

UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

CARRERA DE CIENCIA DE LA COMPUTACIÓN



**Estudio de Cinco Algoritmos de Optimización usando
HPC para Fabricar Dos Dispositivos Nanofotónicos:
*bend-90° y WDM***

TESIS

Para optar el título profesional de Licenciado en Ciencia de la
Computación

AUTOR:

José Leonidas García Gonzales

ASESOR

Jorge Luis Gonzalez Reaño

Lima - Perú

11 de diciembre de 2021

Índice general

	Pág.
CAPÍTULO 1 Motivación y Contexto	1
1.1 Introducción	3
1.2 Descripción del Problema	5
1.3 Justificación	7
1.4 Objetivos	7
1.5 Aportes	8
CAPÍTULO 2 Marco Teórico	9
2.1 Dispositivos de estudio	10
2.1.1 <i>Bend</i>	10
2.1.2 <i>Wavelength Demultiplexer</i> de dos canales (WDM)	11
2.2 Diseño inverso	12
2.2.1 Parametrización	12
2.2.1.1 Parametrización basada en topología	12
2.2.1.2 Parametrización por conjuntos de nivel	13
2.2.1.3 Parametrización por segmentos	14
2.2.2 Definición de la función objetivo	15
2.2.3 Simulación	16
2.2.4 Estrategias de optimización	16
2.2.5 Transformaciones	17

2.3 Preparación para fabricación	19
CAPÍTULO 3 Trabajos relacionados	20
3.1 Su et al. (2020)	20
3.2 Piggott et al. (2017)	21
3.3 Prosopio-Galarza et al. (2019)	21
3.4 Hammond et al. (2020)	22
3.5 Elsawy et al. (2020) , Gregory et al. (2015) , Schneider et al. (2019)	22
CAPÍTULO 4 Propuesta	24
4.0.1 Parametrización	24
4.0.1.1 <i>Bend</i>	24
4.0.1.2 WDM	25
4.0.2 Función objetivo	26
4.0.3 Optimización	26
4.0.3.1 Optimización continua	28
4.0.3.2 Optimización discreta	28
4.0.3.3 Optimización de fabricación	29
4.0.4 Preparación para fabricación	29
4.0.5 Fabricación	30
CAPÍTULO 5 Resultados Preliminares	31
5.1 Experimento 1	31
5.2 Experimento 2	33
5.3 Experimento 3	36

Índice de tablas

2.1	Evaluación cualitativa de las librerías MEEP y SPINS	16
-----	----------------------------------------------------------------	----

Índice de figuras

1.1	Diseños tradicionales y obtenidos a partir de diseño inverso de un <i>bend</i> y un WDM	5
1.2	<i>Bend</i> con una región de diseño discretizada en 18×18 píxeles. Cada píxel negro representa la presencia de <i>Si</i> y cada píxel blanco de <i>SiO₂</i>	6
2.1	Intensidad de campo eléctrico para un <i>bend-90°</i> de radio interno de $0.25 \mu m$	11
2.2	Parametrización por píxeles para un <i>bend-90°</i>	13
2.3	Parametrización por conjuntos de nivel para un <i>bend-90°</i>	14
2.4	Función de discretización con $\eta = 0.5$ y distintos valores de β	18
4.1	Metodología del trabajo de investigación	25
4.2	Dimensiones del diseño del <i>bend</i> a utilizar.	26
4.3	Dimensiones del diseño del WDM a utilizar.	27
5.1	Gráfica del número de simulaciones vs transmitancia producto de optimizar un <i>bend</i> usando CMA-ES en el experimento 1.	31
5.2	Intensidad de campo eléctrico y transmitancia del diseño obtenidos en el experimento 1 simulado bajo una resolución de $40 nm$. La línea blanca ubicada en la parte inferior derecha indica la escala de $1 \mu m$ en la figura.	32
5.3	Gráfica del número de simulaciones vs transmitancia producto de optimizar un <i>bend</i> usando CMA-ES en el experimento 2.	33
5.4	Intensidad de campo eléctrico y transmitancia del diseño obtenidos en el experimento 2 simulado bajo una resolución de $40 nm$	34
5.5	Distribución de la permitividad eléctrica del diseño del experimento 2.	35

5.6	Intensidad de campo eléctrico y transmitancia del diseño obtenidos en el experimento 2 simulado bajo una resolución de 40 <i>nm</i> tras ser discretizado.	35
5.7	Intensidad de campo eléctrico y transmitancia del diseño obtenidos en el experimento 3 simulado bajo una resolución de 40 <i>nm</i>	36

Capítulo 1

Motivación y Contexto

La fotónica es una ciencia que utilizamos diariamente, por ejemplo, mediante los cables ópticos. Los principales beneficios que ofrece son (i) elevado ancho de banda en comunicaciones, (ii) bajo consumo energético, (iii) interconexiones ópticas independientes de la distancia ([Glick et al., 2018](#)). El potencial de la fotónica se logra evidenciar en aplicaciones que aprovechan estos beneficios; por ejemplo, (i) interconexiones ópticas en centrales de datos ([Shen et al., 2019](#)), (ii) redes neuronales ópticas ([Shen et al., 2017](#)), (iii) internet de las cosas ([Glick et al., 2018](#)).

