses

---

5-8-2023

## Optical Phased Array for lidar systems at blue wavelengths

Long H. Nguyen  
lhn3149@rit.edu

Follow this and additional works at: <https://repository.rit.edu/theses>

---

### Recommended Citation

Nguyen, Long H., "Optical Phased Array for lidar systems at blue wavelengths" (2023). Thesis. Rochester Institute of Technology. Accessed from

This Thesis is brought to you for free and open access by the RIT Libraries. For more information, please contact [repository@rit.edu](mailto:repository@rit.edu).

**Optical Phased Array  
for lidar systems at blue wavelengths**

by

**Long H. Nguyen**

Thesis

Submitted in Partial Fulfillment of the Requirements for the Degree  
of

**MASTER OF SCIENCE IN ELECTRICAL ENGINEERING**

Department of Electrical and Microelectronic Engineering

Kate Gleason College of Engineering

Rochester Institute of Technology, Rochester, NY

Supervised by:

**Dr. Stefan Preble**

May 8, 2023

# Efficient Optical Phased Array for lidar system at blue wavelengths

Approved by:

---

Dr. Stefan Preble, Professor

*Thesis Advisor, Department of Electrical and Microelectronic Engineering*

---

Dr. Gregory Howland, Assistant Professor

*Committee Member, School of Physics and Astronomy*

---

Dr. Jing Zhang, Associate Professor

*Committee Member, Department of Electrical and Microelectronic Engineering*

---

Dr. Ferat E. Sahin, Professor

*Department Head, Department of Electrical and Microelectronic Engineering*

# Thesis Release Permission Form

Rochester Institute of Technology  
Kate Gleason College of Engineering

Title:

Efficient Optical Phased Array  
for lidar system at blue wavelengths

I, Long H. Nguyen, hereby grant permission to the Wallace Memorial Library to reproduce my thesis in whole or part.

---

Long H. Nguyen

---

Date

I dedicate this Thesis to my loving family: my mother (Mẹ) Nguyễn Thúy Hải, my father (Ba) Nguyễn Văn Thanh, and sister Nguyễn Thanh Thảo Linh for the never-ending love and support.

Without them, I could not be in the position I am today.

## Acknowledgements

This thesis marks the end of my incredible 5-year journey at Rochester Institute of Technology where I have grown professionally and personally. There are a number of people whose support is significant to me and I am forever thankful for that.

I thank my Thesis advisor, Professor Stefan Preble, for the research opportunity and for being an inspiring and supportive companion.

I thank Dr. Mishkat Bhattacharya and Dr. Donald Figer for giving me the valuable research experience early in my Bachelor years.

I thank my labmates and soon-to-be Doctors, Vijay and Hector, for their advice and support in the completion of this thesis.

I thank my dear professors, Dr. Howland, Dr. Kurinec, Dr. Pearson and Dr. Patru for eagerly exposing me to the exciting field of electronics and photonics and for becoming my engineering role models.

I thank my close friends Duc, Yen, Tuan, Liam and the RIT Table Tennis club that play a major part in my journey at RIT.

I thank the members of my Thesis Committee, Dr. Howland and Dr. Zhang, for taking their valuable time to review my thesis.

# Abstract

## Efficient Optical Phased Array for lidar system at blue wavelengths

Optical Phased Array has become a popular solution for light detection and ranging (lidar) due to its compact size, steering ability without physical moving parts, and integration with conventional CMOS manufacturing processes. With the rising demand in advanced driver assistance systems (ADAS) and surveying in agriculture and oceanography industries, there is a need for a durable, high performing, and integrable lidar module. For that, a suitable integrated OPA is required to carry sufficient power and have large steering angle at a particular wavelength range. This thesis is specifically focused on blue wavelengths for use in applications such as augmented/virtual reality, displays, oceanography, atmospheric sensing, and quantum information systems based on atoms/ions.

The steering angle of a OPA depends heavily on the distribution of emitters. Ideally, the OPA can steer 180 degrees when the spacing of the array elements are exactly a half wavelength apart. However, at this spacing, fabrication and crosstalk between waveguides become a major issue. Consequently, the emitters must be spaced farther apart, reducing the steering angle due to the creation of side-lobes.

To address this challenge, in this work, we present and demonstrate OPAs that utilize non-periodic element spacings. Specifically, we investigated emitters that are distributed using a Golomb ruler or with random distributions. We were able to achieve a large steering angle at a blue wavelength (450nm) without any emitters at a half-wavelength spacing. We also demonstrate that by varying the widths of the waveguides, crosstalk can be significantly reduced. These methods could be applied to other wavelengths to increase the OPA performance and robustness.

## List of contributions

- Demonstrated the Golomb ruler distribution in OPA for side-lobe suppression and power efficiency.
- Demonstrated randomly distributed OPA emitter array that exhibits large steering angle and low side-lobes:
  - Designed using a fully parameterized Genetic algorithm.
- Investigated the variation of waveguide widths (at 450-nm wavelength) pattern design to minimize crosstalk.
- Implemented the designs on a Silicon Nitride Photonic Integrated Chip:
  - Simulated the core components of the OPA (end-fire waveguide emitter arrays and bends).
  - Completed a GDS tapeout of the chip design with 8 different OPA designs (constant spacing, Golomb, Random, and Varying width waveguide).

# Contents

|          |                                       |           |
|----------|---------------------------------------|-----------|
| <b>1</b> | <b>Introduction</b>                   | <b>1</b>  |
| 1.1      | Light Detection and Ranging . . . . . | 3         |
| 1.1.1    | Mechanism . . . . .                   | 3         |
| 1.1.2    | Applications . . . . .                | 4         |
| 1.1.3    | Components . . . . .                  | 5         |
| 1.2      | Beam Steering mechanism . . . . .     | 7         |
| 1.2.1    | Non-solid-state methods . . . . .     | 7         |
| 1.2.2    | Limitations . . . . .                 | 9         |
| 1.3      | Photonic Integrated Circuit . . . . . | 9         |
| 1.3.1    | Existing OPA . . . . .                | 11        |
| <b>2</b> | <b>Theory</b>                         | <b>14</b> |

|          |                                     |           |
|----------|-------------------------------------|-----------|
| 2.1      | Wave interference . . . . .         | 14        |
| 2.2      | Phased Array Theory . . . . .       | 16        |
| <b>3</b> | <b>Simulation</b>                   | <b>20</b> |
| 3.1      | Simulation Setup . . . . .          | 20        |
| 3.2      | Constant Distribution . . . . .     | 22        |
| 3.2.1    | Ideal Source . . . . .              | 22        |
| 3.2.2    | Linear phase difference . . . . .   | 25        |
| 3.2.3    | Gaussian Distribution . . . . .     | 27        |
| 3.3      | Non-constant Distribution . . . . . | 31        |
| 3.3.1    | Periodic Distribution . . . . .     | 31        |
| 3.3.2    | Golomb Distribution . . . . .       | 32        |
| 3.4      | Random Distribution . . . . .       | 34        |
| 3.4.1    | Genetic Algorithm . . . . .         | 34        |
| 3.5      | Phase Modulation Scheme . . . . .   | 38        |
| 3.5.1    | Theory . . . . .                    | 38        |
| 3.5.2    | Result . . . . .                    | 41        |
| <b>4</b> | <b>Modeling</b>                     | <b>43</b> |

|          |  |           |
|----------|--|-----------|
| 4.1      | Setup . . . . .  | 43        |
| 4.2      | Waveguide . . . . .  | 45        |
| 4.2.1    | Material Consideration . . . . .                             | 45        |
| 4.2.2    | Geometry and Dimension . . . . .                             | 47        |
| 4.3      | OPA Array . . . . .  | 50        |
| 4.3.1    | varFDTD . . . . .  | 50        |
| 4.3.2    | Spacing pattern . . . . .                                    | 52        |
| 4.3.3    | Width pattern . . . . .                                      | 56        |
| <b>5</b> | <b>Layout and Preliminary Experiment on Ring Resonator</b>   | <b>59</b> |
| 5.1      | Layout . . . . .   | 59        |
| 5.1.1    | Tools . . . . .  | 59        |
| 5.1.2    | Photonic chip GDS . . . . .                                  | 60        |
| 5.2      | Preliminary Experimental Result of Blue wavelength . . . . . | 69        |
| 5.2.1    | Ring Resonator . . . . .                                     | 69        |
| 5.2.2    | Experimental Setup . . . . .                                 | 70        |
| 5.2.3    | Resonance Result . . . . .                                   | 71        |
| <b>6</b> | <b>Conclusion</b>  | <b>77</b> |

|          |                                   |           |
|----------|-----------------------------------|-----------|
| <b>7</b> | <b>Appendix</b>                   | <b>79</b> |
| .1       | OPA Simulation . . . . .          | 79        |
| .2       | Genetic Algorithm . . . . .       | 94        |
| .3       | Phase Modulation Scheme . . . . . | 105       |

# List of Figures

|     |  |    |
|-----|--|----|
| 1.1 | Basic mechanism of lidar [1] . . . . .   | 4  |
| 1.2 | Typical components of a lidar system . . . . .   | 5  |
| 1.3 | Responsivity to wavelength for common semiconductor material . . .   | 7  |
| 1.4 | MEMs-based OPA implementations (a) Galvometer-based scanner [2]<br>(b) High-contrast grating mirror [3] [(c) Electrostatic comb-driven<br>mirror [4] . . . . .             | 8  |
| 1.5 | Existing OPA demonstrations (a) Large scale 4096-element OPA [5]<br>(b) end-fire blue-wavelength OPA [6] (c) high resolution aliasing-free<br>beam steering [7]) . . . . . | 13 |
| 2.1 | Wave interference scenerio [8] . . . . .   | 16 |
| 2.2 | The coordinate used for deriving far field equation [9] . . . . .  | 17 |

|      |  |    |
|------|--|----|
| 3.1  | Simulation setup (a) Far field approximation [10] (b) Simulation region  | 21 |
| 3.2  | Polar plot constant spaced OPA at various spacing and number of emitters . . . . .   | 23 |
| 3.3  | Simulation plot of an example OPA with $N = 16$ , $d = 4 \lambda$ . . . . .  | 24 |
| 3.4  | (a) Main lobe's FWHM vs spacing (b) Max steering angle vs spacing  | 25 |
| 3.5  | Far field simulation between theory and simulation at various phase shift (a) $\pi/2$ and $3\pi/2$ shift (theory) (b) $3\pi/2$ shift (simulation) . . .  | 26 |
| 3.6  | Simulated and approximated steering angle Shift over $2 \pi$ phase shift   | 26 |
| 3.7  | Gaussian beam visualization . . . . .  | 28 |
| 3.8  | Far field intensity for various constant spaced OPA, modeled by Gaussian beam . . . . .  | 29 |
| 3.9  | (a) Overlaid shifted Gaussian-modeled OPA (b) Simulated steered angle with phase difference (c) Max steering and FWHM with spacing .   | 30 |
| 3.10 | Far field simulation of periodic spacing distribution (a) $\lambda$ - $2\lambda$ pattern (b) $\lambda$ - $2\lambda$ - $\lambda$ pattern (c) $\lambda$ - $2\lambda$ - $3\lambda$ - $4\lambda$ pattern . . . . . | 32 |
| 3.11 | Far field simulation of 16-waveguide Golomb distribution . . . . .   | 33 |
| 3.12 | Genetic Algorithm : Fitness over generations . . . . .   | 37 |
| 3.13 | Far field of a best distribution at immediate generations . . . . .  | 37 |

|      |  |    |
|------|--|----|
| 3.14 | Coordinate setup for non-constant pitch OPA Phase Modulation Scheme  | 38 |
| 3.15 | Simulation result of Phase Scheme for ideal sources . . . . .  | 41 |
| 3.16 | Simulation result of Phase Scheme for Gaussian sources . . . . .   | 42 |
| 4.1  | (a) Refractive index of Si, SiO <sub>2</sub> , SiN from 0.25 $\mu m$ to 1.25 $\mu m$ (b)<br>Extinction Coefficient of Si, SiO <sub>2</sub> , SiN from 0.25 $\mu m$ to 1.25 $\mu m$ . . . | 47 |
| 4.2  | Effective index result of SiN slab - thickness sweep from 0 to 0.3 $\mu m$ .   | 48 |
| 4.3  | Effective index result of SiN strip - waveguide width sweep from 0 to<br>0.5 $\mu m$ . . . . .   | 49 |
| 4.4  | FDE simulation of fundamental mode of 300 nm x 150 nm SiN strip<br>waveguide (a) TE mode (b) TM mode . . . . .   | 50 |
| 4.5  | varFDTD effective index estimation of a 11-element OPA . . . . .   | 52 |
| 4.6  | FDTD OPA Simulation Region of length 20 $\mu m$ . . . . .  | 53 |
| 4.7  | Input source in terms of spectrum and time . . . . .   | 53 |
| 4.8  | FDTD simulation of constantly spaced OPA from 500 nm to 750 nm<br>50 nm interval (order from left to right, top to bottom) . . . . .   | 54 |
| 4.9  | (a) Intensity along central waveguide, the closest waveguide, the sec-<br>ond closest waveguide at 500 nm spacing (b) Coupling length vs spacing   | 55 |
| 4.10 | Simulation region for width pattern 4 . . . . .  | 57 |

|      |  |    |
|------|--|----|
| 4.11 | FDTD Simulation of widths patterns (in rows: top to bottom is pattern 1 to 4 consecutively) at 500 nm (left column) and 550 nm (right column) spacing . . . . .  | 58 |
| 5.1  | Chip Layout . . . . .  | 62 |
| 5.2  | Input-Split-Trombone stage of each design . . . . .  | 63 |
| 5.3  | Two different phase shifter designs (a) for design 1-3 (b) for design 4-8  | 64 |
| 5.4  | (a) OPA Design 1 - Constant Spacing - Width pattern 4 (b) OPA Design 2 - Constant Spacing $0.75 \mu m$ - Constant Width . . . . .  | 66 |
| 5.5  | (a) OPA Design 3 - Golomb N 16 (b) OPA Design 4 - Random Distribution . . . . .  | 67 |
| 5.6  | (a) OPA Design 5 - N 8 - Constant width - Constant spacing $0.9 \mu m$<br>(b) OPA Design 6 - N 4 - Constant width - Constant spacing $0.75 \mu m$<br>(c) OPA Design 7 - N 4 - Constant width - Constant spacing $0.6 \mu m$<br>(d) OPA Design 8 - N 8 - Constant width - Constant spacing $0.75 \mu m$ | 68 |
| 5.7  | Ring resonator configurations (a) all pass (b) add-drop . . . . .  | 69 |
| 5.8  | Experimental setup for ring resonators at blue wavelengths . . . . .   | 70 |
| 5.9  | Crystal calibration: Optimal temperature vs output wavelength . . . . .  | 71 |
| 5.10 | Top down view of a ring resonance at resonance at 990.114 nm . . . . .   | 72 |

|      |   |    |
|------|---|----|
| 5.11 | 100 nm gap (a) First resonance (b) Second resonance . . . . . | 73 |
| 5.12 | 150 nm gap (a) First resonance (b) Second resonance . . . . . | 73 |
| 5.13 | 200 nm gap (a) First resonance (b) Second resonance . . . . . | 74 |
| 5.14 | 250 nm gap (a) First resonance (b) Second resonance . . . . . | 74 |
| 5.15 | 300 nm gap (a) First resonance (b) Second resonance . . . . . | 75 |
| 5.16 | (a) Resonance wavelength vs gap (b) FSR vs gap . . . . .      | 76 |

# List of Tables

- 3.1 Selected Golomb Rulers . . . . . 33
- 4.1 Waveguide width pattern . . . . . 56
- 5.1 Six OPA designs on the chip layout . . . . . 61

# Chapter 1

## Introduction

Optical phased arrays (OPAs) have emerged as promising technology for controlling light beams in various applications such as remote sensing, imaging, and communication systems. OPAs are capable of steering the direction of laser beams without mechanical movement, thus providing an efficient means to control light. Although OPAs have been known for some time, their development has primarily focused on the near-infrared range due to the challenges associated with fabrication and coupling.

This dissertation presents several OPA designs operating at blue wavelengths (450 nm) that enhance the steering range and power delivery. The focus on blue or visible-wavelength OPAs is of particular importance, especially in applications such as augmented/virtual reality, displays, oceanography, atmospheric sensing, and quantum information systems based on atoms/ions.

Chapter 1 provides an in-depth background on technology and the driving forces

behind it. The advantages of using OPAs are explored, including their fast response times, scalability, and flexibility. Furthermore, a brief overview of current state-of-the-art OPAs in academia including those used in the visible spectrum is presented.

In Chapter 2, we delve deeper into the phased array mechanism, deriving characteristic parameters approximation of the system such as maximum steering angle, and Full-width-half-max (FWHM).

Chapters 3 and 4 are devoted to various OPA designs' simulation and modeling work. We describe the various techniques utilized to simulate the performance of the OPA at blue wavelengths in MATLAB and Lumerical. Simulation results of numerous array distributions like constant spacing, periodic spacing, Golomb spacing, random spacing, and periodic width distribution, are comprehensively studied in these chapters.

Chapter 5 describes the chip layout with 8 different OPA designs and the preliminary experimental results of blue wavelength on ring resonators. Open-source tools like Nazca and KLayout are used extensively to develop the layout. With the potential of the Si Nitride platform for blue wavelength, anticipated challenges and future testing plans are discussed.

## 1.1 Light Detection and Ranging

Lidar (Light Detection and Ranging) is a method for detecting objects and determining ranges by using laser. The idea of lidar is similar to that of radar, which utilizes radio or microwaves. However, because of the much shorter wavelength of light, the realization of lidar came later after the invention of the laser. Nowadays, lidar is commonly used to create high-resolution maps and navigation for autonomous cars.

### 1.1.1 Mechanism

Similar to radar, the fundamental principle of lidar is to send the light beam from a laser and measure the reflected light from the environment. The emitted light is either amplitude/frequency modulated or comes in a short pulse. The pulse will then pass through a system of optical, mechanical, and electrical components to direct it to the target. Finally, the lidar detector records and extracts the time and phase difference of the light received. From that information, spatial information of the environment is calculated.

Spatial information from lidar systems is available in three types: one-dimensional, two-dimensional, and three-dimensional. To enable a large and precise scanning area, 2D and 3D lidar needs some kind of light deflecting or steering system, whereas 1D lidar does not need to and is commonly used as a laser rangefinder.

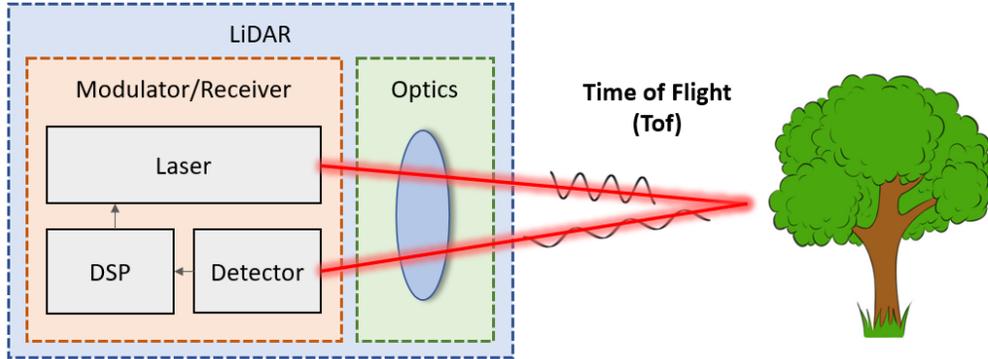


Figure 1.1: Basic mechanism of lidar [1]

### 1.1.2 Applications

Lidar has proven to be a valuable tool in many fields due to its accuracy and speed. For example, lidar has been demonstrated to achieve an accuracy of 10 cm or better. This level of accuracy is much higher than traditional sensing techniques, such as photogrammetry. A study by Velodyne showed that lidar systems could detect objects up to 120 meters away with an accuracy of a few centimeters. In comparison, radar systems have a detection range of up to 80-100 meters with several meters in accuracy [11, 12]. Lidar has already been used for a wide range of fields including agriculture surveying, scanning, autonomous driving, robotics for factory automation.

As lidar is becoming more compact in size, lidar modules are becoming popular in consumer mobile applications such as face detection, 3D scanning to health monitoring, and VR/AR.

### 1.1.3 Components

Depending on application and specification, engineers have devised many lidar systems based on four major components of the lidar: emitter, beam steering, measurement process, and receiver.

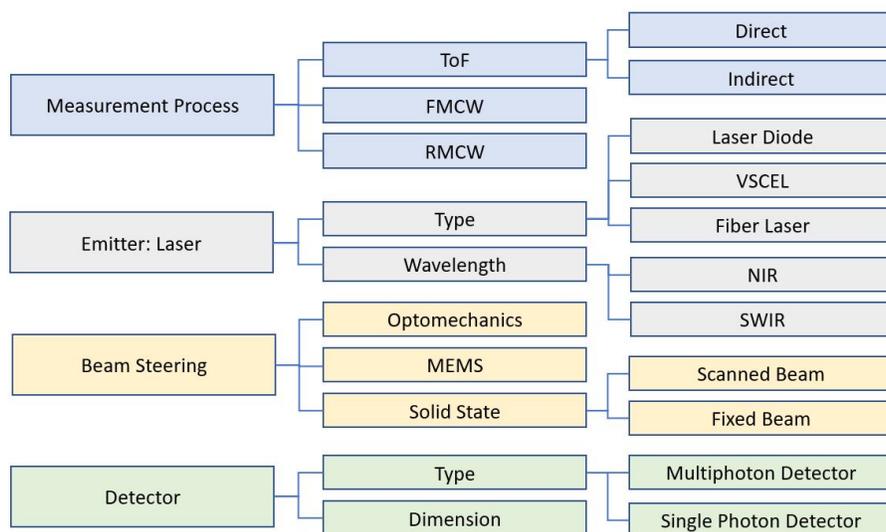


Figure 1.2: Typical components of a lidar system

There are many metrics for choosing a laser for lidar systems, such as power delivery, range, ultra-short pulses, and wavelength range. While module solutions such as fiber laser are suitable for large-scale high-altitude surveying and scanning, integrated solutions like the laser diode and the vertical-cavity surface-emitting laser (VCSEL) are easy to scale and also achieve good performance.

Another significant factor of the laser source is the operating wavelength, which impacts aspects of resolution, power delivery, and atmospheric absorption. The most

common wavelength range for commercial lidar is at the near-infrared (NIR) band, which ranges roughly from 850nm to 950nm. It is because the wavelength in NIR is much more weather-proof and has a higher range. A study by Wojtanowski et al. [13] on rangefinder performance deterioration measured that signal transmission at 1550 nm is about 4-5 times worse than at 905 nm. In addition, we can create efficient silicon-based avalanche photodiodes to build low-cost and high-sensitivity detectors at the NIR range. On the other hand, 1550 nm is popular in communication systems due to low fiber loss, as good as 0.142 dB/km, and higher eye-safe emission power [14].

Like light sources, photodetectors are chosen for specific requirements and wavelength ranges. The detector material defines its sensitivity to the wavelength of interest, depicted in Fig 1.3. Si-based detectors are used for wavebands less than 1  $\mu m$ , while InGaAs are implemented for longer wavelengths. Other materials, such as InGaAs/InP heterostructures, have also been demonstrated, but this is offset by the high cost [15].

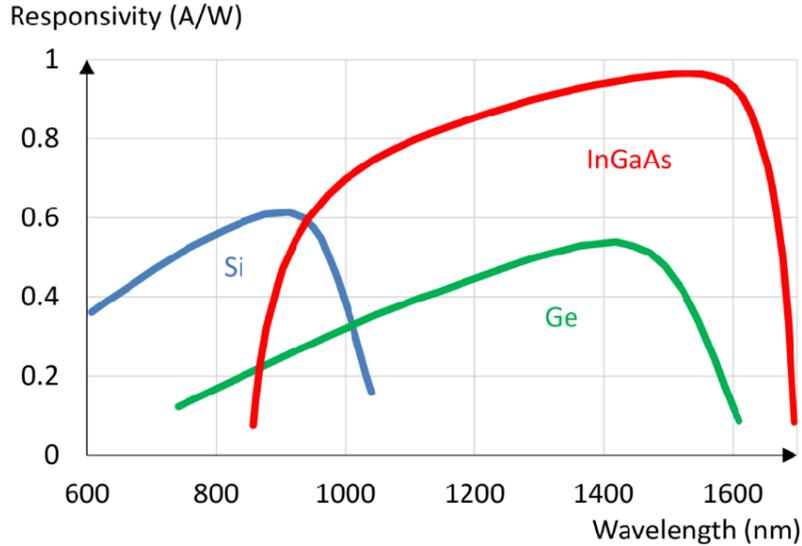


Figure 1.3: Responsivity to wavelength for common semiconductor material

The most popular type of detector used in commercial automotive lidar is the single-photon avalanche diode (SPAD). They work by amplifying the signal produced by the photodiodes through a large number of ionizing events. It has the ability to detect a single photon, thus suitable in challenging environments where the return signal is weak or the background light is strong.