La importancia de estas aplicaciones en computación se evidencia aún más en la computación de alto rendimiento (HPC). Un problema presentado en el Top500 sistemas HPC es que desde el año 2010, el ratio entre el ancho de banda entre nodos y el poder de procesamiento por nodo ($byte/FLOP$) ha decrecido en un factor de seis. Así, se está llegando a un punto donde la capacidad para interconectar nodos está limitando el desempeño de sistemas HPC en programas que hacen uso extensivo de transferencias de memoria. Ante este problema se han propuesto arquitecturas que aprovechan la transmisión óptica porque esta puede realizar interconexiones a distancias del orden de metros. De esta manera se logra la transferencia de grandes cantidades de información con un bajo consumo energético y manteniendo un elevado ancho de banda ([Anderson et al., 2018](#), [Shen et al., 2019](#)).

Las aplicaciones señaladas anteriormente demuestran el potencial de la fotónica, mas estas aún no logran ponerse en práctica. Para que cada una de ellas funcione se necesita fabricar chips de fotónica integrada con una elevada eficiencia y densidad. Pero los

diseños actuales aún son muy ineficientes por lo cual sigue conviniendo utilizar dispositivos electrónicos ([Glick et al., 2018](#), [Vuckovic, 2019](#)). Ante esta dificultad, la fotónica en silicio (SiP) se presenta como un buen candidato para fabricar estos chips. La idea se refuerza por la adaptación de procesos de semiconductor complementario de óxido metálico (CMOS) para fabricar dispositivos SiP. Esto es un gran beneficio porque los procesos CMOS son una tecnología bien establecida y con precios accesibles ([Glick et al., 2018](#), [Lukas Chrostowski, 2010](#), [Shen et al., 2019](#)).

Por estos motivos, una cantidad considerable de investigaciones estudian dispositivos SiP ([Molesky et al., 2018](#)). El presente trabajo se centrará en dos de ellos: *bend-90°* y *wavelength demultiplexer* de dos canales. De aquí en adelante nos referiremos a estos simplemente como *bend* y WDM, respectivamente.

El documento está organizado de la siguiente manera:

El capítulo 1 brinda una introducción al tema de investigación, describe el problema a detalle, justifica la relevancia de resolverlo, define los objetivos y señala los aportes del trabajo.

El capítulo 2 desarrolla conceptos fundamentales en fotónica necesarios para entender el resto del documento.

El capítulo 3 presenta una revisión del estado del arte y discute sus principales características.

El capítulo 4 expone la metodología usada en la investigación.

El capítulo 5 muestra los resultados preliminares obtenidos con los experimentos.

1.1 Introducción

La fotónica está atrayendo el interés de la industria debido a su potencial en términos de escalabilidad y los beneficios de costo-eficiencia. Este potencial es evidente, por ejemplo, con los siguientes tres puntos. Primero, si se quiere mantener la tendencia que cada 10 años se mejore en un factor de 1000 el rendimiento de los sistemas electrónicos, entonces parece ser indispensable la convergencia de estos con sistemas fotónicos (Glick et al., 2018). Segundo, existe una inversión billonaria en la fabricación de transistores cuyos procesos se están adaptando para fabricar chips de fotónica integrada (Lukas Chrostowski, 2010). Tercero, hay prometedoras aplicaciones en (i) interconexiones ópticas (Shen et al., 2019), (ii) redes neuronales ópticas (Shen et al., 2017), (iii) internet de las cosas (Glick et al., 2018).

SiP se está estudiando con el fin de mostrar el potencial de la fotónica, pero esta tecnología aún se encuentra en la misma etapa de expansión que tenía la electrónica en los años 1970s. Afortunadamente, ya existen *foundries* y procesos como el CMOS que se pueden emplear como infraestructura para la fabricación de chips SiP (Lukas Chrostowski, 2010). Pero, los chips SiP siguen siendo ineficientes por lo cual se está buscando optimizar dispositivos fundamentales que lo componen (Vuckovic, 2019). Para esto existen dos estrategias: diseño tradicional y diseño inverso.

En el diseño tradicional se define el dispositivo con geometrías simples que permiten obtener funciones analíticas de sus propiedades físicas (Hughes and Fan, 2016, Song and Xie, 2008). Esto se realiza para poder optimizar la función obtenida a partir de los parámetros que la definan. Dicha optimización se suele ejecutar haciendo un barrido de los parámetros, con algoritmos genéticos o usando *particle swarm optimization*. Es un enfoque simple, pero que ha obtenido buenos resultados. Sin embargo, existen tres grandes inconvenientes con este planteamiento. Primero, solo estamos explorando una pequeña fracción de todos los posibles diseños. Segundo, por lo general no es conocido el límite

de rendimiento del dispositivo. Tercero, al trabajar en la escala de nanómetros, existen casos como el *bend* y WDM que presentan un bajo rendimiento con diseños tradicionales (Molesky et al., 2018, Su et al., 2020).

En el diseño inverso se busca hacer una mayor exploración de todos los posibles diseños. Para ello, ya no nos limitamos a solo usar diseños intuitivos, ver figura 1.1. Ahora, definimos geometrías arbitrarias y usamos simulaciones computacionales para determinar las propiedades físicas del dispositivo (Molesky et al., 2018, Su et al., 2020). Este enfoque ha logrado conseguir mejores resultados que los obtenidos por el diseño tradicional (Molesky et al., 2018, Su et al., 2018). Sin embargo, este planteamiento viene acompañado de nuevos desafíos.

El diseño inverso ha recibido mucha atención en fotónica durante los últimos 20 años (Molesky et al., 2018). Los autores en Su et al. (2020) muestran estrategias para optimizar un *bend* y WDM. En Piggott et al. (2017), Su et al. (2018) se incorporan restricciones de fabricación para obtener dispositivos que al fabricarse mantengan un buen rendimiento. No obstante, la búsqueda de estos diseños suele realizarse aplicando algoritmos cuya selección y configuración es mayormente debido a un tedioso proceso de prueba y error. Otros trabajos como los de Elsaywy et al. (2020), Gregory et al. (2015), Schneider et al. (2019) comparan distintos algoritmos de optimización en el diseño inverso de dispositivos fotónicos.