## 1.2 Beam Steering mechanism

### 1.2.1 Non-solid-state methods

For 2D and 3D lidar, beam deflection or beam steering plays a crucial role in lidar performance. The steering mechanism can be broadly characterized into three

groups: Optoelectromechanical, MEMS, and solid-state scanning.

In the early days, beam steering was performed through a system of optical, electronics, and mechanical parts. The simplest method is to mount the laser to a rotating station. However, that system suffers from limited steering angles and rotation axis. Another scheme is to fix the laser and allow the optics system to move. The emitted beam from a static laser will hit a movable mirror and lens, which can be controlled by electric motors. Many demonstrations of galvanometric scanning could be used as off-the-shelf products or can achieve a scan error of less than 2% for micrometer precision.

Microelectromechanical system (MEMS) mirrors offer a more compact and integrable solution to optoelectromechanical scanning. It provides the same functionality but drastically reduces the size to less than  $100\ \mu\text{m}$ . MEMS mirrors have been demonstrated to have a scanning rate of 0.42 MHz [16], sub-mrad resolution [17], and large FoV [4]. For extensive applications of the MEMS mirror, it can be referred to [12, Holmstrom et al.].

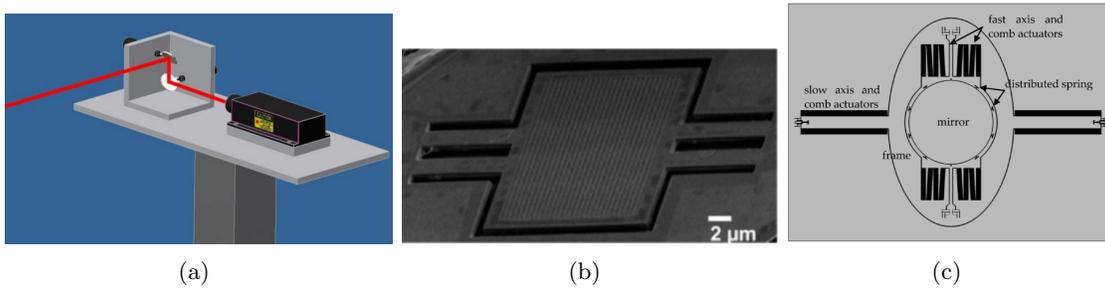


Figure 1.4: MEMS-based OPA implementations (a) Galvanometer-based scanner [2] (b) High-contrast grating mirror [3] (c) Electrostatic comb-driven mirror [4]

### 1.2.2 Limitations

All of the steering solutions we have described above are based on an optomechanical part that tends to be large and bulky. Although micromirrors can solve this problem, reliability concerns are present in the fabrication and integration of the system. Additionally, because it is a mechanical moving part, it limits steering speed and reduces durability, which can be a major failure point in the system. To overcome these challenges, a new wave of innovation is to develop all-solid-state lidar, and the optical phased array is a crucial part of it.

## 1.3 Photonic Integrated Circuit

Since the invention of transistors and silicon fabrication processes in the mid-20th century, integrated circuits have transformed the entire field of electronics. The idea is to fabricate every electronic component onto a single monolithic chip. With the advent of semiconductor fabrication techniques, such as lithography, etching, and implantation, we can scale down devices and increase the performance of microchips given an area.

The photonic integrated circuit has a motivation similar to that of the traditional integrated circuit. Historically, photonic processes have been highly specialized for particular applications, utilizing different materials. With individual devices packaged separately and connected by fibers, it is common for communication systems to be made of many different material systems. RF CMOS for high-bandwidth elec-

tronics (e.g., serializers and deserializers), CMOS digital parts, lithium niobate for modulators, indium phosphide for lasers, germanium for photodetectors, and MEM-based switches. Each of these devices is made in a fundamentally incompatible process that significantly drives up the fabrication cost. Another high final device cost and yield loss emerges from the photonic packaging processes, which generally require 5- and 6-axis alignments with sub-micrometer accuracy [18].

Over the past decade, silicon has been studied and proven to be an excellent material for building photonic devices. At the same time, considerable effort has been made to reuse the mature and advanced CMOS process for photonics. It is in its nature that the two processes are not directly transferable, where photonics requires processing of the range of 90 nm up to microns compared to high precision microelectronics chips fabrication of 16 nm or less. However, progress is being made to rearrange and reuse modular process steps with distinct process flows being developed to build silicon photonics.

There are a vast number of applications that can use complex silicon photonic systems, the most common being digital data communication. This is driven by the fundamental differences in physics between electrons being Fermion particles, and photons, which are Bosons. Electrons are great for computation because fermions cannot be in the same place simultaneously, and they interact strongly with each other. As a result, it is possible to build massively nonlinear switching devices in electronics that are used to make logic gates. Photons, on the other hand, do not tend to interact and can transmit long distances without significant losses [19]. Therefore,

it is possible to transmit terabits of data per second through a single optical fiber using techniques such as wavelength division multiplexing [20].

Beyond data communications, new applications for photonics are constantly being explored, including condensed matter physics [21], biosensing [22], optical gyroscopes, microwave signal processing [23], and even electronics-photonics microprocessor [24]. This paper focuses on the application of photonics to the development of lidar.

### 1.3.1 Existing OPA

There are two ways to emit OPA light outside the chip: surface-emitting and end-fire emitting. Most OPA demonstrations use grating couplers or antennas to scatter light vertically out of the chip's surface. This method can control the light beam in 2-dimension and proves to be a versatile solution for lidar. Considerable effort has been made to create an efficient large-scale broadside OPA with a high steering angle. One of the largest OPAs developed contains 4096 elements in a 64x64 array with a footprint of  $576 \mu m \times 576 \mu m$  using Si at 1550 nm [5]. A large aperture of 4mm x 4mm of 1024 antennas ( $4 \mu m$  pitch) is also demonstrated at 1550 nm with a maximum steering angle of 23 degrees [25, 26]. Several variations of 2-D scanning OPA are demonstrated by Doylend [27], Toshihiko [28], and Hutchison [7].

An alternative topology for OPA uses an end-fire array that will guide and emit light at the edge of the photonic chip. This method is easier to design and fabricate, while not causing physical restrictions from both dimensions. Since the emitted beam will be in the same plane as the chip, it will experience a lower loss than grating

couplers for emission despite some reflection loss at the end of the waveguide.

This thesis is focused on visible wavelength, in particular, blue-wavelength OPA. This requires a waveguide material that is transparent at blue wavelengths. We selected SiN due to its CMOS compatibility. However, the biggest challenge of realizing a blue wavelength OPA using SiN is that the waveguides cannot be spaced at a half-wavelength (225nm) because the waveguides themselves are at least that wide, and there is also considerably coupling between them. To address this challenge [6, Min Chul et. al] demonstrated a SiN-based OPA at blue wavelengths that achieved 50 degrees of steering by spacing the waveguides aperiodically with an average spacing of 31.5 lambdas. This thesis explores a variety of approaches for realizing blue-wavelength OPA designs, in particular, the use of alternative aperiodic and waveguide designs.

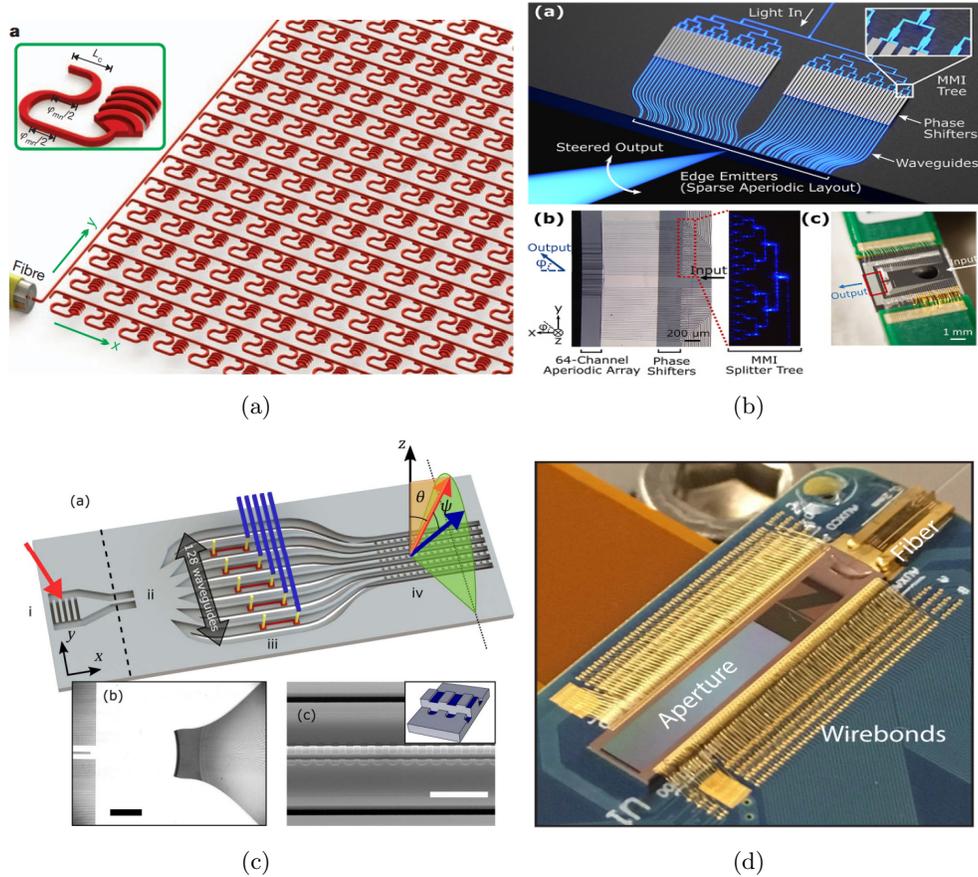


Figure 1.5: Existing OPA demonstrations (a) Large scale 4096-element OPA [5] (b) end-fire blue-wavelength OPA [6] (c) high resolution aliasing-free beam steering [7] (d) long-range for data communication OPA [26]

# Chapter 2

## Theory

### 2.1 Wave interference

The operation of a phased array is based on the concept of wave interference. Light is an electromagnetic wave and has the properties of amplitude, frequency, and phase. The frequency of a wave visually manifests itself as the wavelength in a medium, which is the distance between two consecutive crests or troughs. The dependency between wavelength and frequency for an electromagnetic wave can be described as follows.

$$\lambda = \frac{c_0}{nf}, \quad (2.1)$$

where  $\lambda$  and  $f$  are wavelength and frequency of the wave,  $c_0$  is the speed of light in vacuum, and  $n$  is the refractive index of the medium.

A phase is a mathematical quantity used to describe the state of a wave that depends on both space and time. Wave interference is a fundamental concept that describes the interaction of waves. In order for light-wave interference to occur, there are 3 conditions that need to be satisfied. First, all the waves must be at the same wavelength otherwise the interference will be canceled out over space and time as the phase difference keeps changing. A small difference in wavelength will lead to a beating effect similar to that of a sound wave; however, this is undesirable for beam steering applications. Second, all waves must be coherent, in which the phase difference between each wave does not change. Ordinary light from the Sun or light bulb consists of a mixture of different wavelength rays whose phases and direction are random. For precise control over the light source, the laser must be used because it can achieve a narrow linewidth, highly directional, and high-intensity beam. For this application, the usage of only one laser is desired to mitigate the possible variation in the beam.

The interference pattern can be constructive, where the waves are in phase and create a larger pattern, or destructive, where the waves cancel each other out and create a smaller or zero amplitude. This superposition of waves over space forms interference fringes, which are regions of constructive and destructive interference.

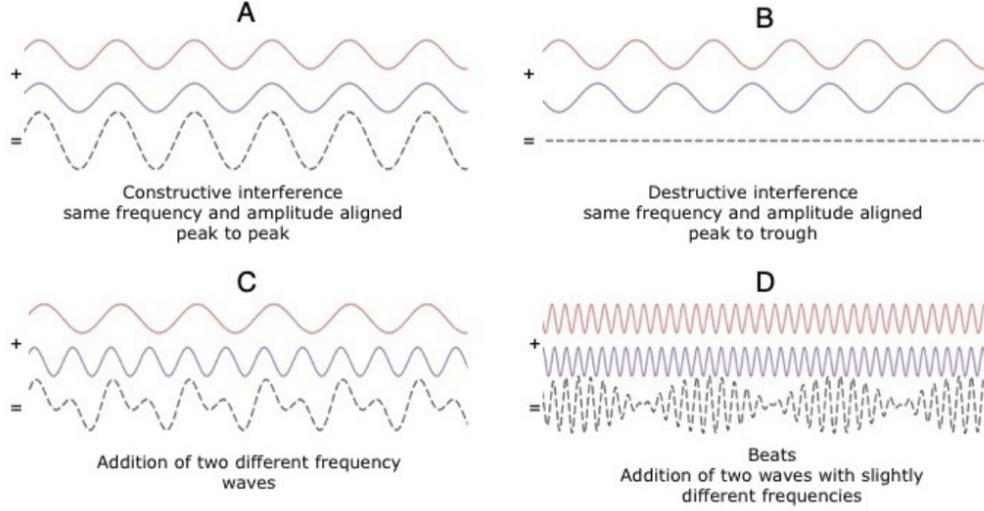


Figure 2.1: Wave interference scenerio [8]

## 2.2 Phased Array Theory

Phased Array is a phase-controlled array of antennas that creates a beam of electromagnetic waves that can be directed electronically to point in different directions without moving the antennas. The beam is the wave interference pattern from the antenna distribution and its phase. Assuming ideal point sources, the electric field profile from a phased array is the superposition of the electric field of the sources.

$$E(x, y, z) = \sum_{n=0}^{N-1} A_n \exp\left(-j\left(\frac{2\pi\mathbf{r}_n}{\lambda} + \phi_n\right)\right), \quad (2.2)$$

where  $N$  is the total number of antennas while  $A_n$ ,  $\mathbf{r}_n$ , and  $\phi_n$  are the amplitude, distance to the point of interest, and initial phase of the  $n^{\text{th}}$  antenna in the array.

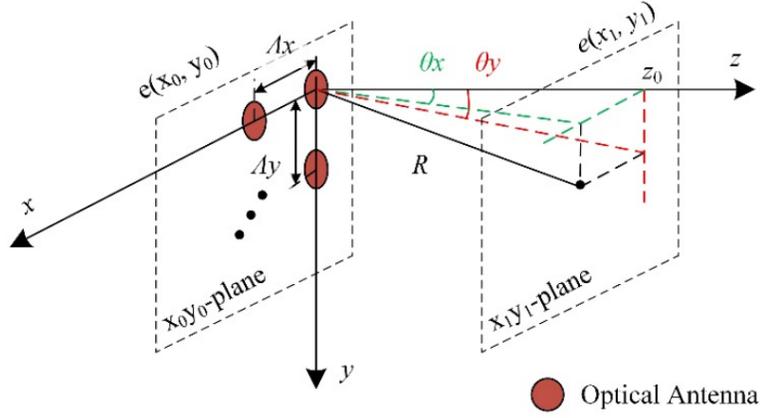


Figure 2.2: The coordinate used for deriving far field equation [9]

For an  $M \times N$  OPA arranged in a rectangle with constant lateral and longitudinal spacing between two consecutive emitters of  $\Lambda_x$ ,  $\Lambda_y$ , we choose the origin in the upper right emitter and the coordinate as shown in Fig. (2.2). Along the  $x$  and  $y$  axes, each emitter will be added with a constant phase of  $\Delta\phi_x$  and  $\Delta\phi_y$  to the previous one, respectively. Additionally, assuming that all emitters are identical and ideal, the far-field is the sum contribution of each emitter in the array.

$$E(x, y, z) = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} A_n \exp(-j(k\mathbf{r}_{mn} + m\Delta\phi_x + n\Delta\phi_y)), \quad (2.3)$$

where the distance from the horizontal element  $m^{th}$  and the vertical element  $n^{th}$  to

the point of interest is:

$$\begin{aligned}
r_{mn}^2 &= (x_0 - m\Lambda_x)^2 + (y_0 - n\Lambda_y)^2 + z_0^2, \\
r_{mn}^2 &= x_0^2 + y_0^2 + z_0^2 - 2(x_0m\Lambda_x + y_0n\Lambda_y), \\
r_{mn}^2 &= r_{00}^2 \left( 1 - \frac{2}{r_{00}^2} (x_0m\Lambda_x + y_0n\Lambda_y) \right) \\
r_{mn} &= r_{00} \sqrt{1 - \frac{2}{r_{00}^2} (x_0m\Lambda_x + y_0n\Lambda_y)} \\
r_{mn} &= r_{00} \left( 1 - \frac{1}{r_{00}^2} (x_0m\Lambda_x + y_0n\Lambda_y) \right)
\end{aligned} \tag{2.4}$$

In the far field where  $z_0 \gg M\Lambda_x$  and  $z_0 \gg N\Lambda_y$ , (2.4) and (2.3) can be approximated:

$$r_{mn} \approx r_{00} - m\Lambda_x \sin \theta_x - n\Lambda_y \sin \theta_y \tag{2.5}$$

$$E(\phi_x, \phi_y) = A_0 e^{-jR_{00}} \sum_{m=0}^{M-1} \exp(jm(k\Lambda_x \sin \theta_x - \Delta\phi_x)) \sum_{n=0}^{N-1} \exp(jn(k\Lambda_y \sin \theta_y - \Delta\phi_y)) \tag{2.6}$$

Using the exponential sum formula, the normalized array factor  $T(\phi_x, \phi_y)$  is:

$$\begin{aligned}
I(\phi_x, \phi_y) &= \frac{|E(\phi_x, \phi_y)|}{M * N} \\
&= \frac{\sin\left(\frac{M(k\Lambda_x \sin \theta_x - \Delta\phi_x)}{2}\right)}{M \sin\left(\frac{k\Lambda_x \sin \theta_x - \Delta\phi_x}{2}\right)} \times \frac{\sin\left(\frac{N(k\Lambda_y \sin \theta_y - \Delta\phi_y)}{2}\right)}{N \sin\left(\frac{k\Lambda_y \sin \theta_y - \Delta\phi_y}{2}\right)}
\end{aligned} \tag{2.7}$$

Looking at the  $x$  dimension, the factor of the array shows a maximum when both the numerator and the denominator become zero.

$$\begin{aligned} \frac{1}{2}(k_0\Lambda\theta_0 - \Delta\phi_x) &= q\pi \\ \sin\theta_x &= \left(q + \frac{\Delta\phi_x}{2\pi}\right) \frac{\lambda}{\Lambda_x} \\ \sin\theta_{x,q+1} - \sin\theta_{x,q} &= \frac{\lambda}{\Lambda_x} \end{aligned} \tag{2.8}$$

When  $\Lambda_x < \lambda/2$ ,  $\sin\theta_{x,q+1} - \sin\theta_{x,q} > 2$  which yields no solution for any  $q$ . This constraint of spacing  $\lambda/2$  is well known in phased arrays, as it is the theoretical limit to ensure only one lobe or no aliasing [29–32].

Two important parameters in characterizing the emitted beam are the full width half maximum (FWHM) and the free spectrum range (FSR). While FWHM is the width that contains at least half the peak power of the beam, FSR is the angle between two grating lobes.

When the angles  $\theta_x$  are relatively small, we can approximate FWHM and FSR as follows [9, 33]:

$$\Delta\theta_{FWHM,x} \approx 0.886 \frac{\lambda}{N_x\Lambda_x \cos\theta_x}, \tag{2.9}$$

$$\Delta\theta_{FSR,x} \approx \frac{\lambda}{\Lambda_x} \tag{2.10}$$

## Chapter 3

# Simulation

### 3.1 Simulation Setup

To accurately simulate an arbitrary distribution far field using a 2D simulation setup, we start by defining a simulation domain that is large enough to encompass both the far-field and near-field pattern. The boundary between the near- and far-field regions is vaguely defined. A good rule of thumb is to use the Fraunhofer distance  $d_F = 2d^2/\lambda$  to estimate the far-field region, where  $d$  is the aperture of the array.

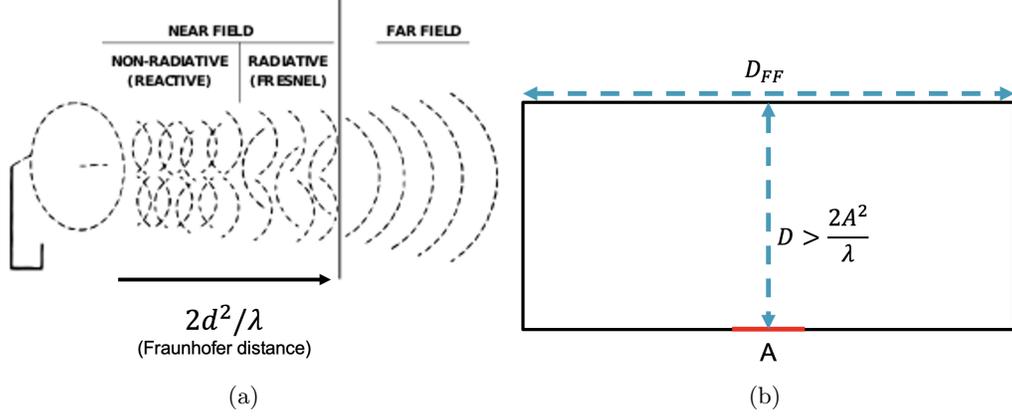


Figure 3.1: Simulation setup (a) Far field approximation [10] (b) Simulation region

Once the simulation domain has been defined, we create a 2D grid that covers the entire domain. The grid should have a spacing much smaller than the wavelength of the EM wave and be compatible with the angular resolution chosen to 0.05 degrees. Then, the OPA array is placed at the center of the bottom side of the simulation region. At the far field, the region length (denoted as  $D_{FF}$  in Fig 3.1), must be at least 100 times the width  $D$  to obtain a maximum angle of 88.9 degrees. For 2D simulation where the focus is on how the beam is formed from the array, region length can be kept as short as two times width to reduce simulation time.

After the grid has been created, the electric field at each point is the sum of the electric field contributions from each emitter, which involves calculating the wave propagation at each point.

Once we have calculated the electric field at each point on the grid, we can then calculate the intensity of the field at each point. This is done by squaring the

magnitude of the electric field. The resulting intensity distribution represents the far-field radiation pattern of the emitters.

Visualizing the far-field radiation pattern involves plotting the intensity distribution on a 2D graph, where the x and y axes represent the position of each point on the grid, and the color or shading represents the intensity of the field at each point.

## 3.2 Constant Distribution

### 3.2.1 Ideal Source

Using theory approximation in the far field, Figure 3.2 shows the far field intensity of 8, 16, 32 emitter array (in columns from left to right, respectively) with increasing spacing  $\lambda/2$ ,  $\lambda$ , and  $2\lambda$  (in rows top to bottom). At first glance, the results quantitatively confirm a number of parameter relationships in OPA design.

- only the spacing  $\lambda/2$  does not have side lobes that will yield a maximum beam steering of 180 degrees.
- larger spacing introduces more grating lobes and decreases the angle difference between lobes, lowering the maximum steering angle.
- increasing the number of emitters decreases the beam width without changing the position of the grating lobes.