Aún cuando una cantidad considerable de investigaciones están usando el diseño inverso para optimizar dispositivos fotónicos, el *bend* y WDM, dispositivos fundamentales para los circuitos fotónicos, aún no logran aplicación comercial (Molesky et al., 2018). Además, existe una carencia de estudios con un enfoque computacional que intenten abordar el problema. Así, el objetivo de esta tesis es aplicar el conocimiento en computación para optimizar estos dispositivos y conseguir diseños que al fabricarse muestren un desempeño elevado.



(a) *Bend* con diseño tradicional



(b) *Bend* obtenido con diseño inverso. Extraído de (Su et al., 2020)



(c) WDM con diseño tradicional



(d) WDM obtenido con diseño inverso. Extraído de (Su et al., 2020)

FIGURA 1.1: Diseños tradicionales y obtenidos a partir de diseño inverso de un *bend* y un WDM

1.2 Descripción del Problema

Para poder describir el funcionamiento de un dispositivo se calcula la distribución del campo eléctrico, para ello se resuelven las ecuaciones de Maxwell (Schneider et al., 2019). Una forma de encontrar la solución a estas ecuaciones en cualquier geometría es utilizando un método numérico llamado diferencias finitas en el dominio de frecuencias (FDFD) (Su et al., 2020). Con este planteamiento se selecciona una región rectangular

a optimizar y se la divide en $n \times m$ píxeles como si fuera una imagen, ver Figura 1.2. Luego, cada píxel se rellena con dos posibles materiales: óxido de silicio (SiO_2) o silicio (Si) (Molesky et al., 2018).

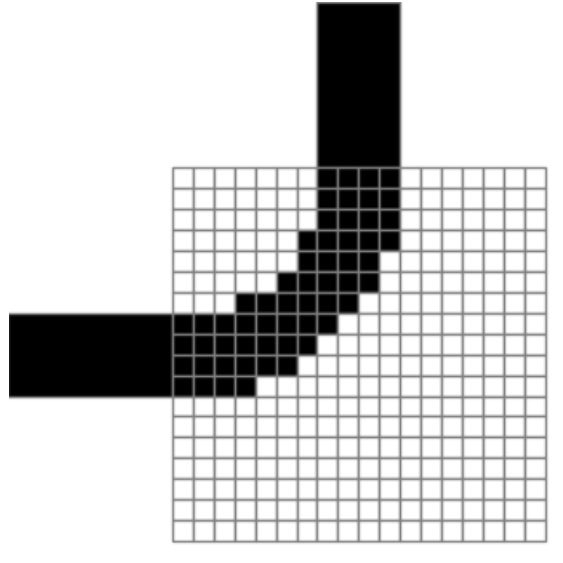


FIGURA 1.2: *Bend* con una región de diseño discretizada en 18×18 píxeles. Cada píxel negro representa la presencia de Si y cada píxel blanco de SiO_2

El diseño inverso comienza definiendo los requerimientos del dispositivo para luego tratar de buscar entre los $2^{n \times m}$ posibles diseños algún candidato que se adapte a lo que se busca (Molesky et al., 2018, Su et al., 2020). Como prueba de concepto, trabajos como el de Malheiros-Silveira and Delalibera (2020) parametrizaron $2^{10 \times 10}$ posibles geometrías. Así, se presentan algunas dificultades con esta estrategia:

1. No es viable evaluar todos los posibles diseños por haber un número excesivamente elevado de ellos (Vuckovic, 2019).
2. Las simulaciones computacionales son muy costosas en términos de memoria y tiempo (Kudyshev et al., 2020).
3. El espacio de búsqueda es no convexo (Su et al., 2018).
4. No todos los diseños son fabricables por limitaciones físicas (Su et al., 2020).

5. Cada dispositivo es una clase distinta de problema, es decir, no necesariamente funcionará la misma estrategia para cada dispositivo (Molesky et al., 2018).

Además, la fabricación viene con otros desafíos, principalmente:

1. Errores de precisión (Piggott et al., 2017).
2. Sensibilidad ante cambios de temperatura (Vuckovic, 2019).

Considerando las anteriores dificultades, el problema es usar diseño inverso y encontrar geometrías que muestren buen desempeño en simulaciones computacionales y que puedan asegurar mantener un óptimo funcionamiento al ser fabricados. Este problema se estudiará para dos dispositivos nanofotónicos (i) *bend* y (ii) WDM.

1.3 Justificación

El *bend* y WDM son dispositivos fundamentales en los circuitos fotónicos que tienen aplicación directa, por ejemplo, en sistemas HPC (Shen et al., 2017). Por otro lado, desde el punto de vista computacional, este problema es interesante porque ya hay estrategias computacionales conocidas para resolverlo, desde algoritmos evolutivos (Hansen, 2016) hasta redes neuronales (Goodfellow et al., 2015) y *depth learning* (Malkiel et al., 2018). Además, debido al alto costo computacional de las simulaciones (Schneider et al., 2019), el trabajo requiere de computación de alto desempeño. Así, es probable que se pueda obtener buenos resultados en la investigación aplicando el conocimiento ya existente en computación.

1.4 Objetivos

- Fabricar un *bend* y WDM con eficiencias mayores al noventa por ciento.

- Usando HPC, estudiar el desempeño y la convergencia de cinco algoritmos de optimización populares usados para optimizar dispositivos nanofotónicos.

1.5 Aportes

Este trabajo busca brindar una comparativa de las técnicas de optimización más relevantes que se aplican para optimizar un *bend* y un WDM cuando estos son parametrizados con un elevado número de variables.

Capítulo 2

Marco Teórico

En el presente capítulo se explican conceptos y terminología relacionada al diseño inverso de dispositivos fotónicos. Para ello se desarrolla tres secciones. En primer lugar, se describe las propiedades físicas de interés de un *bend* y WDM. En segundo lugar, se desarrolla en cinco pasos como utilizar el diseño inverso para optimizar los dispositivos de estudio.

- Se especifica tres estrategias comúnmente usadas para parametrizar la región de diseño.
- Se muestra como definir la función objetivo para un *bend* y WDM.
- Se señala los programas donde se puede simular el comportamiento de los diseños que desarrollemos.
- Se expone como se explican estrategias de optimización para un dispositivo fotónico.
- Se da a conocer transformaciones que se pueden incluir al plan de optimización para incorporar restricciones de fabricación.

En tercer lugar, se explica la conversión de los resultados de la optimización a un formato listo para ser fabricado.

2.1 Dispositivos de estudio

2.1.1 *Bend*

Un *bend* es un dispositivo fotónico que se encarga de guiar un haz de ondas para que gire.

En general, al estudiar dispositivos fotónicos es de especial interés la distribución del campo eléctrico. Este campo se descompone en una componente transversal eléctrica (TE) y en una componente transversal magnética (TM) de acuerdo a la siguiente ecuación:

$$E = E^{TE} + E^{TM} \quad (2.1)$$

Donde E^{TE} es la componente paralela al dispositivo y E^{TM} es la componente restante ([Hohenester, 2020](#)).

La intensidad de estos campos es de especial interés pues nos dan una idea del rendimiento del dispositivo. Por ejemplo, en la figura 2.1 se muestra la intensidad del campo E^{TE} cuando un haz de luz está pasando por el dispositivo. Como podemos observar, parte del campo está fuera del *bend* representado por color morado. Así, se visualiza de forma gráfica que el dispositivo está teniendo considerables pérdidas de energía lo cual es un indicador de un mal rendimiento.

Por otro lado, para evaluar el desempeño de un *bend* de forma numérica se suele calcular la transmitancia como la relación entre la intensidad del haz que sale del dispositivo con la intensidad con la que entra ([Su et al., 2020](#)). Esto se expresa mediante la ecuación 2.2:

$$T = \frac{I}{I_0} \quad (2.2)$$

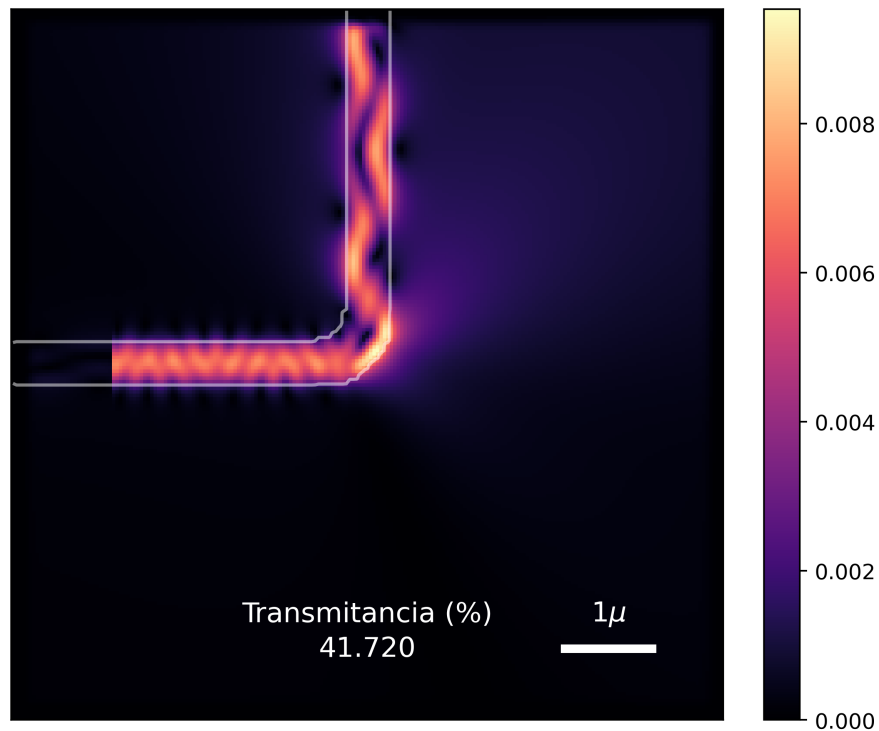


FIGURA 2.1: Intensidad de campo eléctrico para un *bend*-90° de radio interno de 0.25 μm

2.1.2 Wavelength Demultiplexer de dos canales (WDM)

Un WDM es un dispositivo fotónico que se encarga de guiar un haz de ondas de acuerdo a su longitud de onda. Así, estos suelen trabajar con dos longitudes de onda y guían las de un tipo por la guía de onda superior y las de otro tipo por la guía de onda inferior.

Similar al caso del *bend* se estudia su campo eléctrico y transmitancia para medir el rendimiento de estos dispositivos.

Un dispositivo similar al WDM es un *splitter*. Este dispositivo tiene la misma geometría, pero trata de dividir lo que recibe.