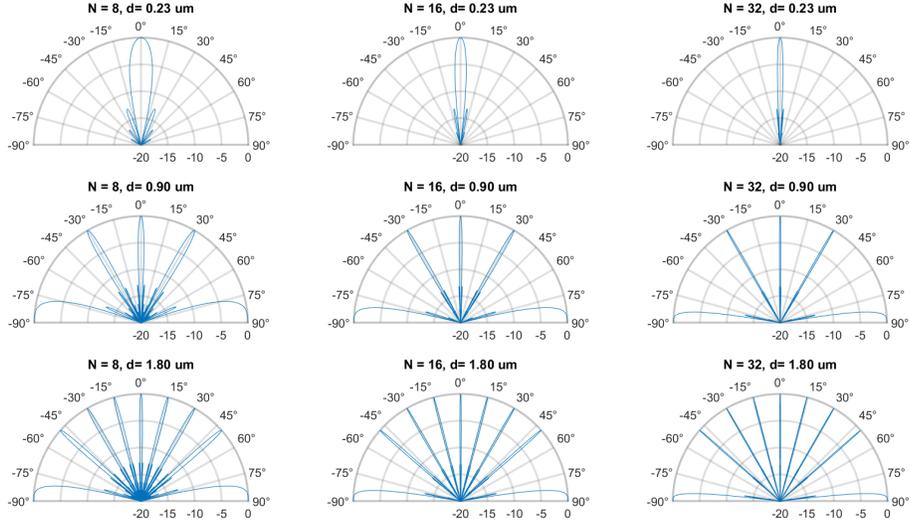


Figure 3.2: Polar plot constant spaced OPA at various spacing and number of emitters

Using the simulation setup described above on the same set of spacing and emitters, an intensity of the entire field is achieved, yielding similar far-field results compared to the theory approximation. In addition, near-field radiation is also captured. For example, a simulation of 16-waveguide OPA spaced  $4\lambda$  is presented in Fig. 3.3, which includes a plot of central lobes in  $\pm 1$ -degree range, Cartesian plot and polar plot of the far field, and an image of the entire simulation region.

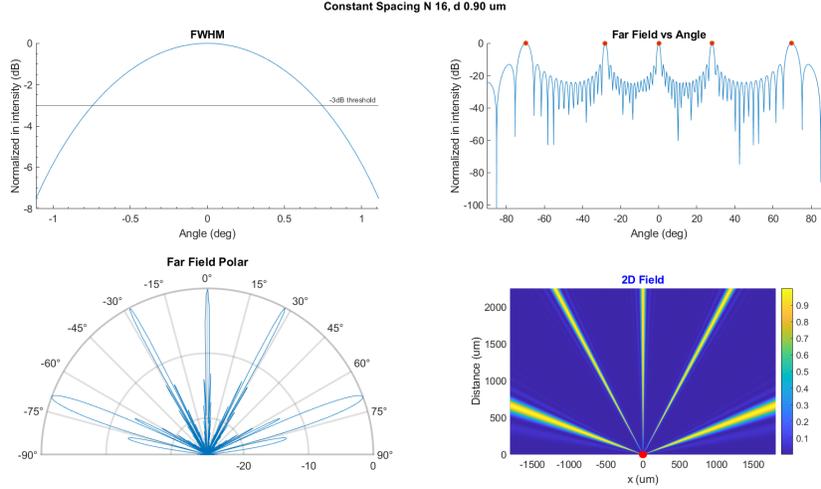


Figure 3.3: Simulation plot of an example OPA with  $N = 16$ ,  $d = 4 \lambda$

FWHM of each configuration is evaluated only for the main lobe because it is typically the region of interest for communication or sensing applications. The FWHM is calculated as the distance between the intersections of the normalized intensity curve and the -3dB line. On the other hand, the maximum steering angle is the angle between the central lobe and the first-order lobe at which the OPA can be steered without receiving a false signal.

Fig 3.4 compares central lobes' FWHM and maximum steering angle between approximation (3) and simulation over various spacings and the number of waveguides.

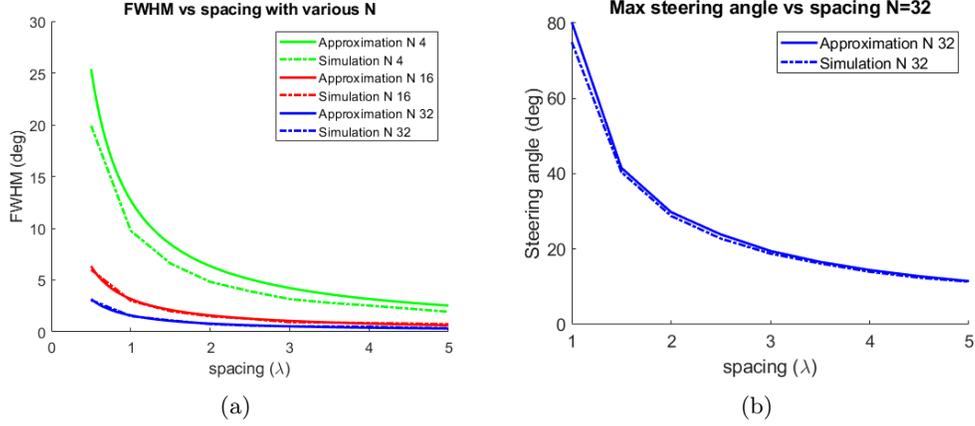


Figure 3.4: (a) Main lobe's FWHM vs spacing (b) Max steering angle vs spacing

### 3.2.2 Linear phase difference

One of the main advantages of solid-state OPA is that each waveguide phase can be controlled individually, which can be described by the phase modulation distribution, which depends on the array spacing distribution. For constant spacing distribution, linear phase distribution can be used where each waveguide will have a constant phase difference to the adjacent waveguide in order. Call  $\phi$  the phase difference between 2 adjacent waveguides. Fig. 3.5 presents the far field intensity of 32 elements of OPA shifted by  $\phi = 3\pi/4$ , and Fig. 3.6 is the result when  $\phi$  is sweeping over  $2\pi$ .

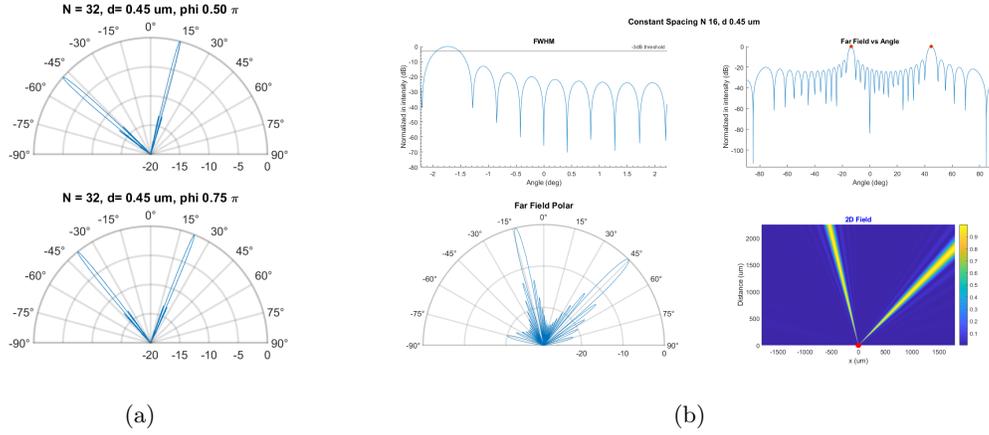


Figure 3.5: Far field simulation between theory and simulation at various phase shift (a)  $\pi/2$  and  $3\pi/2$  shift (theory) (b)  $3\pi/2$  shift (simulation)

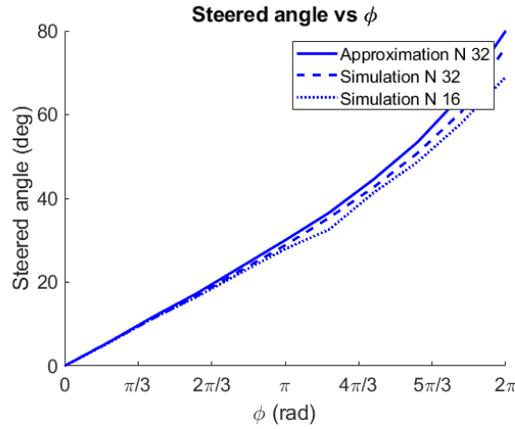


Figure 3.6: Simulated and approximated steering angle Shift over  $2\pi$  phase shift

From the results, both theory and simulation match well for constant spacing. In addition to constant spacing, only simulation can be used to visualize and calculate the far-field beam.

### 3.2.3 Gaussian Distribution

The actual beam emitted by the waveguides can be more accurately described by the Gaussian beam. Assuming polarization in the x direction and propagation in the +z direction, the electric field in the phasor notation is given by

$$\mathbf{E}(r, z) = E_0 \frac{w_0}{w(z)} \exp\left(\frac{-r^2}{w(z)^2}\right) \exp\left(-i\left(kz + k\frac{r^2}{2R(z)} - \phi(z)\right)\right) \hat{\mathbf{x}}, \quad (3.1)$$

where

- r is the radial distance from the center axis of the beam,
- z is the axial distance from the beam's focus (or "waist"),
- $E_0 = E(0, 0)$ , the electric field amplitude (and phase) at the origin (r = 0, z = 0),
- $w(z) = w_0 \sqrt{1 + \left(\frac{z}{z_R}\right)^2}$
- $w_0 = w(0)$  is the waist radius,
- $R(z) = z \left[1 + \left(\frac{z_R}{z}\right)^2\right]$  is the radius of curvature of the beam's wavefronts at z,
- $\phi(z) = \arctan\left(\frac{z}{z_R}\right)$  is the Guoy phase (attributable to the phase velocity)
- $z_R = \frac{\pi w_0^2 n}{\lambda}$  is the Rayleigh range

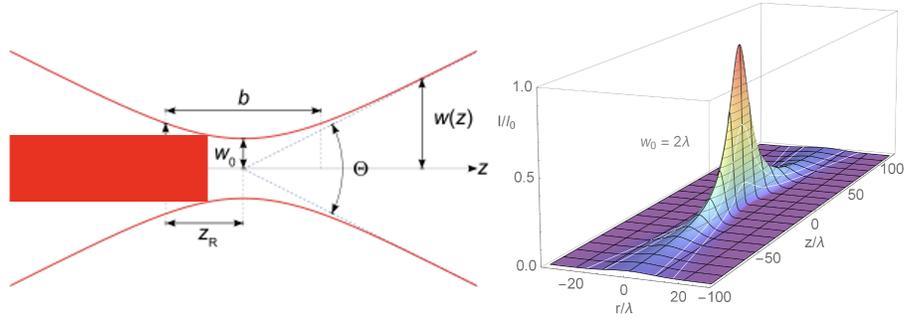


Figure 3.7: Gaussian beam visualization

The Gaussian emitted from the waveguide is described by the waist radius  $w_0 = 0.15\mu\text{m}$ . Fig 3.8 shows the far field intensity of 16 waveguide OPA at different constant spacings, modeled by the Gaussian beam.

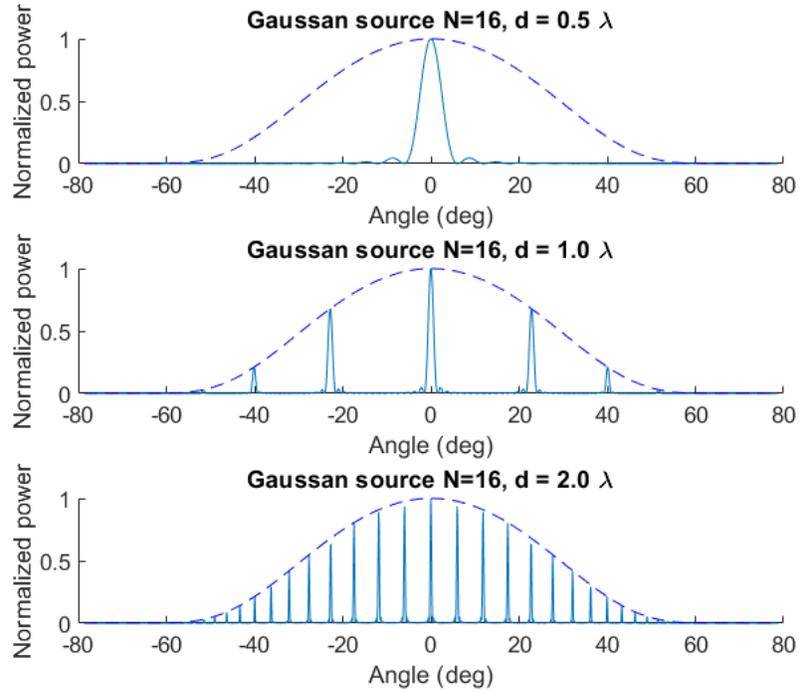
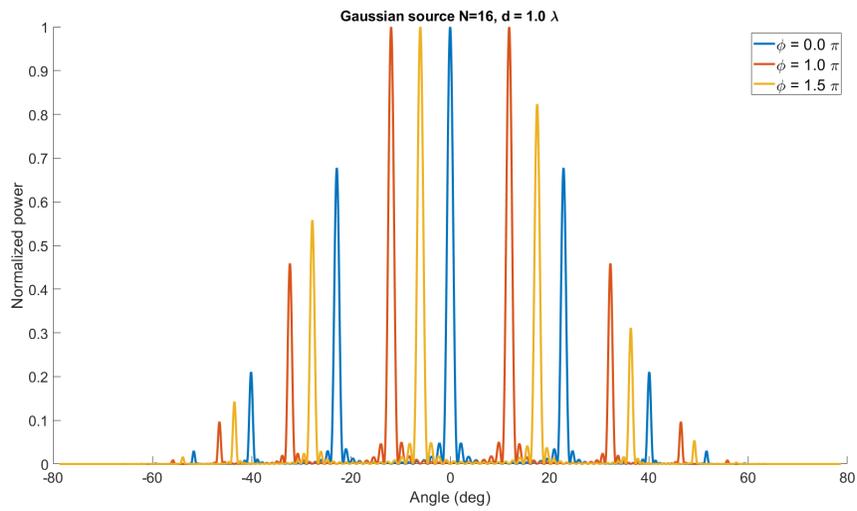


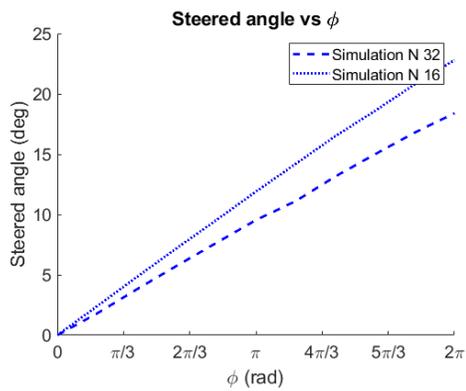
Figure 3.8: Far field intensity for various constant spaced OPA, modeled by Gaussian beam

The main lobe and grating lobes are enclosed by a Gaussian envelope, which is the Gaussian beam of an array with one emitter.

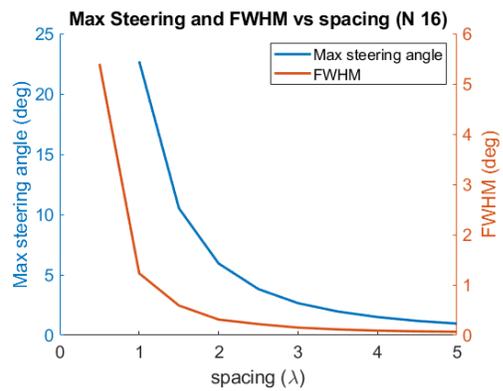
Similarly to the ideal source case, the beam can be steered by a linear phase distribution. FWHM and maximum steering angle can be extracted by varying the emitters' spacing.



(a)



(b)



(c)

Figure 3.9: (a) Overlaid shifted Gaussian-modeled OPA (b) Simulated steered angle with phase difference (c) Max steering and FWHM with spacing

### 3.3 Non-constant Distribution

Although constant distribution has been studied extensively and has been effective in implementing in a myriad of RF applications, it inherits a difficult problem for photonics: shorter wavelength means smaller spacing between waveguides to achieve no aliasing ( $\lambda/2$ ) to the point where coupling becomes a detrimental issue. Therefore, nonconstant spacing distribution is one of the solutions to alleviate the waveguide coupling problem while also achieving the required maximum steering angle.

#### 3.3.1 Periodic Distribution

Periodic distribution is the waveguide spacing pattern that repeats itself over the entire OPA. The idea is that the grating lobes will be suppressed when two or more spacings, not sharing the same angles of the grating lobes, are mixed. For example, when we mix  $\lambda$  and  $2\lambda$  spacings, the grating lobes created by the constant  $\lambda$  and  $2\lambda$  are suppressed. The more  $\lambda$  spacing waveguides in the OPA, the more similar the far field will look to the  $\lambda$  constant spacing OPA.

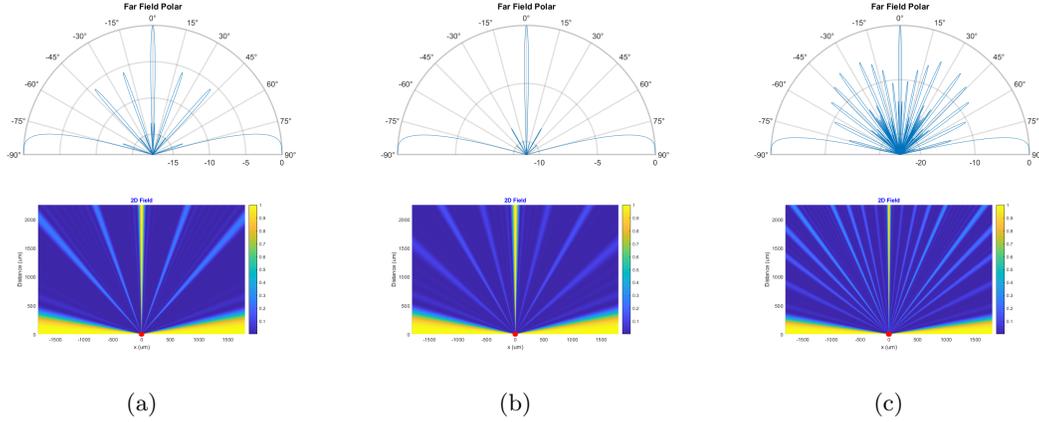


Figure 3.10: Far field simulation of periodic spacing distribution (a)  $\lambda$ - $2\lambda$  pattern (b)  $\lambda$ - $2\lambda$ - $\lambda$  pattern (c)  $\lambda$ - $2\lambda$ - $3\lambda$ - $4\lambda$  pattern

### 3.3.2 Golomb Distribution

In mathematics, a Golomb ruler is a set of marks at integer positions along a ruler such that no two pairs of marks are the same distance apart. The number of marks on the ruler is its order, and the largest distance between two of its marks is its length. [34] [35]

There is no requirement that a Golomb ruler be able to measure all distances up to its length, but if it does, it is called a perfect Golomb ruler. It has been proved that there is no perfect Golomb ruler for five or more marks [4]. A Golomb ruler is optimal if there is no shorter Golomb ruler of the same order. At the current time of writing, an optimal Golomb ruler up to order 28 has been found.

Golomb rulers are used in the design of phased arrays of radio antennas. In radio astronomy, one-dimensional synthesis arrays can have the antennas in a Golomb

ruler configuration in order to obtain minimum redundancy of the Fourier component sampling.

For example, a 16-element OPA will utilize the 16th-order Golomb ruler, then will be scaled to a reference spacing of  $\lambda/2$

Table 3.1: Selected Golomb Rulers

| Order | Length | Ruler   |
|-------|--------|---|
| 4     | 6      | 0 1 4 6   |
| 8     | 34     | 0 1 4 9 15 22 32 34                                 |
| 12    | 85     | 0 2 6 24 29 40 43 55 68 75 76 85                    |
| 16    | 177    | 0 1 4 11 26 32 56 68 76 115 117 134 150 163 168 177 |

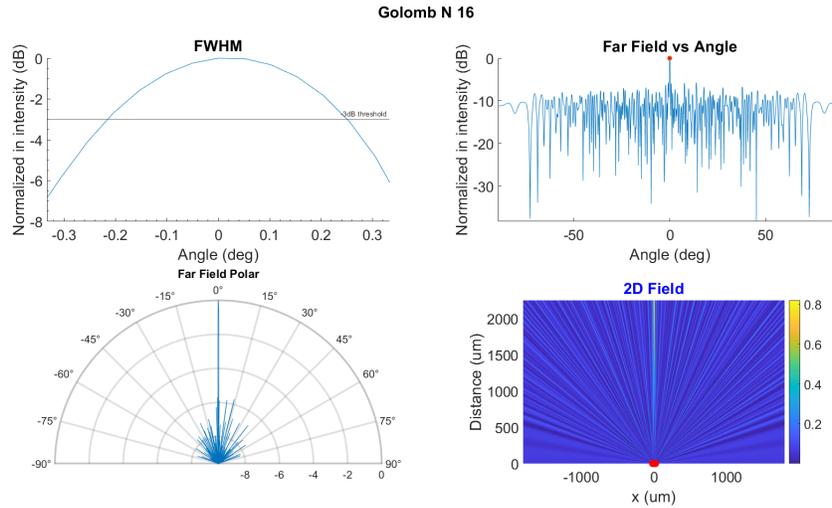


Figure 3.11: Far field simulation of 16-waveguide Golomb distribution

The result shows that all the grating lobes are suppressed by the non-repeating distance distribution of the waveguides. The far-field pattern from the Golomb spac-

ing is achieved by the combination of the wide aperture OPA which leads to small FWHM, and sparse waveguide distribution, which ensures no grating lobes.

## **3.4 Random Distribution**

This section introduces a more algorithmic way to create an OPA using a genetic algorithm. The idea is similar to Golomb distribution where grating lobes are suppressed by mixing the waveguide position in a random fashion. Unlike Golomb, this method can potentially create a larger array that is not limited by the possible optimal Golomb ruler.

### **3.4.1 Genetic Algorithm**

Global search algorithms, such as genetic algorithms, have been used to solve problems in nanophotonics, including fiber Bragg gratings [36], photonic crystal band gaps [37]. Genetic algorithms are inspired by natural evolution and operate by repeatedly selecting, varying, and replicating successful individuals in a population of candidate solutions [38]. These algorithms are well suited for finding solutions to problems dealing with complex search spaces that do not have smooth gradients leading to an optimum. Specifically, genetic algorithms are well suited for searching open-ended design spaces that are not optimally characterized by a finite set of parameters. [38].

## Mechanism

Important hyperparameters used in the Genetic Algorithm consist of:

- Population size: The number of random distributions taken into account in each generation. A population size of 100 is chosen for this simulation.
- Generation: The number of iterations that GA will run. The last generation of 1000 is chosen for this simulation.
- Crossover rate: probability of a child sharing some information from both parents. The crossover rate of 0.02 is chosen
- Mutation rate: probability of a child value replaced by a random value.

The genetic algorithm used here operates in these steps [39]:

- **Step 1: Initiation:** The algorithm will start with a population size of 100 of random distributions within our constraints such as the spacing range between two waveguides.
- **Step 2: Fitness evaluation:** Each distribution's far-field intensity will be calculated, and optimizing parameter fitness is computed. Fitness is the sum of the normalized magnitude of the center point at far field and the ratio of that peak and the second largest peak.
- **Step 3: Selection.** After the fitness calculation, the next distribution population will be selected using a fitness-dependent selection method. The selection

scheme used is rank selection, where the likelihood of a distribution being selected is proportionate to its rank in the population based on fitness. A selection probability is calculated by division of fitness by total fitness. The selected distributions become the mating pool for next generation population. Therefore, the mating pool will have a higher number of good distributions based on fitness, from which the next population will directly inherit.

- **Step 4: Variation.** Variation is an important factor in the Genetic Algorithm, as it prevents the program from converge to a distribution too quickly. Two parents are selected randomly from the mating pool to create a child. The child inherits half of each parent with a probability of `crossoverRate`; otherwise it replicates one of the parents. After that, each waveguide spacing is subject to vary by  $\lambda$  with a mutation rate.

## Result

As for optimizing algorithm, the convergence of the Genetic Algorithm depends on a number of factors, such as hyperparameters and nonlinear fitness-dependent selection.