2.2 Diseño inverso

2.2.1 Parametrización

Tres estrategias comunes para parametrizar un dispositivo fotónico son: i) parametrización por píxeles, ii) parametrización por conjuntos de nivel, iii) parametrización por segmentos.

2.2.1.1. Parametrización basada en topología

A cada píxel se le asocia un valor dado por la fórmula 2.3 (Su et al., 2020):

$$\varepsilon(x, y) = \varepsilon_{Si} + (1 - \lambda_{x,y})\varepsilon_{SiO_2} \quad \lambda_{x,y} \in [0, 1], \lambda_{x,y} \in \mathbb{R} \quad (2.3)$$

Donde $\varepsilon_{Si} = 3.48$ es la permitividad del Si y $\varepsilon_{SiO_2} = 1.44$ es la permitividad del SiO_2 . Con esta ecuación se mapea el intervalo $[0, 1]$ con el intervalo $[1.44, 3.48]$. Esto se realiza para determinar la permitividad que hay en la ubicación del píxel y poder simular las ecuaciones de Maxwell en el dispositivo. Con esta parametrización obtenemos una cantidad infinita de diseños, mas solo nos interesan aquellos donde $\lambda_{x,y}$ es entero, pues en caso contrario un píxel se mapea a la permitividad de un material desconocido lo cual lo volvería infabricable.

En la figura 2.2 se muestra la parametrización topológica de un *bend-90°*. Las píxeles en gris representan regiones donde se está usando una permitividad desconocida, es decir se ha tomado un $\lambda_{x,y}$ no entero.

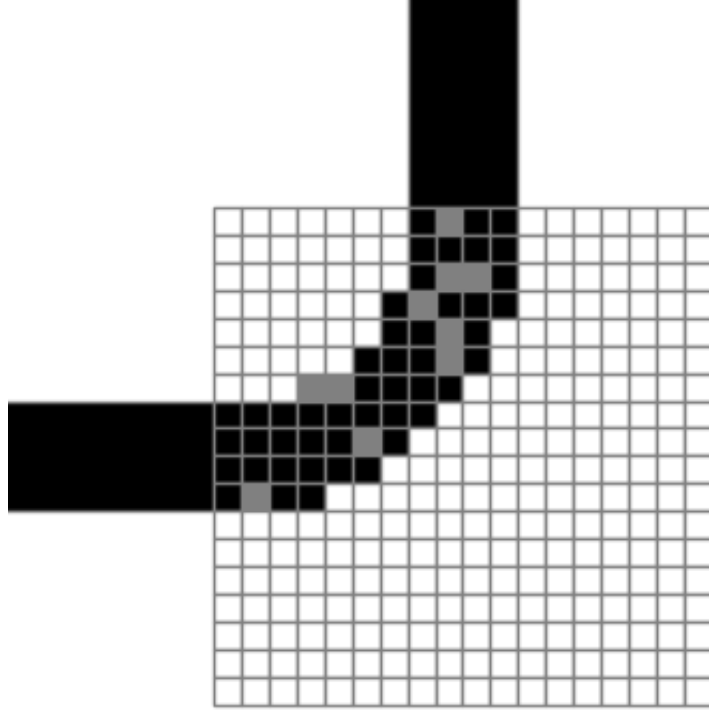


FIGURA 2.2: Parametrización por píxeles para un *bend-90°*

2.2.1.2. Parametrización por conjuntos de nivel

A cada píxel se le asocia una permitividad dada por la siguiente fórmula ([Piggott et al., 2017](#)):

$$\varepsilon(x, y) = \begin{cases} \varepsilon_{Si} & \text{si } \phi(x, y) \leq 0 \\ \varepsilon_{SiO_2} & \text{si } \phi(x, y) > 0 \end{cases}$$

Donde ϕ se define como una función continua. En la figura 2.3 se muestra como esta parametrización permite mantener curvas continuas para representar al dispositivo (líneas amarillas). Luego, los píxeles intentan cubrir la región definida por estas curvas para después poder simular numéricamente las propiedades del dispositivo.

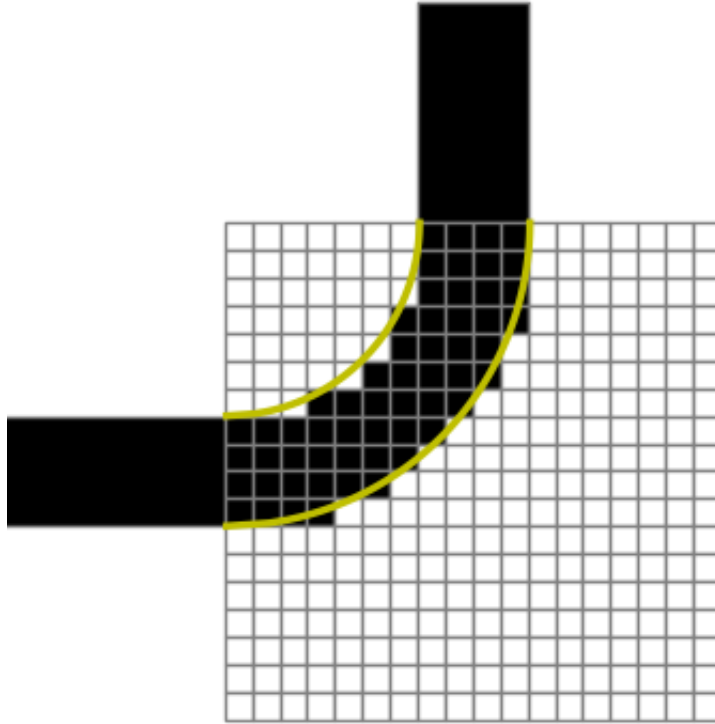


FIGURA 2.3: Parametrización por conjuntos de nivel para un *bend-90°*

2.2.1.3. Parametrización por segmentos

Con esta estrategia se considera la región de diseño como un rectángulo y se lo divide en n segmentos de igual tamaño. Luego, el tamaño de cada segmento será un parámetro y la geometría se forma a partir de estos valores considerando ubicar cada segmento centrado verticalmente.