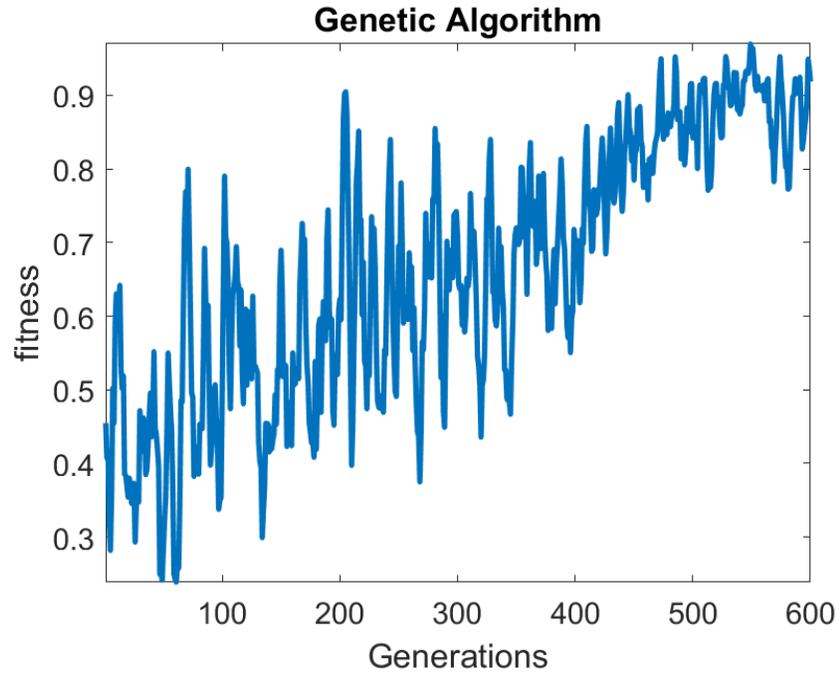


Figure 3.12: Genetic Algorithm : Fitness over generations

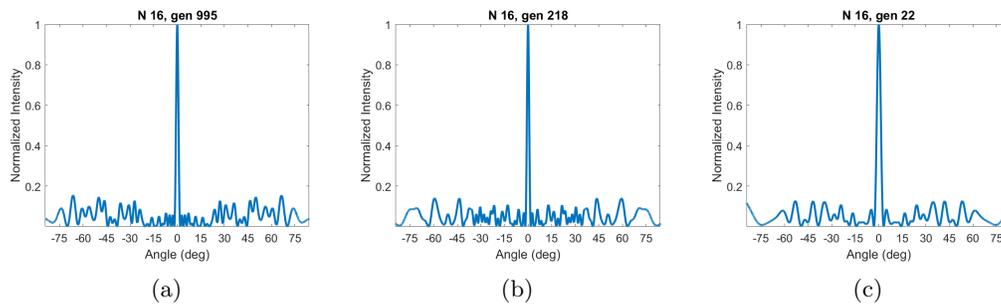


Figure 3.13: Far field of a best distribution at immediate generations

## 3.5 Phase Modulation Scheme

### 3.5.1 Theory

For nonconstant distributions, it is impossible to use a linear phase scheme to steer optically because the amount of phase compensation is different based on waveguide spacing. Therefore, an adaptive phase modulation scheme is required and presented to steer optical beams for an arbitrary distribution.

Consider two emitters with distances to a reference point  $N_x$ ,  $N'_x$ , and a far-field point A at location  $X$  and  $D$  at a coordinate shown in Fig. 3.14.

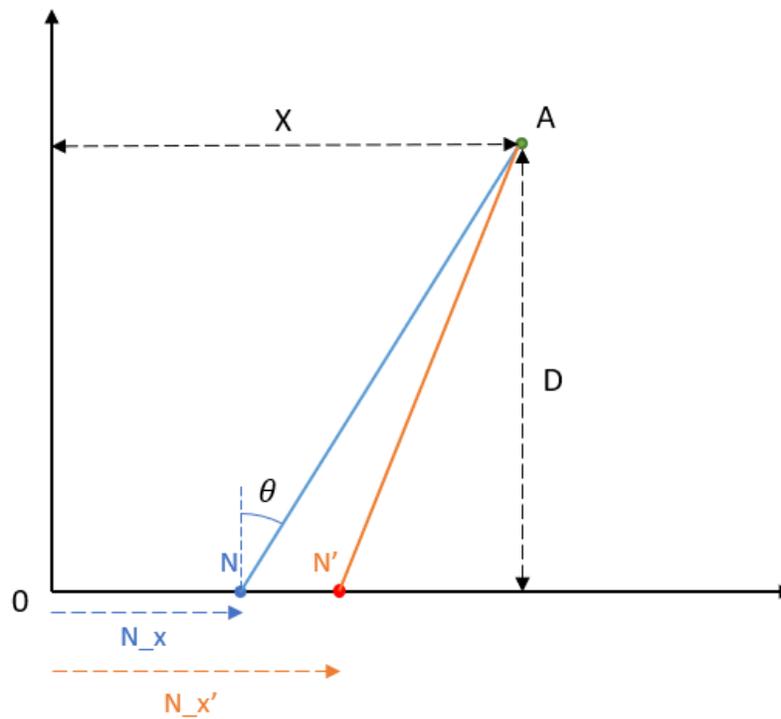


Figure 3.14: Coordinate setup for non-constant pitch OPA Phase Modulation Scheme

Assuming ideal point sources, the total phases of emitter N (called  $\varphi$ ) and emitter N' (called  $\varphi'$ ) to that point are

$$\begin{aligned}\varphi &= -k\sqrt{D^2 + (X - N_x)^2} + \phi \\ \varphi' &= -k\sqrt{D^2 + (X - N'_x)^2} + \phi'\end{aligned}$$

Using far field approximation,  $N_x, N'_x \ll X, D$ , the expression of the total phase becomes

$$\begin{aligned}\varphi &\approx -k\sqrt{D^2 + X^2 - 2XN_x} + \phi \\ &\approx -k\sqrt{(D^2 + X^2) \left(1 - \frac{2XN_x}{D^2 + X^2}\right)} + \phi \\ &\approx -k\sqrt{D^2 + X^2} \left(1 - \frac{2XN_x}{D^2 + X^2}\right)^{1/2} + \phi \\ &\approx -k\sqrt{D^2 + X^2} \left(1 - \frac{XN_x}{D^2 + X^2}\right) + \phi\end{aligned}$$

Similarly, the total phase of a random distribution can be derived:

$$\varphi' \approx -k\sqrt{D^2 + X^2} \left(1 - \frac{XN'_x}{D^2 + X^2}\right) + \phi' \quad (3.2)$$

Interference occurs when  $\varphi' = \varphi$ :

$$\begin{aligned}\phi' &= \phi + k \frac{X}{\sqrt{D^2 + X^2}} (N_x - N'_x) \\ \phi' &= \phi + k \sin \theta (N_x - N'_x)\end{aligned}\tag{3.3}$$

Similarly, when assuming Gaussian sources, the total phases are:

$$\begin{aligned}\varphi &= kD + k \frac{(X - N_x)^2}{2R(z)} - \arctan \left( \frac{D}{z_R} \right) + \phi \\ \varphi' &= kD + k \frac{(X - N'_x)^2}{2R(z)} - \arctan \left( \frac{D}{z_R} \right) + \phi' \\ \varphi' - \varphi &= \phi' - \phi + \frac{k}{2R_z} \left[ (X - N_x)^2 - (X - N'_x)^2 \right] \\ &= \phi' - \phi - \frac{k}{R_z} X (N_x - N'_x) \\ &= \phi' - \phi - \frac{kD}{R_z} \tan \theta (N_x - N'_x) \\ &\approx \phi' - \phi - k \tan \theta (N_x - N'_x)\end{aligned}$$

Interference occurs when  $\varphi' = \varphi$ , thus:

$$\phi' = \phi + k \tan \theta (N_x - N'_x)\tag{3.4}$$

### 3.5.2 Result

This phase modulation scheme will assign different phase shifts to each waveguide based on its distance to the reference OPA. For demonstration, we will choose  $\lambda/2$  constant spacing OPA as our reference waveguides and the steered angles from them as reference angles. For an N arbitrary OPA, each waveguide phase will be shifted by an amount from the scheme with the reference waveguide in order. Then, the arbitrary OPA will steer the same as the OPA with constant spacing  $\lambda/2$ , which is the ideal direction.

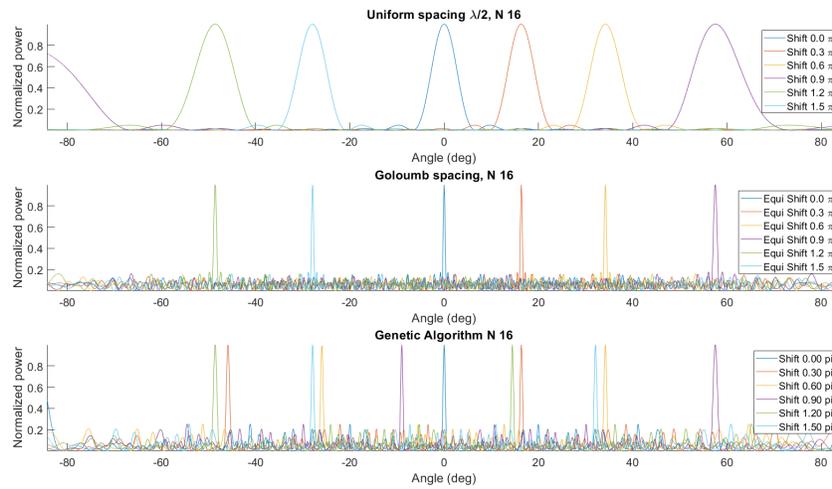


Figure 3.15: Simulation result of Phase Scheme for ideal sources

In the Gaussian beam case:

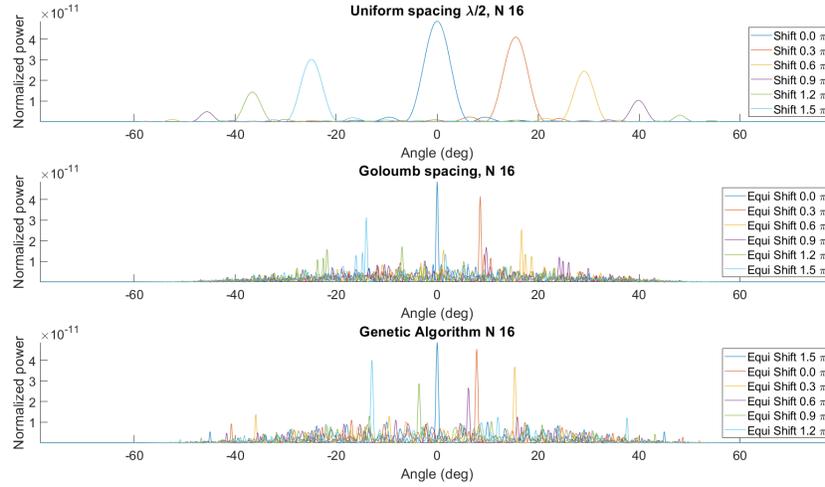


Figure 3.16: Simulation result of Phase Scheme for Gaussian sources

From the simulation results, we achieve 180-degree steering with Golomb distribution without small spacing between waveguides. When an emitted wave is Gaussian, the steering angle is reduced due to the radial attenuation and wavefront curvature; this method can also apply to steer any distribution of OPA.

# Chapter 4

## Modeling

### 4.1 Setup

Finite-difference time domain (FDTD) is a numerical technique used to solve three-dimensional (3D) Maxwell equations in the time domain. This technique is handy for analyzing the interaction of light with complicated structures employing sub-wavelength-scale features. FDTD simulates the propagation of a light pulse that contains a broad spectrum of wavelength components. The system's response to this short pulse is related to the transmission spectrum via the Fourier transform.

In FDTD, the simulation region is divided into a grid of small cells, and the values of the electric and magnetic fields are calculated at each cell based on the updated equations derived from Maxwell's equations. The update equations relate the field values at each cell to the neighboring cells in both space and time domains. The update equations can be derived using various numerical methods such as the central

difference method or Yee's scheme.

FDTD simulations can handle complex geometries and material properties, including anisotropy, dispersion, and nonlinearities. However, FDTD simulations can be computationally intensive and require significant memory and processing power, especially for large simulation regions.

Finite Difference Eigenmode (FDE) is a numerical technique that is used to find the eigenmodes of a structure in the frequency domain. Eigenmodes are the electromagnetic modes that can exist in a given structure that can propagate without changing its shape. Using the finite-difference method, the FDE solver determines the time-harmonic solutions to Maxwell's equations in a discretized finite number of cells within the simulation region.

The finite-difference method involves approximating the derivatives of the electric field with finite differences and solving the resulting matrix equation to obtain the eigenmodes and their corresponding frequencies.

FDE simulations are computationally efficient because they only need to be run once to obtain the eigenmodes and their corresponding frequencies. Once the eigenmodes are obtained, they can be used to calculate the device parameters for a range of input frequencies without the need for additional simulations.

One of the strengths of FDTD is its ability to handle a wide range of material properties, including anisotropy, dispersion, and nonlinearities, making it suitable for simulating the behavior of materials commonly used in photonics. In addition, FDTD can be used to simulate the effect of various parameters on the device performance,

such as input wavelength, polarization, and angle of incidence, which is important for optimizing the device design.

On the other hand, FDE is particularly useful for analyzing the fundamental electromagnetic modes of photonic devices, such as waveguides, resonators, and optical fibers. The eigenmodes obtained from FDE simulations can calculate various device parameters, such as effective refractive index, dispersion, and mode volume. FDE is also useful for optimizing the device geometry and understanding the coupling between different modes.

## **4.2 Waveguide**

Waveguide is the most fundamental building block in any photonic circuit, and the design of a suitable waveguide is essential for robust, high-performance circuits. Waveguide design consideration includes the material for meeting specifications at a wavelength and the waveguide's geometry and dimension.

### **4.2.1 Material Consideration**

#### **Silicon on Insulator (SOI)**

The mainstream silicon photonics circuits are based on silicon-on-insulator (SOI) wafers, where a silicon layer is buried in silicon oxide, forming the core layer to print waveguides. The exploration of alternative material combinations that satisfy the high index contrast and CMOS compatibility is growing in popularity. The re-

quirement to operate in spectral bands where silicon absorbs, particularly the visible spectrum, is a key factor for departing from SOI. Fig. 4.1 shows the high refractive index contrast between Si and SiO<sub>2</sub> and the high extinction coefficient (absorption) of Si at a wavelength. Data is experimentally collected for Si [40], and SiO<sub>2</sub> [41]. Silicon nitride is a significant alternative to silicon for the core and still maintains CMOS compatibility [42].

### **Silicon Nitride**

Silicon nitride (SiN), which has a refractive index of about 2.0 at a wavelength of 1550 nm, is frequently used in CMOS fabrication processes. Compared to silicon, it has a number of advantages, including:

- Transparency range: SiN is a good candidate for applications below 1  $\mu\text{m}$  because it is transparent in most of the visible spectrum to at least 500 nm. This resulted in the demonstration of spectroscopic function and Raman spectroscopy on-chip functionality [43].
- Low loss: SiN waveguides (clad in silica) have a refractive index contrast lower than that of SOI, so this reduces the waveguide loss from scattering. In arrayed waveguide gratings, higher index contrast makes the waveguide more sensitive to sidewall roughness and crosstalk.

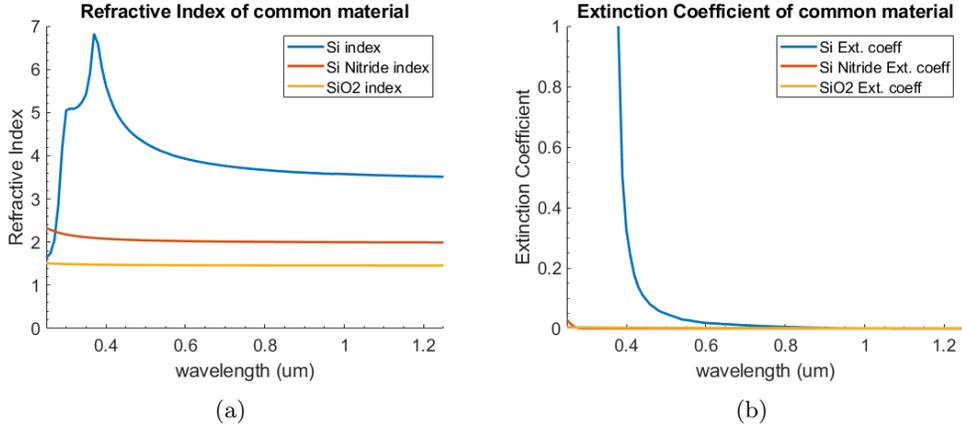


Figure 4.1: (a) Refractive index of Si, SiO<sub>2</sub>, SiN from 0.25  $\mu\text{m}$  to 1.25  $\mu\text{m}$  (b) Extinction Coefficient of Si, SiO<sub>2</sub>, SiN from 0.25  $\mu\text{m}$  to 1.25  $\mu\text{m}$

#### 4.2.2 Geometry and Dimension

There are several types of waveguides that are used in silicon photonics. A strip waveguide is used for routing as it offers tight bend radii. The rib waveguide (also known as ridge waveguide) is used for electro-optic devices such as modulators, since it allows for electrical connections to be made to the waveguide.

Light can be confined inside a waveguide due to the internal interference between the core and the cladding, which has lower refractive index material. As light travels along the waveguide, only constructive interference between reflected light is supported, creating an optical mode. The number of supported modes in a waveguide depends on the width and thickness of the waveguide. A waveguide is made to support the fundamental mode in the majority of photonics applications since it is preferable to minimize modal dispersion.

The general process that must be followed in order to determine a strip waveguide's dimension is as follows:

- First, create a waveguide model in Lumerical using placeholder width and thickness. Add a 1D FDE solver at a point on the symmetry line of the waveguide, which spanned the maximum thickness in the sweep. For each thickness in the sweep, extract the refractive index for each mode (TE, TM, fundamental, and higher order mode). The thickness is set to give the fundamental mode the highest effective index, while the first-order mode effective index just starts rapidly increasing. From the result presented in Fig. 4.2, a thickness of 150 nm is chosen.

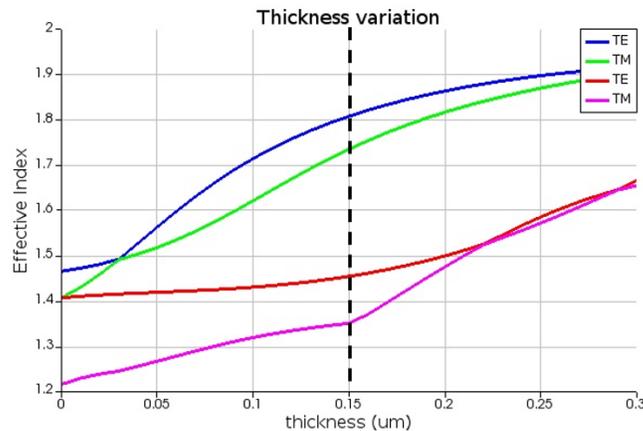


Figure 4.2: Effective index result of SiN slab - thickness sweep from 0 to 0.3  $\mu m$

- Once the thickness is determined and the waveguide model is updated, a 2D FDE solver is placed around the waveguide and normal to the light propagation.

Similarly to thickness selection, a width of 300 nm is chosen based on the result in Fig. 4.3.

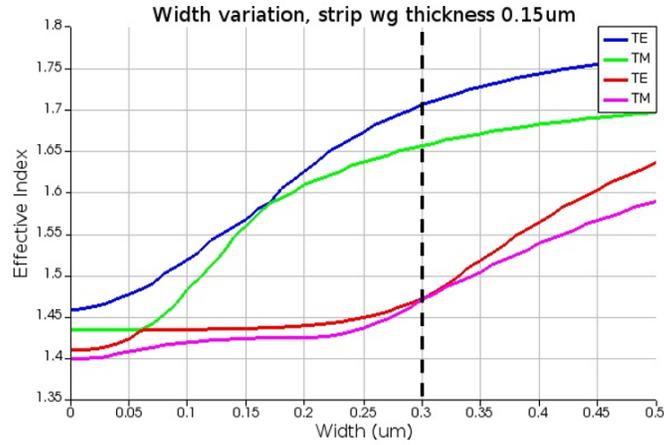


Figure 4.3: Effective index result of SiN strip - waveguide width sweep from 0 to 0.5  $\mu m$

After determining the waveguide dimension, the fundamental TE, TM modes in the strip design are presented in Fig 4.4 which yields effective index of 1.71. The effective index is lower than the intrinsic effective index of SiN due to the effect of silica cladding and the size of the waveguide.

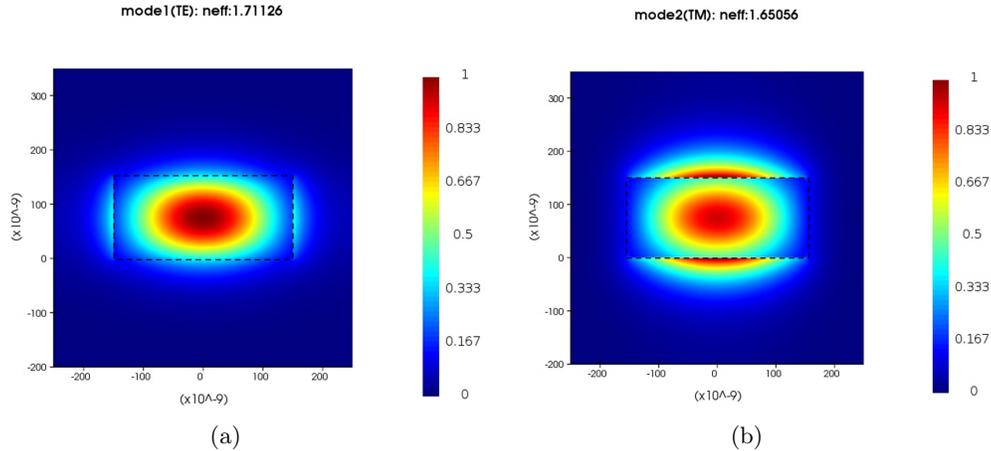


Figure 4.4: FDE simulation of fundamental mode of 300 nm x 150 nm SiN strip waveguide (a) TE mode (b) TM mode

### 4.3 OPA Array

An OPA array is a collection of multiple single waveguides placed in a spacing distribution. As discussed in the previous chapter, it is known that it is ideal for putting waveguides (emitters) at a half-wavelength pitch to each other. However, the optimal distribution is later shown in the FDTD to suffer from excessively high coupling. For demonstration purposes, an array of 7 waveguides is used to evaluate different spacing distributions based on coupling.

#### 4.3.1 varFDTD

The varFDTD approach in MODE Solutions enables the broadband modeling of linear and nonlinear processes in planar waveguide systems in contrast to the FDTD,

which requires extensive computation. The varFDTD approach for planar waveguide components provides precision equivalent to those of 3D FDTD while only requiring the simulation time and memory of a 2D FDTD simulation. The varFDTD solver is ideal for simulating omnidirectional light propagation in optically significant planar-integrated optical components. The method involves reducing the 3D problem to an effective 2D problem by converting the vertical waveguide structure into effective dispersive materials that simultaneously account for material and waveguide dispersion based on the Hammer and Ivanova variational procedure [44] or the reciprocity Theorem [45]. The premise of both theories is that there is negligible coupling between the different slab modes supported by the vertical waveguide structure. This is a fairly good assumption for many devices, such as SOI-based slab waveguide architectures, that enable only two vertical modes with distinct polarizations. With the help of this tool, designers may quickly iterate through a variety of design settings and simulate components that are too big to be handled using 3D FDTD.

Fig 4.5 shows effective index estimation of the OPA of 11 waveguides, which yields average around 1.8 in SiN waveguides. This differs approximately 10% from the actual refractive index obtained from FDTD.

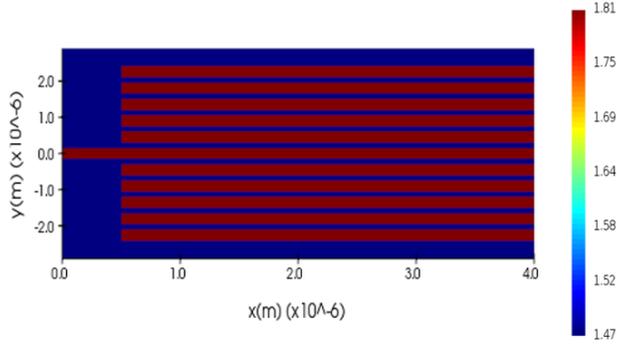


Figure 4.5: varFDTD effective index estimation of a 11-element OPA

### 4.3.2 Spacing pattern

This distribution is the simplest form in which every waveguide is spaced at a constant distance from each other. The distance between two waveguides is defined as the distance between the symmetric axis of the two waveguides. Due to the waveguide width of 300 nm, it is impossible to place the waveguides at half-wavelength pitch (225 nm). Therefore, the smallest spacing tested is 500 nm, which results in 200 nm spacing wall-to-wall.

The simulation region includes an OPA of 7 waveguides positioned similarly to Fig. 4.5 where the waveguide length varies depending on the waveguide pitch. Specifically, a region with a length of 200  $\mu m$  is used to simulate a spacing below 0.75  $\mu m$ , while lengths of 800  $\mu m$  and 1600  $\mu m$  are required to demonstrate the total coupling of the center waveguide to surrounding waveguides at 750 nm and above the spacing. A small model of the 20  $\mu m$  array is shown in Fig. 4.6. Then, varFDTD solver and field monitor are added to cover the entire simulation region, which calculates elec-

tric and magnetic at every point based on varFDTD technique. Last but not least, a mode source is placed at the center waveguide and emits a fundamental mode at 450 nm with 0.25  $\mu m$  variation, which is shown in Fig. 4.7.

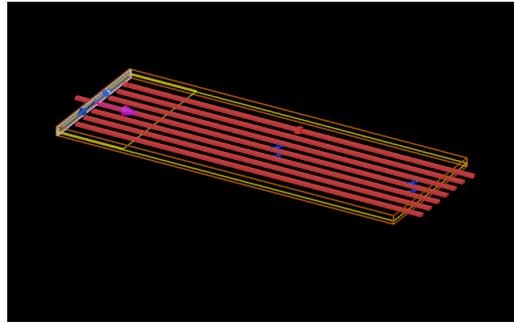


Figure 4.6: FDTD OPA Simulation Region of length 20  $\mu m$

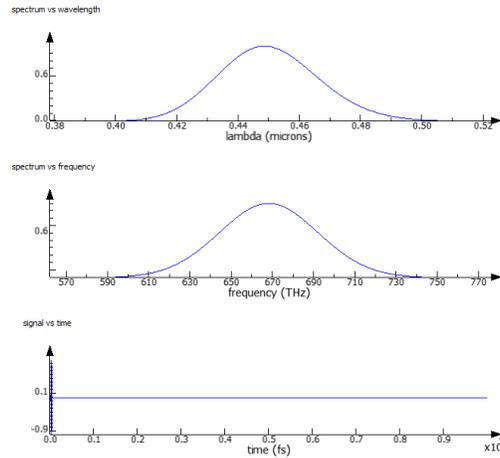


Figure 4.7: Input source in terms of spectrum and time

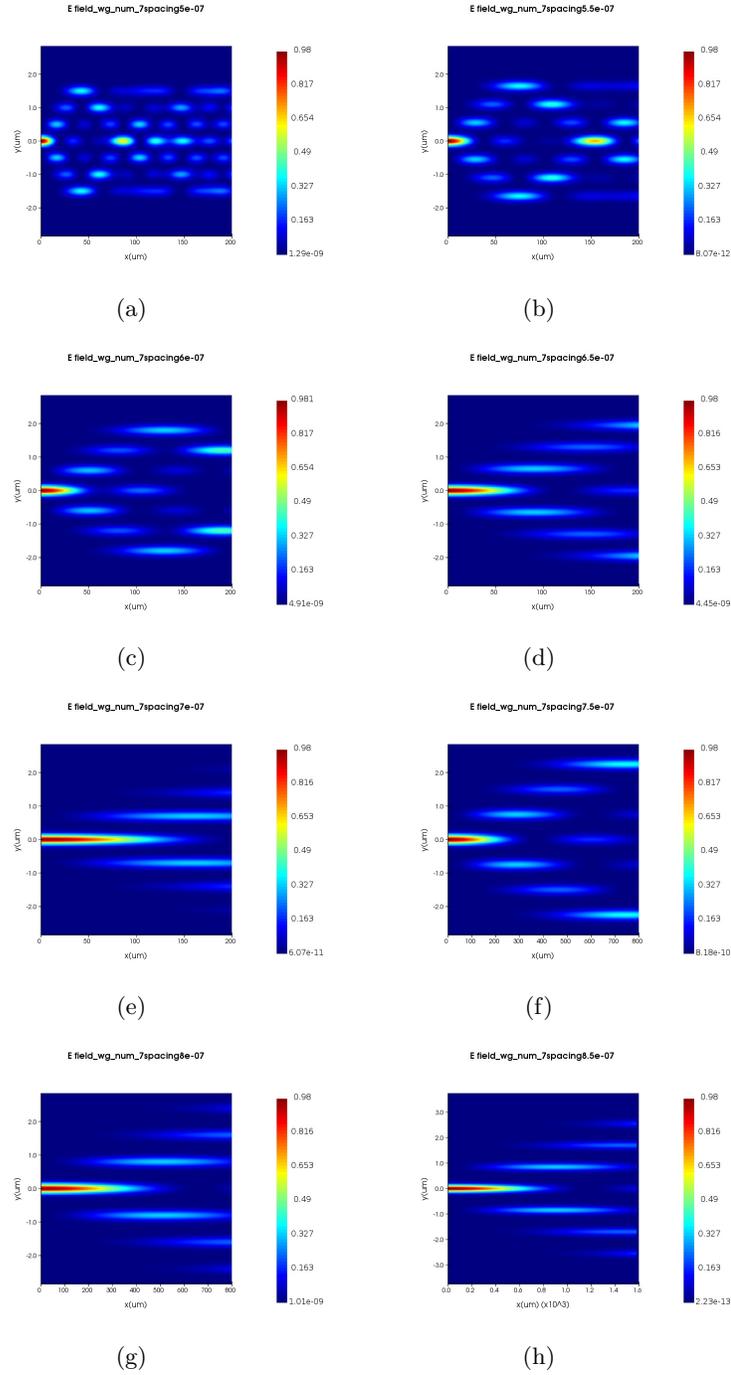


Figure 4.8: FDTD simulation of constantly spaced OPA from 500 nm to 750 nm 50 nm interval (order from left to right, top to bottom)

Fig. 4.8 shows the FDTD simulation of the OPA at a spacing of 500 nm to 750 nm when light is input into the centered waveguide. From the result, it is clear that this type of spacing distribution experiences high coupling for spacing less than 700 nm. To better quantify the coupling effect, coupling length is used, which is defined as the distance where light intensity comes down to 0 at the central waveguide. For example, a 500nm-spaced array will have a coupling length of 20  $\mu m$  based on Fig. 4.9 (a). Similarly, the coupling length for spacing from 500 nm to 850 nm is collected and reported in Fig. 4.9.

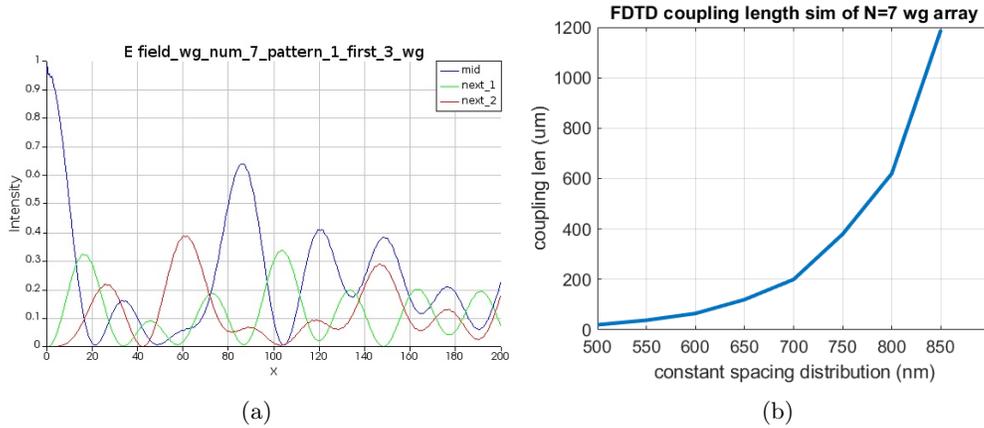


Figure 4.9: (a) Intensity along central waveguide, the closest waveguide, the second closest waveguide at 500 nm spacing (b) Coupling length vs spacing

The result shows the exponential dependence of coupling length to spacing, which agrees with the theory of directional coupler. For our intended array length of 50 nm, spacing smaller than 800 nm would yield unacceptable crosstalk.

### 4.3.3 Width pattern

A solution to reduce coupling without increasing the spacing is to vary the width of the waveguide in some periodic sequence. This idea is inspired by the period emitter spacing to lower the crosstalk due to modal mismatch. Four different width patterns are tested where the widths are chosen to minimize the higher-order mode in a waveguide (Table 4.1).

Table 4.1: Waveguide width pattern

| Pattern | Widths ( $\mu m$ )   |
|---------|----------------------|
| 1       | 0.3, 0.35            |
| 2       | 0.3, 0.45            |
| 3       | 0.3, 0.4, 0.5        |
| 4       | 0.3, 0.35, 0.4, 0.45 |

The previous simulation model can be reused directly with one modification being that the width of waveguides will rotate through the pattern in order. The width pattern has been constructed so that no waveguide overlaps or touches. For example, the simulation region for pattern 4 is shown in Fig. 4.10.

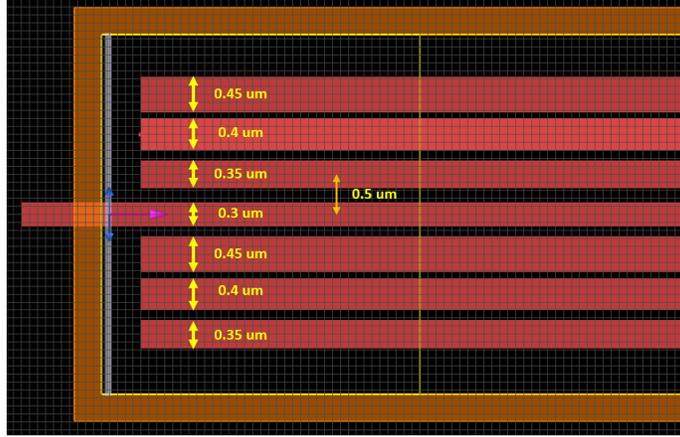


Figure 4.10: Simulation region for width pattern 4

The result in Fig. 4.11 is arranged so that the rows represent the different width patterns 1 to 4 from top to bottom, and the columns show waveguide spacing 500 nm (left column) and 550 nm (right column). From the result, there are a few following points that can be deduced:

- Adding a small perturbation to the width ( $0.3 \mu m$  to  $0.35 \mu m$ ) significantly improves the coupling length. At 500 nm spacing, the coupling length increases from  $20 \mu m$  to  $60 \mu m$  with additional  $0.35 \mu m$  width waveguides.
- Despite mode mismatch in different waveguide widths, coupling will still happen at a very close distance. In pattern 3, at 500 nm spacing, the wall-to-wall distance between the center waveguide ( $0.3 \mu m$ ) and the closest waveguide ( $0.5 \mu m$ ) is only  $0.1 \mu m$ .
- Small variation like in pattern 4 eliminates most coupling at 500 nm pitch.

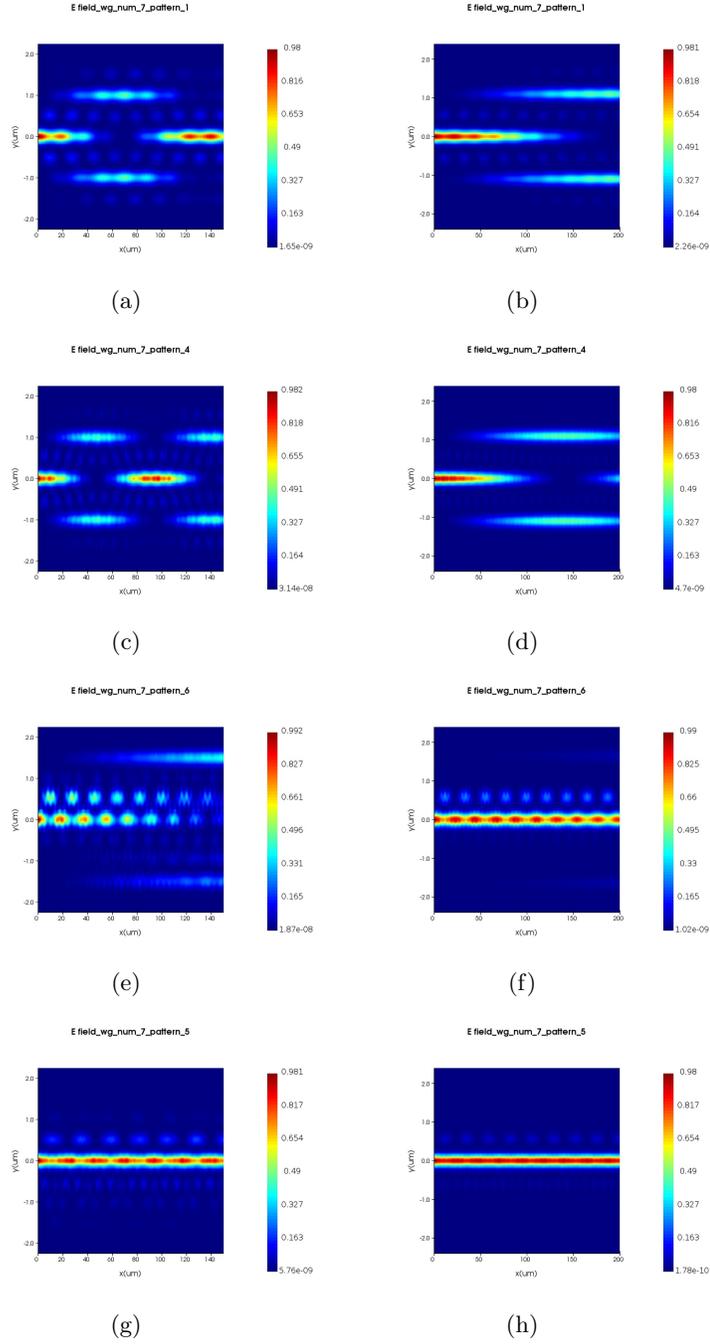


Figure 4.11: FDTD Simulation of widths patterns (in rows: top to bottom is pattern 1 to 4 consecutively) at 500 nm (left column) and 550 nm (right column) spacing

## Chapter 5

# Layout and Preliminary

# Experiment on Ring Resonator

### 5.1 Layout

#### 5.1.1 Tools

With the growth in scale, photonics has seen an ever-increasing number of supporting tools that automate and facilitate the development of photonic chips. Compared to microelectronics, the silicon photonics design ecosystem is still very much in the development phase, especially at the system level, layout, and testing. PDKs from dedicated photonics foundries such as AIM Photonics, Applied Nanotool, and SiEPIC have been created. In addition, several lightweight free, open-source tools have been developed that assist in the layout process and greatly accelerate chip

prototyping. Two popular Python-written libraries are `gdstl` and `gdspy`, which can design layouts, check DRC rules, and run basic simulations. Other similar tools can be named: GDSHelp [46], Berkeley Photonics Generator [47], Nazca, and DPhox. In this work, Nazca and KLayout are used exclusively for quick prototyping.

### **Nazca**

Like most other photonic layout tools, Nazca is a Python-based library that makes it simple to draw layouts and export gds files for photonic components. It is a popular tool for photonics designers thanks to its pre-built library components and tight integration with KLayout, which is an open-source layout editor and viewer.

### **KLayout**

As of today, KLayout is the popular open-source layout tools due to its simplicity and practical functionality. Being both a viewer and editor tool, it can make quick fixes on a variety of formats such as GDS2, OASIS, DXF, CIF, Gerber, and LEF/DEF.

#### **5.1.2 Photonic chip GDS**

The chip contains eight different OPA designs in an area of 8.78 mm x 8.78 mm, which are listed in Table 5.1 and laid out as in Fig. 5.1. All the designs are edge-emitting to three sides of the chip, and each laser has a separate laser input from the top side of the chip. A small area on the chip that is not used by OPAs is used

to test different grating couplers and individual waveguides.

Table 5.1: Six OPA designs on the chip layout

| Design | Number of Emitter | Spacing Distribution  | Width pattern        |
|--------|-------------------|-----------------------|----------------------|
| 1      | 16                | constant $0.6 \mu m$  | Pattern 4            |
| 2      | 16                | constant $0.75 \mu m$ | Constant $0.3 \mu m$ |
| 3      | 16                | Golomb                | Constant $0.3 \mu m$ |
| 4      | 16                | Random from GA        | Constant $0.3 \mu m$ |
| 5      | 8                 | constant $0.9 \mu m$  | Constant $0.3 \mu m$ |
| 6      | 4                 | constant $0.75 \mu m$ | Constant $0.3 \mu m$ |
| 7      | 4                 | constant $0.6 \mu m$  | Constant $0.3 \mu m$ |
| 8      | 8                 | constant $0.75 \mu m$ | Constant $0.3 \mu m$ |

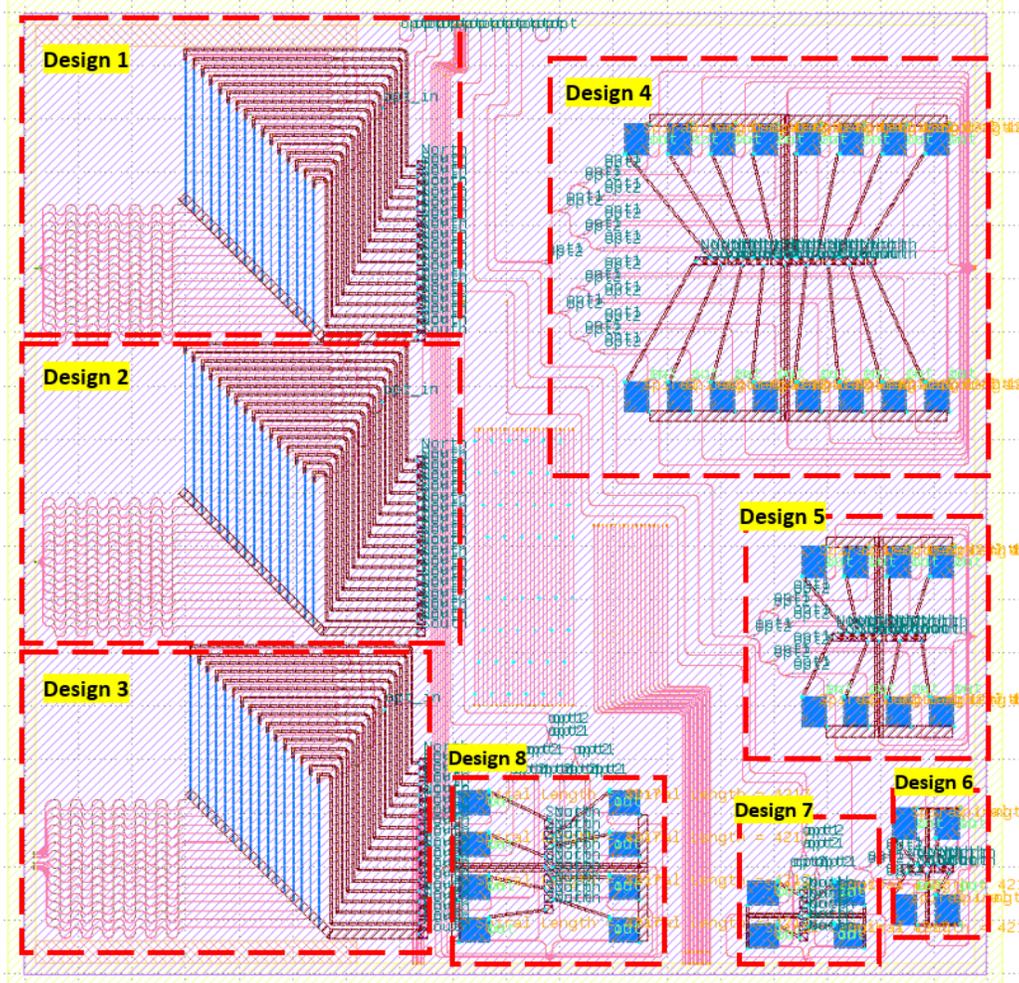


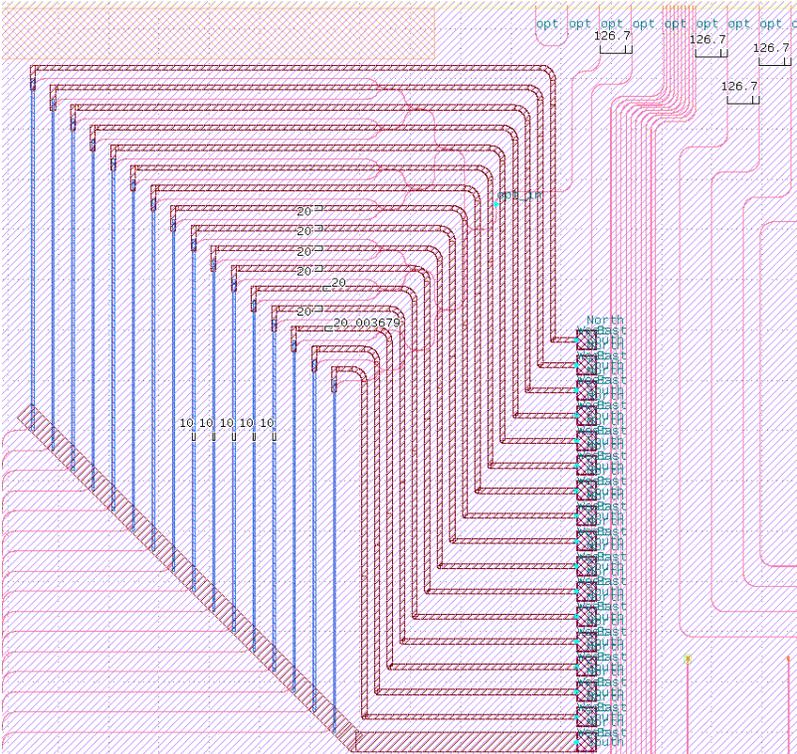
Figure 5.1: Chip Layout

### Input - Split stage - Trombone

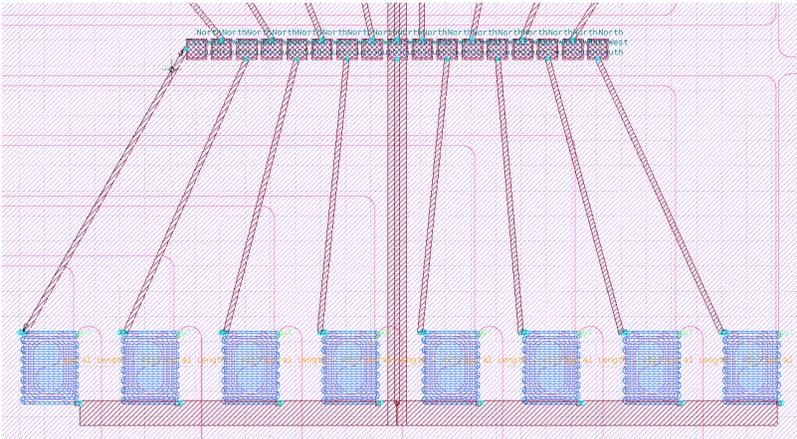
Light entering each OPA design will first go through a similar series of edge-coupling laser input, stages of Y-splitter, and trombones before reaching the phased



Phase shifter



(a)



(b)

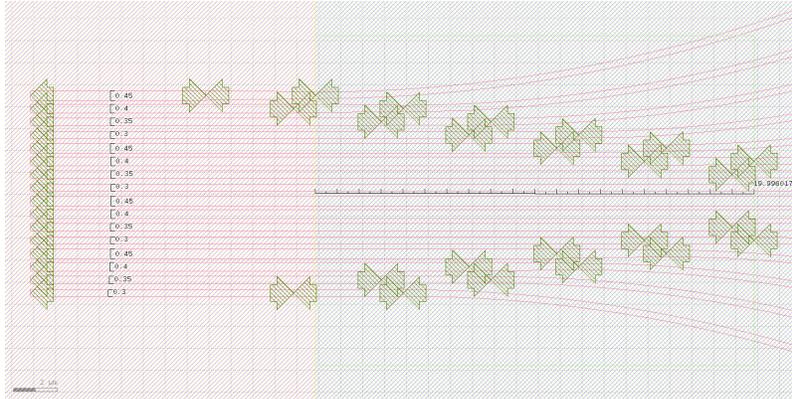
Figure 5.3: Two different phase shifter designs (a) for design 1-3 (b) for design 4-8

### Individual design

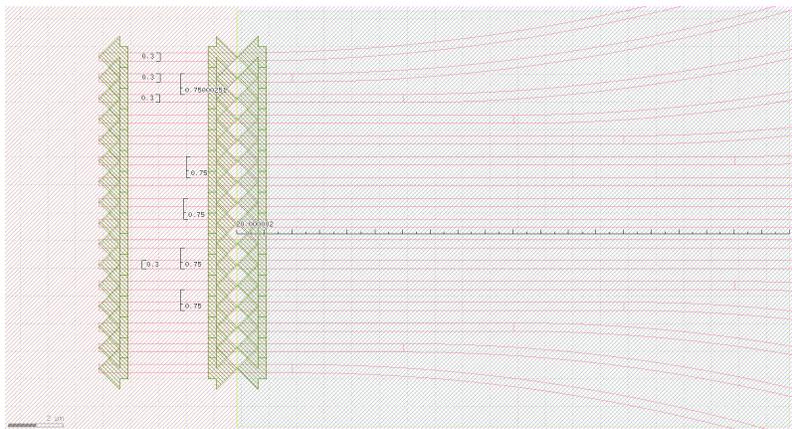
Fig 5.4 shows layouts of the OPA array for designs 1 and 2. Design 1 imposes a constant pitch of  $0.6 \mu m$  and width pattern 4 which are simulated in the previous chapter. Since design 2 has a constant width of  $0.3 \mu m$ , its pitch has to increase to  $0.75 \mu m$ . Both designs have an OPA length of roughly  $20 \mu m$ .