Las distintas estrategias de parametrización mostradas tienen sus propias características. La parametrización por segmentos asegura simetría y regiones compactas, pero tiene un espacio de búsqueda más reducido que las otras opciones. Por otro lado, la parametrización topológica trabaja con un espacio de búsqueda mucho más grande a cambio de poder parametrizar diseños no fabricables. En cambio, la parametrización por niveles se encarga de mantener una curva continua con el fin de mantener el diseño similar a como terminará el dispositivo fabricado.

2.2.2 Definición de la función objetivo

La función objetivo es un valor asociado a una parametrización que nos permite comparar distintas parametrizaciones para determinar que diseño es mejor. En el área de fotónica es común referirse a esta función como la figura de mérito (FOM). Basándonos en [Su et al. \(2020\)](#) se tiene que conviene definir la FOM de la siguiente manera:

1. *Bend*

$$f_{obj}(p) = \max \{T(p)\} \quad (2.4)$$

Es decir, la función objetivo asociada a la parametrización p es maximizar la transmitancia, ver ecuación 2.2, asociada a esa parametrización.

2. WDM

La función objetivo se define como:

$$f_{obj}(p) = \max \{g_0(p, 0)^2 + (1 - g_0(p, 1))^2 + g_1(p, 1)^2 + (1 - g_1(p, 0))^2\} \quad (2.5)$$

Donde $g_0(p, i)$ representa la transmitancia asociada a la parametrización p en la guía de onda i para una longitud de onda de $1400nm$ y $g_1(p, i)$ representa su análogo para una longitud de onda de $1550nm$.

La ecuación 2.5 busca maximizar la transmitancia por la guía de onda superior y minimizarla para la guía de onda inferior cuando se recibe una longitud de onda de $1400nm$ y lo contrario para una longitud de onda de $1550nm$.

2.2.3 Simulación

Existen distintas librerías de Python que nos permiten definir dispositivos fotónicos en base a lo descrito. Dos librerías muy usadas para ello son MEEP ([Oskooi et al., 2010](#)) y SPINS ([Su et al., 2020](#)). Una evaluación cualitativa de sus funcionalidades puede ser vista en la tabla 2.1.

Librería	Usabilidad	Eficiencia	Bugs	Funcionalidad
MEEP	Difícil	Alta	Elevados	Extensa
SPIN	Moderada	Moderada-Alta	Pocos	Básica necesaria

TABLA 2.1: Evaluación cualitativa de las librerías MEEP y SPINS

Ambas librerías se encargan de realizar la simulación en una región cuadrada, para ello dividen la región de simulación como si fuera una cuadrícula. Definiendo la resolución podemos especificar el tamaño de esta cuadrícula. Entre más pequeña sean las divisiones de esta, más cercanos a los valores reales sean los resultados de la simulación. Sin embargo, esto implica un mayor tiempo de ejecución.

2.2.4 Estrategias de optimización

La forma de optimizar un diseño depende de la parametrización utilizada. De esta forma, tenemos los siguientes casos:

1. Parametrización topológica

La ecuación 2.3 ya señala un rango de valores asociados a cada píxel. Pero, para asegurar el obtener un diseño fabricable se divide la optimización en dos pasos ([Su et al., 2020](#)):

a) Optimización continua

Se varía el valor de $\lambda_{x,y}$ en el intervalo $[0, 1]$ sin importar si se obtiene diseños no fabricables.

b) Optimización discreta

Se considera el resultado de la optimización continua como punto inicial del algoritmo de optimización. Luego, se va aplicando una transformación que permita ir convergencia a una parametrización donde cada $\lambda_{x,y}$ tenga un valor entero. Algunos ejemplos de estas transformaciones se pueden encontrar en la siguiente sección.

2. Parametrización por conjuntos de nivel

Como lo detalla [Piggott et al. \(2017\)](#), con este tipo de parametrización es conveniente utilizar una optimización basada en la gradiente y asegurar convergencia con una búsqueda en línea.

3. Parametrización por segmentos

Basándonos en [Prosopio-Galarza et al. \(2019\)](#), con este tipo de parametrización conviene definir un rango de posibles valores para cada segmento. Luego, ejecutar algún algoritmo de optimización, de preferencia uno de optimización global, para finalmente tratar de incorporar restricciones de fabricación en una etapa posterior.

2.2.5 Transformaciones

La aplicación de transformaciones a un diseño se realiza con el fin de obtener dispositivos con curvas suaves para asegurar un buen desempeño al momento de fabricarse ([Su et al., 2020](#)). De acuerdo al tipo de parametrización utilizada, tenemos:

- Para el caso de la parametrización por segmentos, si queremos suavizar el diseño obtenido podemos considerar las alturas de los segmentos e interpolar una curva ([Ma et al., 2013](#)).
- Para el caso de la parametrización por conjunto de nivel se define ϕ de forma que asegure curvas suaves, así no es necesario aplicar transformaciones.

- Para el caso de la parametrización topológica, una forma de suavizar las regiones punteagudas es aplicar una función de suavizado conforme va iterando el algoritmo de optimización aplicado. Basándonos en [Zhang et al. \(2021\)](#), podemos aplicar la siguiente función:

$$s(p) = \frac{\tanh(\beta \times \eta) + \tanh(\beta \times (p - \eta))}{\tanh(\beta \times \eta) + \tanh(\beta \times (1 - \eta))} \quad (2.6)$$

Donde p es la parametrización del dispositivo, $\eta = 0.5$ y β comienza con un valor de 1 y va incrementándose exponencialmente en cada iteración. Como se observa en la figura 2.4, la ecuación 2.6 se encarga de ir haciendo converger los valores de la parametrización a 0 o 1 de acuerdo a cual esté más cercano. Conforme aumenta el valor de β esta convergencia es más rápida.

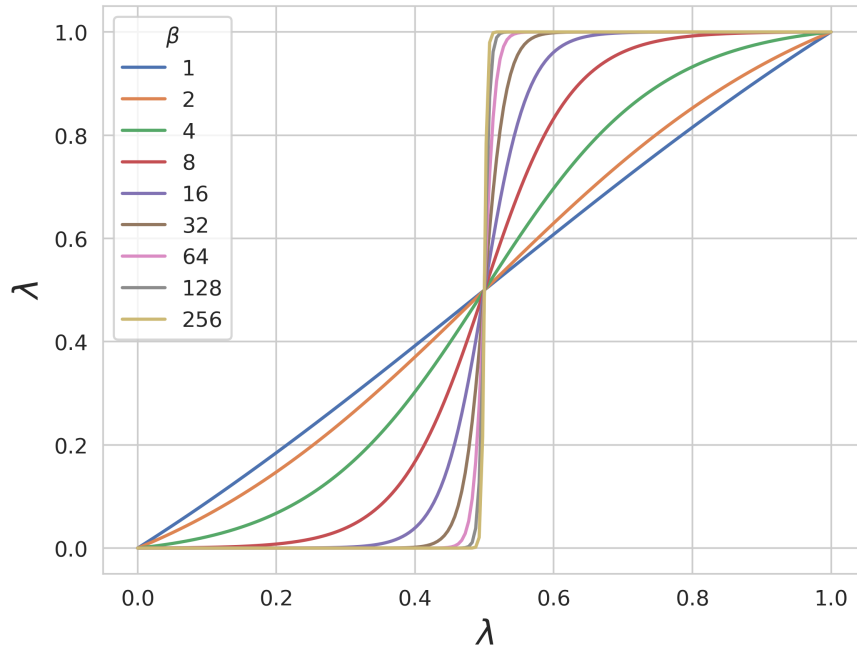


FIGURA 2.4: Función de discretización con $\eta = 0.5$ y distintos valores de β

2.3 Preparación para fabricación

Para poder fabricar nuestros diseños necesitamos representarlos en formato GD-SII. En este formato se especifica los polígonos que representan a nuestro dispositivo. Independientemente de la parametrización utilizada, mediante la simulación con SPINS o MEEP se puede obtener los puntos asociados al contorno del dispositivo. Así, utilizando lo anterior se construye los polígonos ([Bogaerts and Chrostowski, 2018](#)).

Capítulo 3

Trabajos relacionados

En el presente capítulo se resume los puntos principales de cinco trabajos relacionados al problema de investigación.

3.1 [Su et al. \(2020\)](#)

[Su et al. \(2020\)](#) divide una región cuadrada de $2.5\mu m \times 2.5\mu m$ en píxeles de $40nm \times 40nm$ y luego lo une a guías de onda de $40nm$ para diseñar un WDM. La definición de su función objetivo es como la descrita en la ecuación 2.5, pero además muestra ideas de como definir esta función de manera que la optimización intente acercarse a un valor de transmitancia deseado.

Como algoritmo de optimización se utiliza $L - BFGS - B$ ejecutando 100 iteraciones en la etapa continua y 180 en la discreta. El proceso completo de optimización se realizó en SPINS demorando catorce horas y media al ser ejecutado en una computadora con 4 GPU *Titan Black* y 2 CPU con 64 GB de RAM. Sin embargo, solo utilizó menos de 4 GB de RAM y cada simulación demoró alrededor de un minuto.

Para asegurar restricciones de fabricación, [Su et al. \(2020\)](#) se aseguró que el tamaño de cada píxel sea mayor al mínimo tamaño que se podía fabricar.

Desde el punto de vista computacional hay dos aspectos mejorables en este trabajo. Primero, se podría incrementar la cantidad de iteraciones, realizar las optimizaciones más de una vez y probar más algoritmos. Los anteriores puntos aumentarían las posibilidades de obtener mejores resultados. Segundo, la arquitectura utilizada se podría reemplazar por un sistema HPC para acelerar la optimización.

3.2 Piggott et al. (2017)

Piggott et al. (2017) diseña un *3-splitter* dividiendo una región de $3.8\mu m \times 2.5\mu m$ en píxeles de $40nm \times 40nm$. En la definición de su función objetivo busca trabajar con 3 longitudes de onda y que con cada longitud la transmitancia se maximice en una guía de onda de salida y se minimice en las otras, similar al trabajo de Su et al. (2020) en la optimización de un *2-splitter*.

Luego, utilizando SPINS como simulador, realiza la optimización con un algoritmo propio que está basado en *gradient-descent* y *line search*. Con este algoritmo se encarga de incorporar restricciones del mínimo radio de curvatura que puede contener el diseño y limita el tamaño máximo que pueden tener los huecos que se formen en el dispositivo.