Fig 5.5 shows the layouts of the OPA array for designs 3 and 4. Design 3 imposes a Golomb spacing distribution for  $N$  equal to 16 and a constant width. Since design 4 has a constant width of  $0.3 \mu m$  with random spacing distribution obtained by the genetic algorithm. The distribution of waveguide spacing 2 from top to bottom is  $2.7 \mu m, 0.9 \mu m, 1.8 \mu m, 3.15 \mu m, 4.05 \mu m, 2.25 \mu m, 2.25 \mu m, 1.35 \mu m, 2.25 \mu m, 0.9 \mu m, 3.6 \mu m, 3.15 \mu m, 2.25 \mu m, 1.8 \mu m$ .

Fig. 5.6 shows the layouts of the OPA array for design 5, 6, 7, 8 which are various combinations of the number of waveguides (4, 8) and spacing (0.9, 0.75, 0.6). This will show the effect of waveguide number and coupling owing to the small spacing in the far-field pattern.

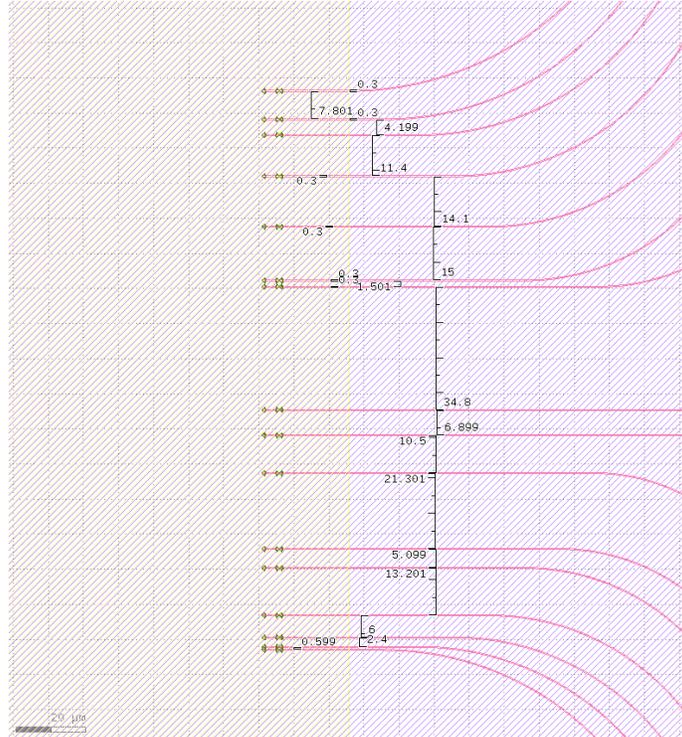


(a)

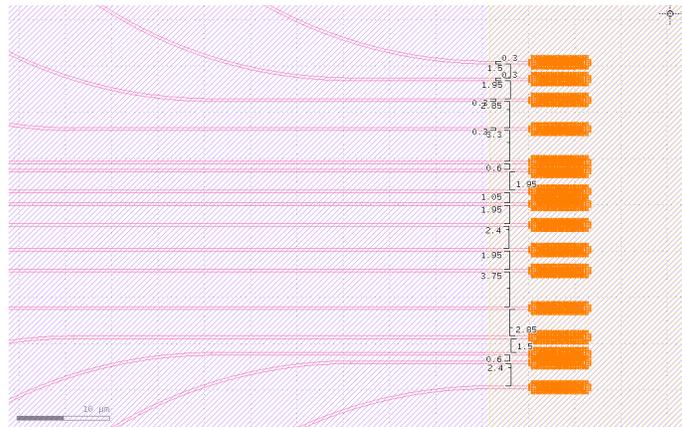


(b)

Figure 5.4: (a) OPA Design 1 - Constant Spacing - Width pattern 4 (b) OPA Design 2 - Constant Spacing  $0.75 \mu m$  - Constant Width

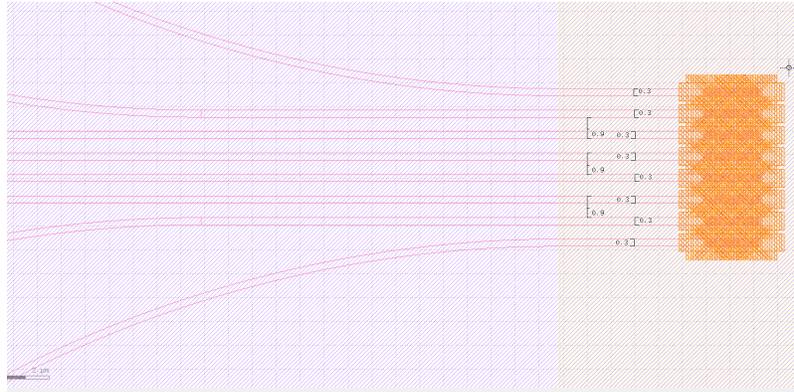


(a)

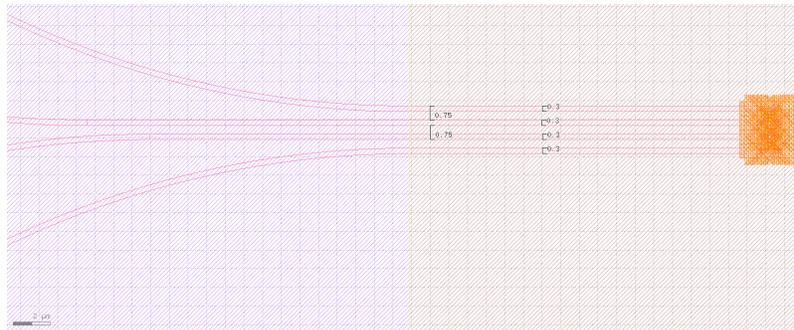


(b)

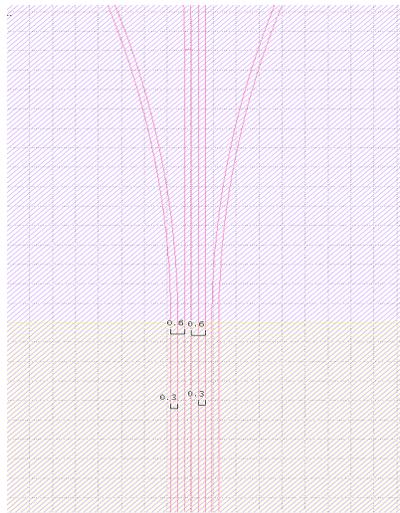
Figure 5.5: (a) OPA Design 3 - Golomb N 16 (b) OPA Design 4 - Random Distribution



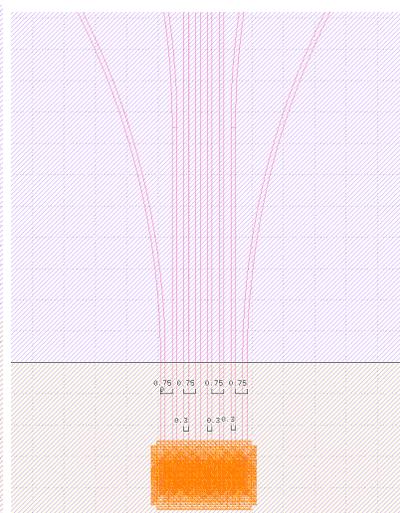
(a)



(b)



(c)



(d)

Figure 5.6: (a) OPA Design 5 - N 8 - Constant width - Constant spacing  $0.9 \mu m$  (b) OPA Design 6 - N 4 - Constant width - Constant spacing  $0.75 \mu m$  (c) OPA Design 7 - N 4 - Constant width - Constant spacing  $0.6 \mu m$  (d) OPA Design 8 - N 8 - Constant width - Constant spacing  $0.75 \mu m$

## 5.2 Preliminary Experimental Result of Blue wavelength

To demonstrate the viability of Silicon Nitride photonics on blue wavelengths, preliminary experiments have been performed prior to testing the OPA designs. The experiment tests the performance of the ring resonators with respect to the gap.

### 5.2.1 Ring Resonator

A ring resonator consists of a loop of the optical waveguide, which can be the shape of a ring or racetrack, with some coupling to the outside world. The loop creates a resonant cavity that traps light at a specific wavelength. There are two configurations of ring resonators, all-pass, and add-drop, as shown in Fig. 5.7.

Ring resonators can be used to filter and modulate light as a result of their temperature-dependent wavelength selection. Its unique properties, such as Q-factor, low insertion loss, and minimal footprint, make it an attractive candidate for developing compact, highly sensitive sensors and high-performance photonic circuits.

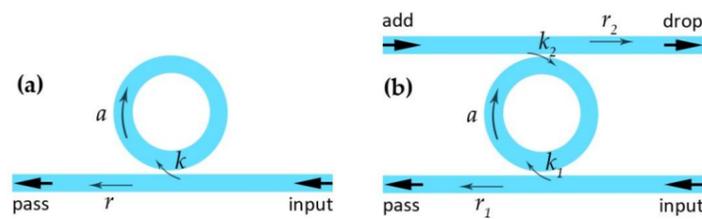


Figure 5.7: Ring resonator configurations (a) all pass (b) add-drop

## 5.2.2 Experimental Setup

The experiment will evaluate the performance of the ring resonators across six different gaps: 100, 150, 200, 250 and 300 nm. First, a light beam is generated by an ultrafast Coherent laser, then passed to an ultra-narrow-linewidth tunable SolsTIS M-square. The operating wavelength used in this experiment is in the range of 980 nm - 990 nm. Then, the laser beam goes via optical fiber to an optically non-linear Lithium niobate (LiNO<sub>3</sub>) crystal to double the frequency by second-harmonic generation. The crystal generates blue light in the 490 nm - 495 nm range, which will be coupled to a PIC through free-space optics. This chip contains an array of add-drop 10- $\mu$ m radius ring resonators that are edge-coupled and edge-emitting. In addition, a temperature controller is used to find the optimal temperature at an output wavelength. Two detectors measure the input power to the PIC and the output power of the add-drop port. Fig. 5.8 summarizes the setup and also shows an example ring resonator at its resonance.

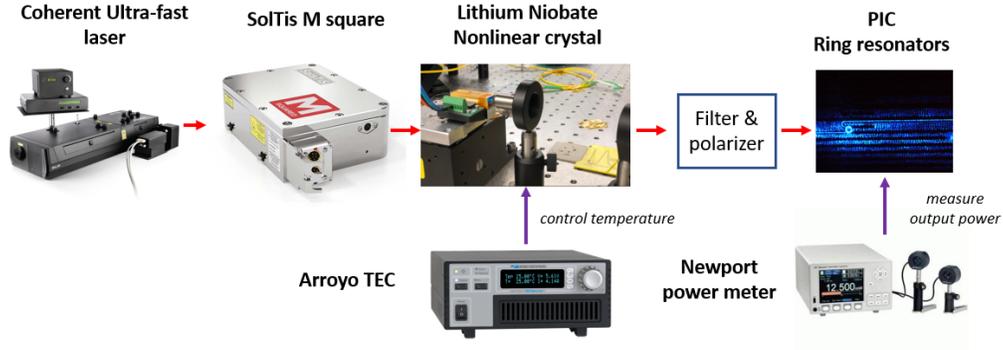


Figure 5.8: Experimental setup for ring resonators at blue wavelengths

### 5.2.3 Resonance Result

#### Crystal Calibration

Fig 5.9 shows the relationship between optimal temperatures and output wavelength. The dependency is linear and relatively weak, with a slope of 12 degrees celsius change per 1 nm frequency. This, however, also limits the wavelength testing range of roughly 490 nm - 495 nm because of the constraint crystal's operating temperature.

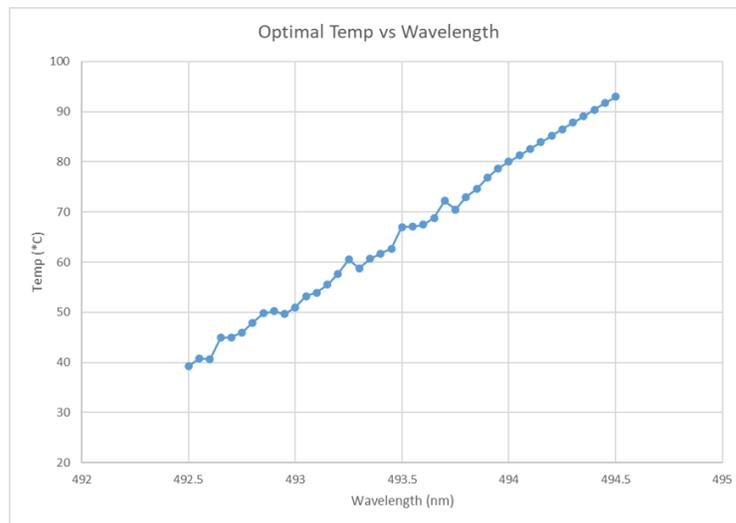


Figure 5.9: Crystal calibration: Optimal temperature vs output wavelength

#### Resonance Results

When sweeping in the testing wavelength range with adjusted temperature, it is visually clear that light is completely coupled to the ring at a certain wavelength,

leaving the through port with little power. Part of the power will connect to the drop port and is measured. Fig. 5.10 shows a top-down view of a ring resonance at 990.114 nm. Two such resonance wavelengths were found during the sweep, and data around those resonance points were recorded and normalized relative to the maximum value. Fig. 5.11, 5.12, 5.13, 5.14, 5.15 shows the experiment at two resonances for each gap.

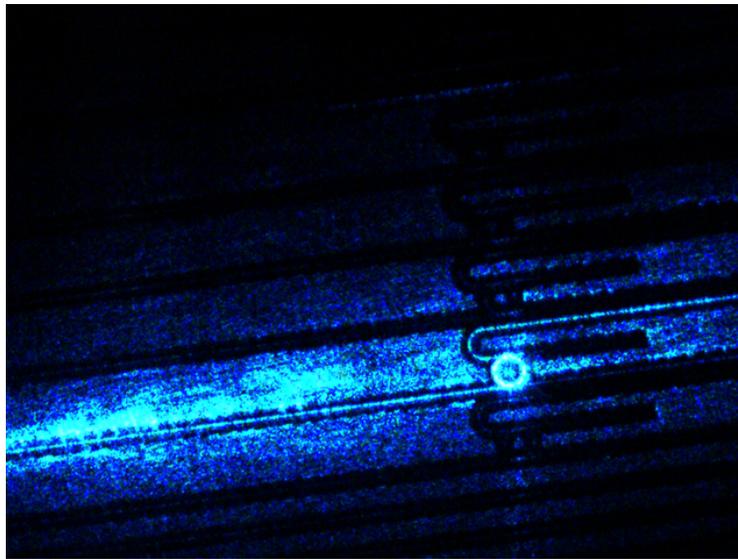


Figure 5.10: Top down view of a ring resonance at resonance at 990.114 nm

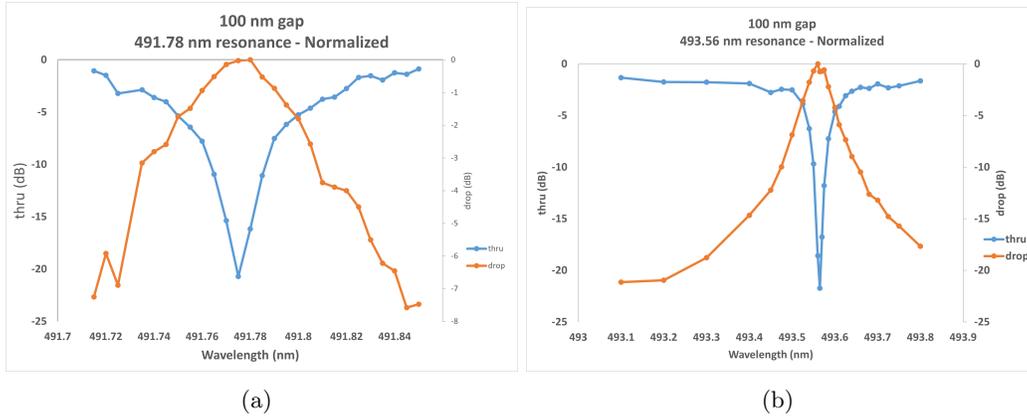


Figure 5.11: 100 nm gap (a) First resonance (b) Second resonance

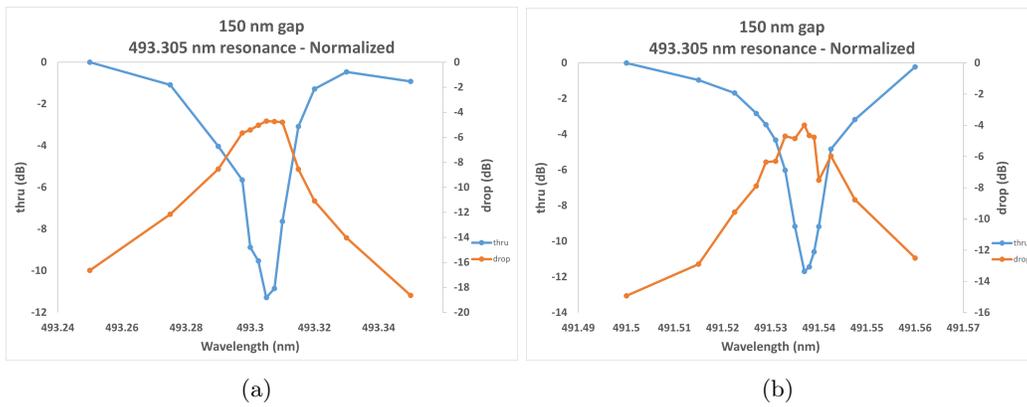


Figure 5.12: 150 nm gap (a) First resonance (b) Second resonance

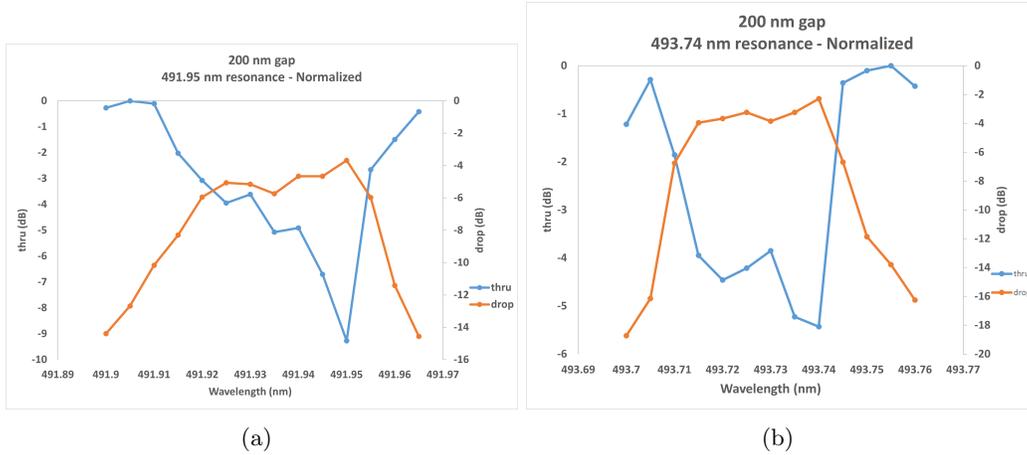


Figure 5.13: 200 nm gap (a) First resonance (b) Second resonance

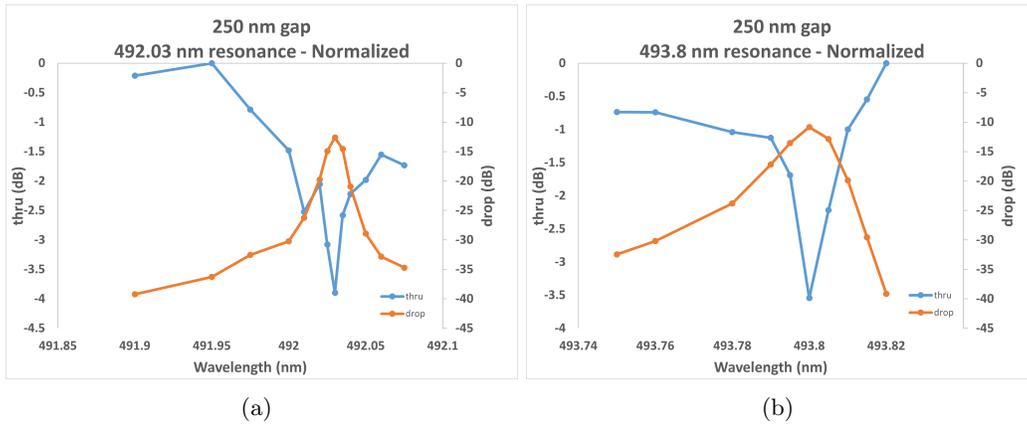


Figure 5.14: 250 nm gap (a) First resonance (b) Second resonance

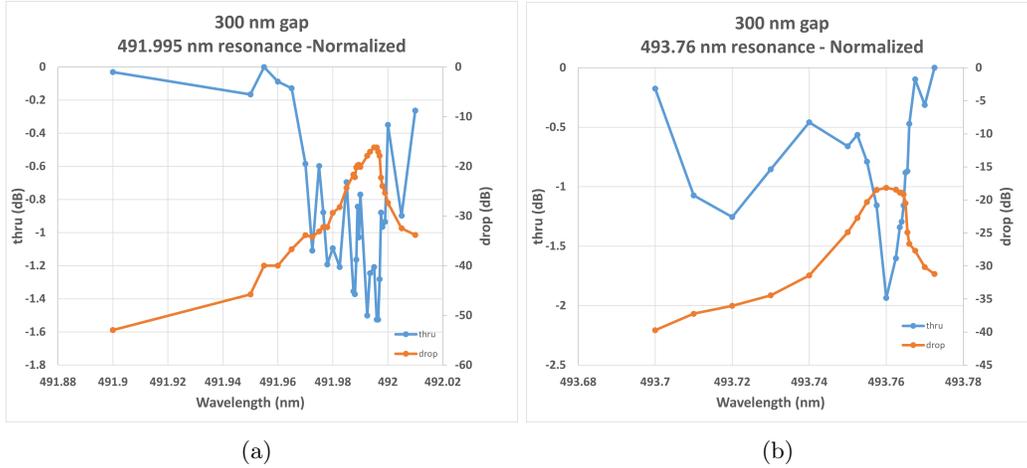


Figure 5.15: 300 nm gap (a) First resonance (b) Second resonance

The experimental results have demonstrated resonances in ring resonators at different gaps. The larger the gap, the less power is coupled to the drop port at resonance. At the 100 nm gap, effectively all power is coupled to the drop port, leaving -20 dB power in the through port. At the 300 nm gap, the drop power is recorded at resonance at -18 dB, while the through port power drops only by -1.4 dB. As shown in Fig. 5.16. The resonance wavelength changes slightly with the gap, but the free spectral range (FSR), defined as the wavelength difference between resonances, is relatively constant. The FSR depends only on the input wavelength, which aligns with our data and the data achieved in [48]

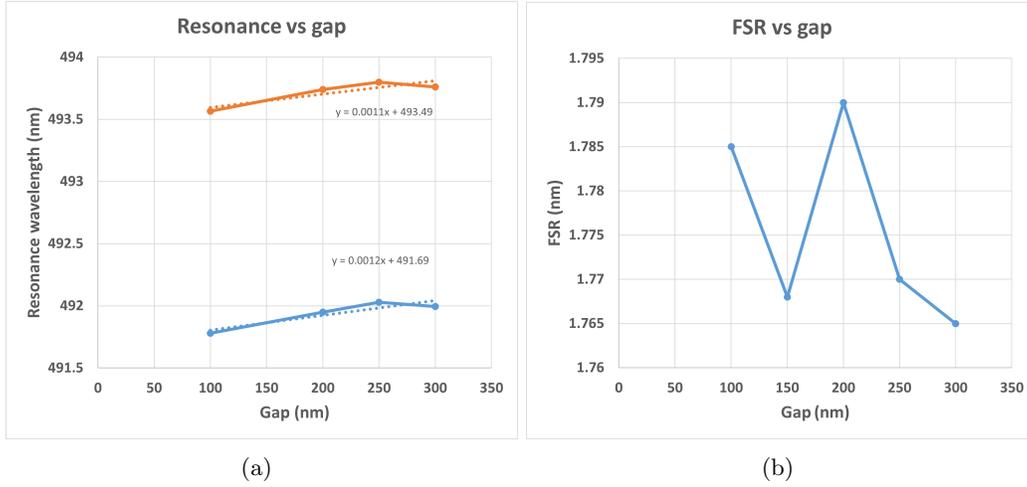


Figure 5.16: (a) Resonance wavelength vs gap (b) FSR vs gap

## Chapter 6

# Conclusion

In this work, we propose various new OPA designs that are suitable for blue wavelengths and avoid a half-lambda pitch while maintaining a large steering angle and a small beam width.

We demonstrate the operation of an OPA and the undesirable effect of OPAs with a spacing greater than half the wavelength. Then, we experiment with the Golomb distribution to create a sparse array to optimize the aperture and 180 degrees of steering capability. We also devised a fully parameterized genetic algorithm to find an optimal spacing distribution based on user-defined fitness. The result is a randomized spacing distribution that produced low-side lobes and high steering angles. Then, a phase modulation scheme is formulated for the Golomb and random distribution that can be applied to any non-constant distribution. All designs and algorithms are simulated with an ideal light source and Gaussian beam.

In addition, we model the OPA and other components in a lidar system and

methodologically come to certain device dimension specifics. By FDTD simulation, we also illustrate width-varying arrays that reduce coupling significantly; thus, we can bring the waveguides even closer.

Lastly, complete layouts of various proposed designs are presented on a photonic integrated chip. Although the chip has not been tested, this work laid a strong foundation for future chip testing once it was made. Preliminary experimental results of ring resonators on a similar silicon nitride-based chip at blue wavelengths proved the feasibility of the platform for these types of applications.

Overall, the findings of this work highlight the potential of the presented OPA with Golomb, random distribution, and varying waveguide widths to increase the OPA performance and robustness. Such approaches could be applied to other wavelengths, further advancing the OPA technology and its integration into lidar and ADAS systems.

# Chapter 7

## Appendix

### .1 OPA Simulation

#### Main file

```
1 %% Model of 1D OPA
2 % Name: Long Nguyen – Rochester Institute of Technology
3 % Optical Phased Array for blue wavelengths 2022 – 2023
4 %% Set up
5 clear; clc; clf
6 close all;
7 figsaveFlag = false ;
8 set(groot, 'defaultAxesFontSize',14);
9 fname = 'E:\RIT\Fall_22\Thesis\OPA-simulation\Thesis\sim2D\
    pics';
```

```

10 N = 16; wl = 0.45; d_opt = wl/2;
11 d = 2*d_opt; D = 5000*d ; % Distance
12 x = -500*floor(N/2)*d:2:500*floor(N/2)*d; y = 0:1:5000*d;
13 k = 2*pi/wl ; phi = 0;
14 [X,Y] = meshgrid(x,y);
15 %% uniform distance case:
16 for d = [d_opt , 2*d_opt , 8*d_opt]
17     N_x = linspace(-floor(N/2)*d, floor(N/2)*d, N);
18     N_y = zeros(size(N_x));
19     [I, I_furthest_norm , width_OPA] = OPA_sim(N, N_x,N_y, X,
20         Y, D, k, d, ...
21         d_opt , phi , fname , figsaveFlag);
22 %% subplot
23 I_ff_array = zeros(5,1); phi = 0;
24 for d = d_opt
25     N_x = linspace(-floor(N/2)*d, floor(N/2)*d, N);
26     N_y = zeros(size(N_x));
27     [I, I_furthest_norm , I_ff_norm , width_OPA] =
28         OPA_sim_subplot(N, N_x,N_y, X, Y, D, k, d, ...
29         d_opt , phi , fname , figsaveFlag);

```

```

30 %% subplot N
31 for N = [4 8 16 32 64]
32     d = d_opt;
33     N_x = linspace(-floor(N/2)*d, floor(N/2)*d, N);
34     N_y = zeros(size(N_x));
35     [I, I_furthest_norm, I_ff_norm, width_OPA] =
        OPA_sim_subplot(N, N_x, N_y, X, Y, D, k, d, ...
36         d_opt, phi, fname, figsaveFlag);
37 end
38 %% No plot - constant
39 X_ff = -1000*D:floor(D/100):1000*D;
40 Angle_ff = rad2deg(atan(X_ff/D));
41 figure;
42 i = 1;
43 for d = 2*d_opt
44     for N = 4
45         for phi = (1:10)*pi/5
46             N_x = linspace(-floor(N/2)*d, floor(N/2)*d, N);
47             N_y = zeros(size(N_x));
48             [I, I_furthest_norm, I_ff_norm, width_OPA] =
                OPA_no_plot(N, N_x, N_y, X, Y, D, k, d, ...
49                 d_opt, phi, fname, figsaveFlag);

```

```

50     subplot(5,2,i);
51     polarplot(Angle_ff*pi/180, I_ff_norm);
52     h_ax = gca; h_ax.ThetaDir = 'clockwise';
53     h_ax.ThetaZeroLocation = 'top';h_ax.ThetaLim =
        [-90 90];
54     h_ax.ThetaTick = -90:15:90;
55     title(sprintf("N = %i, d= %.2f um", N, d));
56     rlim([-20 max(I_ff_norm)]);
57     h_ax.LineWidth = 2.5; h_ax.FontSize = 15;
58     i = i + 1;
59     end
60     end
61 end
62 %% Gaussian
63 wg_w = 0.3; i = 1;
64 for d = [2]*d_opt
65     for N = [16]
66         figure;hold on
67         set(groot, 'defaultAxesFontSize',20);
68         for phi = [0 pi 3*pi/2]
69             D = 5000*d; N_x = linspace(-floor(N/2)*d, floor(
                N/2)*d, N);

```

```

70     N_y = zeros(size(N_x));
71     [Angle_ff, I, I_ff, I_ff_norm, width_OPA]=
        OPA_noplot_Gaussian(N, N_x, N_y, X, Y, D, k,
        wg_w, phi);
72     plot(Angle_ff, I_ff/max(I_ff,[]), "all"),
        DisplayName", sprintf("%s = %.1f %s", '\phi', phi
        /pi, '\pi'), 'LineWidth', 3);
73     end
74     i = i+1;
75     title(sprintf("Gaussian source N=16, d = %.1f %s", d
        /(2*d_opt), '\lambda'), 'FontSize', 20);
76     ylabel("Normalized power");
77     xlabel("Angle (deg)");
78     lg = legend; lg.FontSize = 22;
79     end
80 end
81 %% Periodic
82 j = 1 ;set(groot, 'defaultAxesFontSize', 10);
83 for phi = [0]
84     for j = 1 : 3
85         if j == 1
86             atomic_set = [1, 2, 1];

```

```

87     elseif j == 2
88         atomic_set = [1, 2];
89     elseif j == 3
90         atomic_set = [1, 2, 3, 4];
91     end
92     periodic_set = Period_set(N, atomic_set);
93     N_x = 2*d_opt*(periodic_set - mean(periodic_set));
94     N_y = zeros(size(N_x));
95     [I, I_furthest_norm, I_ff_norm, width_OPA, Angle_ff]
          = OPA_sim_subplot(N, N_x, N_y, X, ...
96         Y, D, k, -1, d_opt, phi, fname, figsaveFlag);
97     end
98 end
99 %% Golomb subplot
100 N=16;set(groot, 'defaultAxesFontSize',20); phi=0;
101 Golomb_distance = GolombRuler(N);
102 N_x = d_opt*(Golomb_distance - mean(Golomb_distance));
103 N_y = zeros(size(N_x));
104 [I, I_furthest_norm, I_ff_norm, width_OPA] = OPA_sim_subplot
          (N, N_x,N_y, X, Y, D, k, 3, d_opt, phi, fname, figsaveFlag
          );
105 %% Gaussian

```

```

106 Golomb_distance = GolombRuler(N);
107 wg_w = 0.3;
108 N_x = d_opt*(Golomb_distance - mean(Golomb_distance));
109 N_y = zeros(size(N_x));
110 [I, I_furthest_norm, width_OPA] = OPA_sim_Gaussian(N, N_x,
    N_y, X, Y, D, k, wg_w, 3, d_opt, phi, fname, figsaveFlag);

```

---

### Ideal source simulation

```

1 function [I, I_furthest_norm, I_ff_norm, width_OPA, Angle_ff]
    = OPA_sim_subplot(N, N_x, N_y, X, Y, D, k, d, d_opt, phi,
    fname, fsaveFlag)
2     min_pks_thres = -2;
3     width_OPA = max(N_x) - min(N_x);
4     E_x_y = zeros(size(X)); X_ff = -100*D:floor(D/500):100*D
    ;
5     Angle_ff = rad2deg(atan(X_ff/D)); E_ff = zeros(size(X_ff
    ));
6     for j = 1:N
7         E_x_y = E_x_y + exp(-1i*(k*sqrt((X-N_x(j)).^2+(Y-N_y
    (j)).^2) + j*phi))/N;

```

```

8         E_ff = E_ff + exp(-1i*(k*sqrt((X_ff-N_x(j)).^2 + D
          ^2) + j*phi))/N;
9     end
10    I = abs(E_x-y).^2;
11    I_ff = abs(E_ff).^2;
12    I_ff_norm = 10*log10(I_ff/max(I_ff));
13    I_furthest = I(end,:);
14    I_furthest_norm = 10*log10(I_furthest/max(I_furthest));
15    %% annotation cases
16    [caption, caption1, fig_name1, caption2, fig_name2,
      caption3, fig_name3, caption4, fig_name4] = annotation
      (N,0,width_OPA,d,d_opt);
17    %% plotting
18    f1 = figure('visible', not(fsaveFlag));
19    subplot(2,2,1);
20    sgtitle(caption, 'fontweight', 'bold', 'fontsize', 22);
21    hold on
22    small_angle = rad2deg(atan(X(1,:)/D)) ;
23    plot(small_angle, I_furthest_norm);
24    set(gca, 'XMinorTick', 'on', 'YMinorTick', 'on')
25    yline(max(I_furthest_norm)-3, '-', '-3dB threshold');
26    if (d ~= -1)

```

```

27         xlim([-1/d 1/d]);
28     end
29     xlabel("Angle (deg)");
30     ylabel("Normalized in intensity (dB)");
31     title(caption1);
32     hold off;
33
34     subplot(2,2,2);
35     hold on
36     plot(Angle_ff, I_ff_norm);
37     [pks1,locs1] = findpeaks(I_ff_norm, Angle_ff, '
38         MinPeakHeight', min_pks_thres);
39     plot (locs1 ,pks1, 'o', 'MarkerFaceColor', 'red', '
40         MarkerSize', 5, 'LineWidth', 2, 'DisplayName', 'Simmed
41         lobes');
42     %fprintf(" Value of d %.3f um \n", d);
43     %fprintf(" Simmed lobes [%s] \n", join(string(locs1)));
44     xlabel("Angle (deg)");
45     ylabel("Normalized in intensity (dB)");
46     title(caption3);
47     axis tight
48     hold off;

```

```

46
47 subplot(2,2,3);
48 polarplot(Angle_ff*pi/180, I_ff_norm);
49 h_ax = gca;
50 h_ax.ThetaDir = 'clockwise';
51 h_ax.ThetaZeroLocation = 'top';
52 h_ax.ThetaLim = [-90 90];
53 h_ax.ThetaTick = -90:15:90;
54 title("Polar plot for the radiation pattern");
55 rlim([min(I_ff_norm)/4 max(I_ff_norm)]);
56 h_ax.LineWidth = 2.5;
57 h_ax.FontSize = 15;
58 title(caption4)
59 hold off
60
61 subplot(2,2,4);
62 hold on
63 s = surf(X,Y,I);
64 s.EdgeColor = 'none';
65 scatter(N_x, N_y, 100, 'filled', 'red');
66 view(2);
67 axis tight;

```

```

68     axis equal;
69     colorbar
70     title(caption2, 'FontWeight', 'bold', 'Color', 'b');
71     xlabel("x (um)");
72     ylabel("Distance (um)");
73
74     if fsaveFlag
75         saveas(f1, fullfile(fname, fig_name1), 'png');
76     end
77 end

```

---

### Gaussian source simulation

```

1 function [Angle_ff, I, I_ff, I_ff_norm, width_OPA] =
    OPA_noplot_Gaussian(N, N_x, N_y, X, Y, D, k, wg_w, phi)
2     min_pks_thres = -2;
3     width_OPA = max(N_x) - min(N_x);
4     E_x_y = zeros(size(X));
5     X_ff = -5*D:floor(D/1000):5*D;
6     Angle_ff = rad2deg(atan(X_ff/D));
7     E_ff = zeros(size(X_ff));
8     n = 1; wl = 0.45;

```

```

9     if length(wg_w) == 1
10         wg_w = wg_w*ones(N,1);
11     end
12     for j = 1:N
13         w0 = wg_w(j)/2; zR = pi*w0^2*n/wl;
14         y_ref = 0.001 + N_y(j); x_ref = N_x(j);
15         z = Y - y_ref; r = X - x_ref;
16         wz = w0.*sqrt(1+(z/zR).^2);
17         Rz = z.*(1+(zR./z).^2);
18         phiZ = atan(z./zR);
19         E_x_y = E_x_y + w0./wz .* exp(-r.^2./wz.^2) .* exp
20             (-1i.*(k.*z + ...
21                 k.* r.^2 ./ 2.*Rz - phiZ + j*phi));
22         z = D - y_ref ; r = X_ff - x_ref;
23         w0 = wg_w(j)/2; zR = pi*w0^2*n/wl;
24         wz = w0.*sqrt(1+(z/zR).^2);
25         Rz = z.*(1+(zR/z).^2);
26         phiZ = atan(z/zR);
27         E_ff = E_ff + w0/wz .* exp(-r.^2./wz.^2) .* exp(-1i
28             .* (k*z + ...
                k.* r.^2 ./ 2.*Rz - phiZ - j*phi));
29     end

```

```

29     I = abs(E_x-y).^2;
30     I_ff = abs(E_ff).^2;
31     I_ff_norm = 10*log10(I_ff/max(I_ff));
32 end

```

---

### Period set, aperiodic set, and Golomb Generator

```

1 function N_x= Aperiodic_set(N)
2     N_x = randi([ floor(N/2) 2*N], 1, N);
3     for i = 1:length(N_x)-1
4         N_x(i) = sum(N_x(i+1:end), 'all');
5     end
6     N_x(end) = 0 ;
7 end

```

```

1 function N_x = Period_set(N, atomic_set)
2     N_x = repmat(atomic_set,1,ceil(N/length(atomic_set)));
3     N_x = N_x(1:N);
4     for i = 1:length(N_x)-1
5         N_x(i) = sum(N_x(i+1:end), 'all');
6     end
7     N_x(end) = 0 ;
8 end

```

```

1 function N_x = GolombRuler(N)
2     if N == 2
3         N_x = [0, 1];
4     elseif N == 3
5         N_x = [0, 1, 3];
6     elseif N == 4
7         N_x = [0, 1, 4, 6];
8     elseif N == 5
9         N_x = [0, 1, 4, 9, 11];
10    elseif N == 6
11        N_x = [0, 1, 4, 10, 12, 17];
12    elseif N == 7
13        N_x = [0, 1, 4, 10, 18, 23, 25];
14    elseif N == 8
15        N_x = [0, 1, 4, 9, 15, 22, 32, 34];
16    elseif N == 9
17        N_x = [0, 1, 5, 12, 25, 27, 35, 41, 44];
18    elseif N == 10
19        N_x = [0, 1, 6, 10, 23, 26, 34, 41, 53, 55];
20    elseif N == 11
21        N_x = [0, 1, 4, 13, 28, 33, 47, 54, 64, 70, 72];
22    elseif N == 12

```

```

23     N_x = [0, 2, 6, 24, 29, 40, 43, 55, 68, 75, 76, 85];
24     elseif N == 13
25         N_x = [0, 2, 5, 25, 37, 43, 59, 70, 85, 89, 98, 99,
26             106];
27     elseif N == 14
28         N_x = [0, 4, 6, 20, 35, 52, 59, 77, 78, 86, 89, 99,
29             122, 127];
30     elseif N == 15
31         N_x = [0, 4, 20, 30, 57, 59, 62, 76, 100, 111, 123,
32             136, 144, 145, 151];
33     elseif N == 16
34         N_x = [0, 1, 4, 11, 26, 32, 56, 68, 76, 115, 117,
35             134, 150, 163, 168, 177];
36     elseif N == 17
37         N_x = [0, 5, 7, 17, 52, 56, 67, 80, 81, 100, 122,
38             138, 159, 165, 168, 191, 199];
39     elseif N == 18
40         N_x = [0, 2, 10, 22, 53, 56, 82, 83, 89, 98, 130,
41             148, 153, 167, 188, 192, 205, 216];
42     elseif N == 19
43         N_x = [0, 1, 6, 25, 32, 72, 100, 108, 120, 130, 153,
44             169, 187, 190, 204, 231, 233, 242, 246];

```

```

38     elseif N == 20
39         N_x = [0, 1, 8, 11, 68, 77, 94, 116, 121, 156, 158,
40             179, 194, 208, 212, 228, 240, 253, 259, 283];
41     elseif N == 21
42         N_x = [0, 2, 24, 56, 77, 82, 83, 95, 129, 144, 179,
43             186, 195, 255, 265, 285, 293, 296, 310, 329, 333];
44     elseif N == 22
45         N_x = [0, 1, 9, 14, 43, 70, 106, 122, 124, 128, 159,
46             179, 204, 223, 253, 263, 270, 291, 330, 341, 353,
47             356];
48     elseif N == 23
49         N_x = [0, 3, 7, 17, 61, 66, 91, 99, 114, 159, 171,
50             199, 200, 226, 235, 246, 277, 316, 329, 348, 350,
51             355, 372];
52     end
53 end

```

## .2 Genetic Algorithm

### GA main

```

1 %% Implement Genetic Algorithm
2 tic

```

```

3 %% Set up simulation & genetic algorithm
4 clear; clc; clf; close all
5 figsaveFlag = true ;
6 fname = 'E:\RIT\Fall_22\Thesis\OPA-simulation\Genetic
      Algorithm';
7 N = 16; wl = 0.45; d_opt = wl/2; d = 2*d_opt;
8 wg_w = 0.3;
9 k = 2*pi/wl ; phi = 0;
10 x = -200*ceil(N/2)*d : 200*ceil(N/2)*d; % framework for
      N_x
11 locmid = 2501;
12 lastgen = 1000;
13 totalPopulation = 50;
14 mutationRate = 0.02;
15 crossoverRate = 0.1;
16 population = zeros(N, totalPopulation);
17 fitness = zeros(1, totalPopulation);
18 onepeak = ones(1, totalPopulation);
19 Emax1 = zeros(1, totalPopulation);
20 mating_idx = zeros(1, totalPopulation);
21 % index is the index of population, value is background
22 % init 1s because we want to find min and 1 is the max.

```

```

23 %% Generate population
24 i = 1;
25 d = (10:2:40)*wl;
26 while (i < totalPopulation + 1)
27     aperiodic_set = Aperiodic_set(N);
28     N_x = 2*d_opt*(aperiodic_set - mean(aperiodic_set));
29     N_x = interp1(x,x, N_x, 'nearest');
30     population(:,i) = N_x;
31     i=i+1;
32 end
33 i = 1;
34 while (i < totalPopulation + 1)
35     j = randi(totalPopulation); k = randi(totalPopulation);
36     parentA = population(:,j); parentB = population(:,k);
37     child = crossover(parentA, parentB, crossoverRate);
38     child = mutate(child, 0.5, wl);
39     population(:,i) = child;
40     i = i + 1;
41 end
42
43 %% Calculate Fitness
44 generation = 1;

```

```

45 bestEmax = zeros(0,lastgen);
46 bestfit = zeros(0, lastgen);
47 bestPeakDiff = zeros(1,lastgen);
48 ftextsave = fname + "\text";
49 fsum = sprintf("Log-GA-%i.txt", randi([0 100]));
50 fileID = fopen(fullfile(ftextsave, fsum), 'wt');
51 ftextgood = "E:\RIT\Fall_22\Thesis\OPA-simulation\Genetic
      Algorithm\shift_check\text";
52 fsumgood = sprintf("Log-GA-N-%i-lastgen-%i-%i.txt", N,
      lastgen, randi([0 100]));
53 fID_good = fopen(fullfile(ftextgood, fsumgood), 'wt');
54 fprintf(fileID, "Genetic Algorithm... \n \n");
55 fprintf(fileID, "Saving distribution with best FOV \n");
56 fprintf(fileID, "Operating wavelength %0.2f um \n", 2*d_opt)
      ;
57 D = 5000*N*40*d_opt ;
58 %% loop
59 while (generation < lastgen + 1 )
60     i = 1;
61     while (i< totalPopulation+1)
62         N_x = population(:, i);

```

```

63     [Emid1, I_ff , Angle_ff , peaksdiff] = OPA_farfield_v4
        (N, N_x, D, k, phi, locmid);
64     Emax1(i) = Emid1;
65     fitness(i) = Emid1 + peaksdiff + rand(1)/100; %
        adding to avoid equal max
66     bestPeakDiff(i)=peaksdiff;
67     i = i+1;
68     end
69 %% Selecting the best
70     [fitmax , idx] = max(fitness);
71     bestfit(generation) = fitmax;
72     peakdiffmax = bestPeakDiff(idx);
73     bestEmax(generation) = Emax1(idx);
74     bestNx = population(:,idx);
75     avgdis = (max(bestNx)-min(bestNx))/N;
76 %% Writing to a file , printing , debugging , shift checking
77     if (peakdiffmax > 4)
78         set(groot , 'defaultAxesFontSize' ,14);
79         [I_ff_norm0 , Angle_ff0] = OPA_farfield(N, bestNx , D,
            k, 0);
80         f1 = figure ( 'visible' , 'off' );
81         plot(Angle_ff0 , I_ff_norm0 , 'LineWidth' ,2.5);

```

```

82     axis tight
83     xticks([-75 -60 -45 -30 -15 0 15 30 45 60 75]);
84     xlabel("Angle (deg)");
85     ylabel("Normalized Intensity");
86     title(sprintf("N %i, gen %i", N, generation));
87     fplotsave = fname + "\shift_check";
88     fig_name1 = sprintf("No_shift_check_(%i)",randi(100)
89         );
89     saveas(f1, fullfile(fplotsave,fig_name1), 'png');
90     fprintf(fID_good, "Emax of generation %i: %.2f \n",
91         generation ,Emax(idx));
91     fmt=['Nx (um) =' repmat(' %.2f ',1,numel(bestNx))];
92     fprintf(fID_good, fmt, bestNx);
93     fprintf(fID_good, " ##### \n");
94 end
95 generation = generation +1;
96 %% Create mating pool, crossover, mutation → new
97     population
97 % Choose mating pool based on probability (FOV value)
98     from previous
98 % population

```

```

99         % higher FOV distribution will appear more in the
          mating pool
100    % randomly choose in mating pool (probability of
          previous population
101    % is embedded)
102    % Then crossover between parents —> child
103    % then DNA of child will mutate (with rate 0.1)
104    % to get new population
105    fitness_prob = fitness/sum(fitness , 'all');
106    [fitness_sort , sortidx] = sort(fitness_prob , 'descend');
107    mating_idx(1:30) = sortidx(1);
108    mating_idx(31:55) = sortidx(2);
109    mating_idx(56:75) = sortidx(3);
110    mating_idx(76:90) = sortidx(4);
111    mating_idx(91:100) = sortidx(5);
112    % return the index of population
113    % from previous population based on its fitness
114    matingPool = population(:, mating_idx);
115    l = 1 ;
116    % choose new population from mating pool
117    while (l < totalPopulation+1)
118        j = randi(100); k = randi(100);

```

```

119     parentA = matingPool(:,j); parentB = matingPool(:,k)
        ;
120     child = crossover(parentA, parentB, crossoverRate);
121     child = mutate(child, mutationRate, wl);
122     if length(child) == length(unique(round(child,2)))
123         l = l + 1;
124         population(:,l) = child;
125     end
126 end
127
128 end
129 %% Plotting results
130 a = randi([0 100]);
131 set(groot, 'defaultAxesFontSize', 14);
132 fplotsave = fname + "\plots";
133 f1 = figure('visible', (figsaveFlag));
134 fig_name1 = sprintf("GA_N%i_gen_%i_(%i)", N, lastgen, randi
        (100));
135 x = 1:1:lastgen;
136 bestEmax_movingmean = movmean(bestEmax, 5);
137 plot(x, bestEmax_movingmean, 'LineWidth', 2.5);
138 xlabel("Generations");

```

```

139 axis tight;
140 ylabel("fitness");
141 title("Genetic Algorithm");
142 if figsaveFlag
143     saveas(f1, fullfile(fplotsave, fig_name1), 'png');
144 end
145 toc

```

---

### Far field calculation

```

1 function [Etarget, I_ff, Angle_ff, peaksdiff] =
    OPA_farfield_v4(N, N_x, D, k, phi, locmid)
2     locmid = 2501;
3     X_ff = -5*D:floor(D/500):5*D;
4     Angle_ff = rad2deg(atan(X_ff/D));
5     E_ff = zeros(size(X_ff));
6     for j = 1:N
7         E_ff = E_ff + exp(-1i*(k*sqrt((X_ff-N_x(j)).^2 + D
            ^2) + j*phi))/N;
8     end
9     I_ff = abs(E_ff).^2;

```

```

10     Etarget = (I_ff(locmid-2)+I_ff(locmid-1)+I_ff(locmid)+
               I_ff(locmid+1)+I_ff(locmid+2))/5; % E when x=0
11     [pks1, locs1] = findpeaks(I_ff, "SortStr","descend");
12     if length(pks1)>1
13         peaksdiff = pks1(1)/pks1(2);
14     else
15         peaksdiff = 10;
16     end
17 end

```

---

### mutate and crossover function

```

1 function child = crossover(parentA, parentB, crossoverRate)
2     child = zeros(1, length(parentA));
3     if (rand(1) < crossoverRate)
4         midpoint = randi(length(parentA));
5         i = 1;
6         while (i < length(parentA)+1)
7             if (i > midpoint)
8                 child(i) = parentA(i);
9             else
10                child(i) = parentB(i);

```

```

11         end
12         i = i + 1;
13     end
14 else
15     if (rand(1) < 0.5)
16         child = parentA;
17     else
18         child = parentB;
19     end
20 end
21 end

1 function mutant = mutate(child , mutationRate , wl)
2     i = 1;
3     mutant = child;
4     while (i < length(child)+1)
5         if (rand(1) < mutationRate)
6             mutant(i) = mutant(i) + wl*randi([-1 1]);
7         end
8         i = i + 1;
9     end
10
11 end

```

### .3 Phase Modulation Scheme

#### Main

```
1 %% Set up
2 clear; clc; clf; close all
3 figsaveFlag = false ;
4 set(groot, 'defaultAxesFontSize', 16);
5 fname = 'E:\RIT\Fall_22\Thesis\OPA-simulation\
        Phase_Distribution_Scheme';
6 N = 16; wl = 0.45; d_opt = wl/2;
7 k = 2*pi/wl; array_unit = 1:N;
8 N_x = linspace(-floor(N/2)*d_opt, floor(N/2)*d_opt, N);
9 D = 100000*N*d_opt ; wg-w = 0.3;
10 phi_val = linspace(0, 3*pi/2, 6); phiN = length(phi_val);
11 ang_peak = zeros(1, phiN); peaksdiff = zeros(1, phiN);
12 f1 = figure; subplot(3,1,1);
13 hold on
14 %% uniform distance case:
15 for d = [d_opt]
16     i = 1;
17     while i < (phiN+1)
18         [Emax, I_ff1, Angle_ff, Ang_peak1, peaksdiff1] =
                OPA_farfield_phase_scheme(N, N_x, D, k, phi_val(i))
```

```

        *array_unit);
19     ang_peak(i) = Ang_peak1; peaksdiff(i) = peaksdiff1;
20     legendname = sprintf("Shift %.1f %s", phi_val(i)/pi
        , '\pi ');
21     plot(Angle_ff, I_ff1, "DisplayName", legendname);
22     i = i + 1;
23     end
24 end
25 xlabel("Angle (deg)"); ylabel("Normalized power");
26 a = sprintf("Uniform spacing %s/2, N %i", '\lambda', N);
27 axis tight; title (a); legend;
28 hold off; N_x_opt = N_x;
29 %% Golomb case
30 Golomb_distance = GolombRuler(N);
31 N_x = d_opt*(Golomb_distance - mean(Golomb_distance));
32 diff_Nx = N_x_opt - N_x;
33 diff_phi = k*diff_Nx; i = 1;
34 subplot(3,1,2); hold on;
35 while i < (phiN+1)
36     phicustom = -diff_phi*sind(ang_peak(i)) + phi_val(i)*
        array_unit;

```

```

37     [Emax, I_ff1 , Angle_ff , Ang_peak1 , peaksdiff1] =
        OPA_farfield_phase_scheme(N, N_x, D, k, phicustom);
38     ang_peak(i) = Ang_peak1; peaksdiff(i) = peaksdiff1;
39     legendname = sprintf("Equi Shift %.1f %s", phi_val(i)/pi
        , '\pi ');
40     plot (Angle_ff , I_ff1 , "DisplayName",legendname);
41     i = i + 1;
42 end
43 xlabel("Angle (deg)"); ylabel("Normalized power");
44 a = sprintf("Goloumb spacing , N %i" , N);
45 axis tight; title (a); legend;
46 hold off;
47 %% Random Distribution
48 N_x= [18.45 16.20 13.95 11.25 9.00 5.40 3.15 1.80 -1.35
        -2.25 -4.05 -9.45 -12.15 -15.30 -16.20 -19.35];
49 i = 1;
50 subplot(3,1,3); hold on;
51 diff_Nx = N_x_opt - N_x;
52 diff_phi = k*diff_Nx;
53 while i < (phiN+1)
54     phicustom = -diff_phi*sind(ang_peak(i)) + phi_val(i)*
        array_unit; %% phase scheme

```

```

55     [Emax, I_ff1 , Angle_ff , Ang-peak1 , peaksdiff1] =
        OPA_farfield_phase_scheme(N, N_x, D, k, phicustom);
56     %ang_peak(i) = Ang-peak1; peaksdiff(i) = peaksdiff1;
57     legendname = sprintf("Equi Shift %.1f %s", phi_val(i)/pi
        , '\pi ');
58     plot (Angle_ff , I_ff1 , "DisplayName",legendname);
59     i = i + 1;
60 end
61 xlabel("Angle (deg)"); ylabel("Normalized power");
62 a = sprintf("Genetic Algorithm N %i", N);
63 axis tight; title (a); legend;

```

---

### Ideal Source

```

1 function [Etarget , I_ff , Angle_ff , Ang-peak , peaksdiff] =
        OPA_farfield_phase_scheme(N, N_x, D, k, phi)
2 %% Output:
3 % 2D Intensity
4 % far field I vs x
5 % determining grating lobes
6 locmid = 2501;
7 X_ff = -10*D:D/500:10*D;

```

```

8     Angle_ff = rad2deg(atan(X_ff/D));
9     E_ff = zeros(size(X_ff));
10    for j = 1:N
11        E_ff = E_ff + exp(-1i*(k*sqrt((X_ff-N_x(j)).^2 + D
            ^2) + phi(j)))/N;
12    end
13    I_ff = abs(E_ff).^2;
14    Etarget = I_ff(locmid);
15    [pks1, locs1] = findpeaks(I_ff, Angle_ff, "SortStr", "
            descend");
16    Ang_peak = locs1(1);
17    peaksdiff = pks1(1)/pks1(2);
18 end

```

---

### Gaussian Source

```

1 function [Emax, I_ff, Angle_ff, Ang_peak, peaksdiff, phasewg
    ] = OPA_ff_Gaussian_v5(N, N_x, D, k, wg_w, phi)
2     phasewg = zeros(1,N);
3     locmid = 2501;
4     X_ff = -5*D:D/500:5*D;
5     Angle_ff = rad2deg(atan(X_ff/D));

```

```

6     E_ff = zeros(size(X_ff));
7     n = 1; wl = 0.45;
8     if length(wg_w) == 1
9         wg_w = wg_w*ones(N,1);
10    end
11    if length(phi) == 1
12        phi = phi*ones(1,N);
13    end
14    for j = 1:N
15        y_ref = 0; x_ref = N_x(j);
16        z = D - y_ref ; r = X_ff - x_ref;
17        w0 = wg_w(j)/2; zR = pi*w0^2*n/wl;
18        wz = w0*sqrt(1+(z/zR)^2);
19        Rz = z*(1+(zR/z)^2);
20        phiZ = atan(z/zR);
21        E_ff = E_ff + w0/wz.*exp(-r.^2/(wz^2)).* exp(-1i
                .* (k*z + ...
22                    k.* r.^2/(2*Rz) - phiZ + phi(j)));
23    end
24    I_ff = abs(E_ff).^2;
25    Emax = I_ff(locmid); % E when x=0
26    [pks1, locs1] = findpeaks(I_ff, Angle_ff, "SortStr",)

```

```
        descend");  
27     Ang_peak      = locs1(1);  
28     peaksdiff    = pks1(1)/pks1(2);  
29 end
```

# Bibliography

- [1] T. Kim, *Realization of Integrated Coherent LiDAR*. PhD thesis, UC Berkeley, 2019.
- [2] T. Hegna, H. Pettersson, and K. Grujic, “Inexpensive 3-D laser scanner system based on a galvanometer scan head,” *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, vol. 38, pp. 277–281, Jan. 2010.
- [3] K. Li, C. Chase, P. Qiao, and C. J. Chang-Hasnain, “Widely tunable 1060-nm VCSEL with high-contrast grating mirror,” *Optics Express*, vol. 25, pp. 11844–11854, May 2017. Publisher: Optica Publishing Group.
- [4] Q. Wang, W. Wang, X. Zhuang, C. Zhou, and B. Fan, “Development of an Electrostatic Comb-Driven MEMS Scanning Mirror for Two-Dimensional Raster Scanning,” *Micromachines*, vol. 12, p. 378, Apr. 2021. Number: 4 Publisher: Multidisciplinary Digital Publishing Institute.

- [5] J. Sun, E. Timurdogan, A. Yaacobi, E. S. Hosseini, and M. R. Watts, “Large-scale nanophotonic phased array,” *Nature*, vol. 493, pp. 195–199, Jan. 2013. Number: 7431 Publisher: Nature Publishing Group.
- [6] M. C. Shin, A. Mohanty, K. Watson, G. R. Bhatt, C. T. Phare, S. A. Miller, M. Zadka, B. S. Lee, X. Ji, I. Datta, and M. Lipson, “Chip-scale blue light phased array,” *Optics Letters*, vol. 45, pp. 1934–1937, Apr. 2020. Publisher: Optica Publishing Group.
- [7] D. N. Hutchison, J. Sun, J. K. Doylend, R. Kumar, J. Heck, W. Kim, C. T. Phare, A. Feshali, and H. Rong, “High-resolution aliasing-free optical beam steering,” *Optica*, vol. 3, pp. 887–890, Aug. 2016. Publisher: Optica Publishing Group.
- [8] “Constructive and destructive interference.”
- [9] S. Zhao, J. Chen, and Y. Shi, “All-Solid-State Beam Steering via Integrated Optical Phased Array Technology,” *Micromachines*, vol. 13, p. 894, June 2022. Number: 6 Publisher: Multidisciplinary Digital Publishing Institute.
- [10] G. M. Djuknic, “Method of measuring a pattern of electromagnetic radiation,” Dec. 2003.
- [11] R. J. Mailloux, *Phased Array Antenna Handbook, Third Edition*. Artech House, Nov. 2017. Google-Books-ID: EKN5DwAAQBAJ.

- [12] S. T. S. Holmström, U. Baran, and H. Urey, “MEMS Laser Scanners: A Review,” *Journal of Microelectromechanical Systems*, vol. 23, pp. 259–275, Apr. 2014. Conference Name: Journal of Microelectromechanical Systems.
- [13] J. Wojtanowski, M. Zygmunt, M. Kaszczuk, Z. Mierczyk, and M. Muzal, “Comparison of 905 nm and 1550 nm semiconductor laser rangefinders’ performance deterioration due to adverse environmental conditions,” *Opto-Electronics Review*, vol. 22, Sept. 2014.
- [14] Y. Tamura, H. Sakuma, K. Morita, M. Suzuki, Y. Yamamoto, K. Shimada, Y. Honma, K. Sohma, T. Fujii, and T. Hasegawa, “The First 0.14-dB/km Loss Optical Fiber and its Impact on Submarine Transmission,” *Journal of Lightwave Technology*, vol. 36, pp. 44–49, Jan. 2018.
- [15] C. Yu, M. Shangguan, H. Xia, J. Zhang, X. Dou, and J.-W. Pan, “Fully integrated free-running InGaAs/InP single-photon detector for accurate lidar applications,” *Optics Express*, vol. 25, pp. 14611–14620, June 2017. Publisher: Optica Publishing Group.
- [16] B.-W. Yoo, M. Megens, T. Sun, W. Yang, C. J. Chang-Hasnain, D. A. Horsley, and M. C. Wu, “A  $32 \times 32$  optical phased array using polysilicon sub-wavelength high-contrast-grating mirrors,” *Optics Express*, vol. 22, pp. 19029–19039, Aug. 2014. Publisher: Optica Publishing Group.
- [17] S.-J. Lee, J. Lim, S. Moon, J. Lee, K. Kim, Y. Park, and J.-H. Lee, “MEMS Scanner-Based Biaxial LiDAR System for Direct Detection of Three-

Dimensional Images,” in *2018 International Conference on Optical MEMS and Nanophotonics (OMN)*, pp. 1–5, July 2018.

- [18] M. Hochberg, N. C. Harris, R. Ding, Y. Zhang, A. Novack, Z. Xuan, and T. Baehr-Jones, “Silicon Photonics: The Next Fabless Semiconductor Industry,” *IEEE Solid-State Circuits Magazine*, vol. 5, no. 1, pp. 48–58, 2013. Conference Name: IEEE Solid-State Circuits Magazine.
- [19] J. M. Senior and M. Y. Jamro, *Optical fiber communications: principles and practice*. Harlow, England ; New York: Financial Times/Prentice Hall, 3rd ed ed., 2009. OCLC: ocn226304746.
- [20] C. Brackett, “Dense wavelength division multiplexing networks: principles and applications,” *IEEE Journal on Selected Areas in Communications*, vol. 8, pp. 948–964, Aug. 1990. Conference Name: IEEE Journal on Selected Areas in Communications.
- [21] M. Li, W. H. P. Pernice, C. Xiong, T. Baehr-Jones, M. Hochberg, and H. X. Tang, “Harnessing optical forces in integrated photonic circuits,” *Nature*, vol. 456, pp. 480–484, Nov. 2008. Number: 7221 Publisher: Nature Publishing Group.
- [22] J. Hu, X. Sun, A. Agarwal, and L. C. Kimerling, “Design guidelines for optical resonator biochemical sensors,” *JOSA B*, vol. 26, pp. 1032–1041, May 2009. Publisher: Optica Publishing Group.

- [23] M. Burla, L. R. Cortés, M. Li, X. Wang, L. Chrostowski, and J. Azaña, “Integrated waveguide Bragg gratings for microwave photonics signal processing,” *Optics Express*, vol. 21, pp. 25120–25147, Oct. 2013. Publisher: Optica Publishing Group.
- [24] C. Sun, M. T. Wade, Y. Lee, J. S. Orcutt, L. Alloatti, M. S. Georgas, A. S. Waterman, J. M. Shainline, R. R. Avizienis, S. Lin, B. R. Moss, R. Kumar, F. Pavanello, A. H. Atabaki, H. M. Cook, A. J. Ou, J. C. Leu, Y.-H. Chen, K. Asanović, R. J. Ram, M. A. Popović, and V. M. Stojanović, “Single-chip microprocessor that communicates directly using light,” *Nature*, vol. 528, pp. 534–538, Dec. 2015. Number: 7583 Publisher: Nature Publishing Group.
- [25] C. V. Poulton, M. J. Byrd, M. Raval, Z. Su, N. Li, E. Timurdogan, D. Coolbaugh, D. Vermeulen, and M. R. Watts, “Large-scale silicon nitride nanophotonic phased arrays at infrared and visible wavelengths,” *Optics Letters*, vol. 42, pp. 21–24, Jan. 2017. Publisher: Optica Publishing Group.
- [26] C. V. Poulton, M. J. Byrd, P. Russo, E. Timurdogan, M. Khandaker, D. Vermeulen, and M. R. Watts, “Long-Range LiDAR and Free-Space Data Communication With High-Performance Optical Phased Arrays,” *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 25, pp. 1–8, Sept. 2019. Conference Name: IEEE Journal of Selected Topics in Quantum Electronics.
- [27] J. K. Doylend, M. J. R. Heck, J. T. Bovington, J. D. Peters, L. A. Coldren, and J. E. Bowers, “Two-dimensional free-space beam steering with an optical

- phased array on silicon-on-insulator,” *Optics Express*, vol. 19, pp. 21595–21604, Oct. 2011. Publisher: Optica Publishing Group.
- [28] T. Baba, T. Tamanuki, H. Ito, M. Kamata, R. Tetsuya, S. Suyama, H. Abe, and R. Kurahashi, “Silicon Photonics FMCW LiDAR Chip With a Slow-Light Grating Beam Scanner,” *IEEE Journal of Selected Topics in Quantum Electronics*, vol. 28, pp. 1–8, Sept. 2022. Conference Name: IEEE Journal of Selected Topics in Quantum Electronics.
- [29] H. J. Visser, “Array and Phased Array Antenna Basics | Wiley.”
- [30] R. C. Hansen, “Phased Array Antennas, 2nd Edition | Wiley.”
- [31] C. A. Balanis, “Antenna Theory: Analysis and Design, 4th Edition | Wiley.”
- [32] R. Mailloux, “Phased array theory and technology,” *Proceedings of the IEEE*, vol. 70, pp. 246–291, Mar. 1982. Conference Name: Proceedings of the IEEE.
- [33] K. Van Acoleyen, *Nanophotonic beamsteering elements using silicon technology for wireless optical applications*. dissertation, Ghent University, 2012. ISBN: 9789085785255.
- [34] M. Beck, T. Bogart, and T. Pham, “Enumeration of Golomb Rulers and Acyclic Orientations of Mixed Graphs,” Oct. 2011. arXiv:1110.6154 [math].
- [35] J. P. Robinson, “Optimum Golomb Rulers,” *IEEE Transactions on Computers*, vol. 28, pp. 943–944, Dec. 1979. Publisher: IEEE Computer Society.

- [36] L. Poladian, S. Manos, and B. Ashton, “Multi-objective and constrained design of gratings using genetic algorithms,” in *2005 Pacific Rim Conference on Lasers & Electro-Optics*, pp. 950–952, July 2005.
- [37] A. Gondarenko, S. Preble, J. Robinson, L. Chen, H. Lipson, and M. Lipson, “Spontaneous Emergence of Periodic Patterns in a Biologically Inspired Simulation of Photonic Structures,” *Physical Review Letters*, vol. 96, p. 143904, Apr. 2006. Publisher: American Physical Society.
- [38] M. Mitchell, *An introduction to genetic algorithms*, vol. 34. The MIT Press, Sept. 1996. ISSN: 0009-4978, 1523-8253 Issue: 01 Journal Abbreviation: Choice Reviews Online.
- [39] S. Preble, “DYNAMIC SILICON NANOPHOTONIC DEVICES,” June 2007. Accepted: 2007-06-05T11:23:45Z.
- [40] C. Schinke, P. Christian Peest, J. Schmidt, R. Brendel, K. Bothe, M. R. Vogt, I. Kröger, S. Winter, A. Schirmacher, S. Lim, H. T. Nguyen, and D. MacDonald, “Uncertainty analysis for the coefficient of band-to-band absorption of crystalline silicon,” *AIP Advances*, vol. 5, p. 067168, June 2015. Publisher: American Institute of Physics.
- [41] L. V. R.-d. Marcos, J. I. Larruquert, J. A. Méndez, and J. A. Aznárez, “Self-consistent optical constants of  $\text{SiO}_2$  and  $\text{Ta}_2\text{O}_5$  films,” *Optical Materials Express*, vol. 6, pp. 3622–3637, Nov. 2016. Publisher: Optica Publishing Group.

- [42] S. Romero-García, F. Merget, F. Zhong, H. Finkelstein, and J. Witzens, “Silicon nitride CMOS-compatible platform for integrated photonics applications at visible wavelengths,” *Optics Express*, vol. 21, pp. 14036–14046, June 2013.
- [43] R. Baets, A. Z. Subramanian, S. Clemmen, B. Kuyken, P. Bienstman, N. Le Thomas, G. Roelkens, D. Van Thourhout, P. Helin, and S. Severi, “Silicon Photonics: silicon nitride versus silicon-on-insulator,” in *Optical Fiber Communication Conference*, (Anaheim, California), p. Th3J.1, OSA, 2016.
- [44] M. Hammer and O. V. Ivanova, “Effective index approximations of photonic crystal slabs: a 2-to-1-D assessment,” *Optical and Quantum Electronics*, vol. 41, pp. 267–283, Mar. 2009.
- [45] A. W. Snyder and J. D. Love, *Optical waveguide theory*. London: Chapman and Hall, 1983. OCLC: 911300756.
- [46] H. Gehring, M. Blaicher, W. Hartmann, and W. H. P. Pernice, “Python based open source design framework for integrated nanophotonic and superconducting circuitry with 2D-3D-hybrid integration,” *OSA Continuum*, vol. 2, pp. 3091–3101, Nov. 2019. Publisher: Optica Publishing Group.
- [47] K. Hakhamaneshi, N. Werblun, P. Abbeel, and V. Stojanovic, “BagNet: Berkeley Analog Generator with Layout Optimizer Boosted with Deep Neural Networks,” July 2019. arXiv:1907.10515 [cs, eess].

- [48] L. Chrostowski and M. Hochberg, *Silicon Photonics Design: From Devices to Systems*. Cambridge: Cambridge University Press, 2015.