En este trabajo, una estrategia alternativa podría haber sido aplicar algún algoritmo de optimización global en la etapa de optimización continua para luego recién utilizar uno basado en la gradiente. Considerando que los resultados de los algoritmos basados en la gradiente suelen ser dependientes del diseño inicial, esta estrategia hubiera sido interesante de evaluar.

3.3 Prosopio-Galarza et al. (2019)

Prosopio-Galarza et al. (2019) utiliza parametrización por segmentos para dividir un rectángulo de $2\mu m \times 1.5\mu m$ en 13 rectángulos verticales idénticos que particionan la región de diseño. Luego, esto es unido a tres guías de onda fijas de $0.5\mu m$ para formar un *splitter* de un espesor de $0.2\mu m$.

Como función objetivo se establece maximizar la transmitancia. Seguidamente, por separado utiliza tres algoritmos para optimizar el diseño: i) *Particle Swarm Optimization*, ii) *Shrinking Box algorithm*, iii) *Steepest Ascent algorithm*.

La simulación se realiza usando ANSY Lumerical FDTD. Los experimentos solo se repitieron una vez y se ejecutó cada algoritmo durante 30 iteraciones. Probablemente, el usar este simulador mediante la interfaz gráfica limitó la automatización de los experimentos para conseguir una mayor cantidad de datos a comparar. Pero, es destacable que en su investigación [Prosopio-Galarza et al. \(2019\)](#) logra determinar que los ángulo más agudos, los cuales son los más adecuados como regla práctica de diseño, son formados con la optimización mediante *Particle Swarm Optimization* (PSO).

3.4 [Hammond et al. \(2020\)](#)

[Hammond et al. \(2020\)](#) utiliza MEEP para optimizar un *splitter* utilizando parametrización topológica. Lo más relevante de su trabajo es que a cada diseño le aplica transformaciones para simular la posible contracción o dilatación de estos al fabricarse. Con lo anterior hace un intento de detectar errores de fabricación. Sin embargo, el trabajo carece de la fabricación de los resultados para validar la propuesta.

3.5 [Elsawy et al. \(2020\)](#), [Gregory et al. \(2015\)](#), [Schneider et al. \(2019\)](#)

[Schneider et al. \(2019\)](#) compara cinco algoritmos de optimización para optimizar dos dispositivos fotónicos. Los algoritmos usados son (i) L-BFGS-B, (ii) PSO, (iii) *differential evolution* (DE), (iv) optimización bayesiana, (v) búsqueda aleatoria. Para ello parametriza los dispositivos con menos de 15 parámetros. Cada algoritmo tiene sus propias particularidades, por eso utiliza limita la cantidad de evaluaciones a 2500. Esto le permite comparar la cantidad de simulaciones con el FOM obtenido. Además, realiza la optimización 6 veces por cada algoritmo para terminar comparando los valores medios.

Los resultados de [Schneider et al. \(2019\)](#) muestran que la optimización bayesiana supera a los otros algoritmos en términos de FOM y de rapidez de convergencia. Además,

logra confirmar que la optimización de dispositivos fotónicos requiere de un elevado número de simulaciones. Sin embargo, los algoritmos solo son ejecutados con valores por defecto debido a que la optimización de sus parámetros no es viable por el costo computacional implicado.

[Gregory et al. \(2015\)](#) realiza una investigación similar a la de [Schneider et al. \(2019\)](#), pero se centra en un solo algoritmo: covariance matrix adaptation evolution strategy (CMA-ES). También logra concluir la importancia de realizar un elevado número de simulaciones y por ello propone al CMA-ES como un buen algoritmo a utilizarse en fotónico. Esto se debe a que este algoritmo solo requiere un punto inicial sobre el cual comenzar, es un método de optimización global y cambiando el parámetro σ podemos limitar la búsqueda alrededor de un diseño.

Por otro lado, [Elsawy et al. \(2020\)](#) hace una revisión de cuatro algoritmos populares en la optimización de dispositivos fotónicos: (i) algoritmos genéticos (GA), (ii) PSO, (iii) CMA-ES, (iv) optimización bayesiana. Similar al trabajo de [Schneider et al. \(2019\)](#), sus conclusiones sugieren que la optimización bayesiana logra superar a otros algoritmos de optimización en distintos escenarios.

Capítulo 4

Propuesta

Como se muestra en la figura 4.1, en esta investigación se seguirán los siguientes siete pasos: (i) parametrización, (ii) función objetivo, (iii) optimización continua, (iv) optimización discreta, (v) optimización de fabricación, (vi) preparación para fabricación y (vii) fabricación.

Se comienza con un diseño aleatorio y los pasos son secuenciales ya que el resultado de uno es necesario para comenzar el siguiente. Esto se realiza así para mantener un buen resultado (Yang and Fan, 2017). El valor de β es en referencia a la ecuación 2.6, esto representa el factor discretizador de nuestros diseños.

En las siguientes subsecciones se explica en detalle cada una de los pasos de la metodología seguida en esta tesis.

4.0.1 Parametrización

El *bend* y WDM de dos canales se parametrizarán usando la parametrización basada en topología. La descripción detallada para los dos dispositivos de estudio se presentan en las siguientes dos subsecciones.

4.0.1.1. *Bend*

En la figura 4.2 se muestra el diseño y dimensiones del *bend* a utilizar. Los rectángulo negros representan las guías de onda, estos tienen las mismas dimensiones aunque en distinta orientación. La región gris representa la región de diseño, esta región ha sido

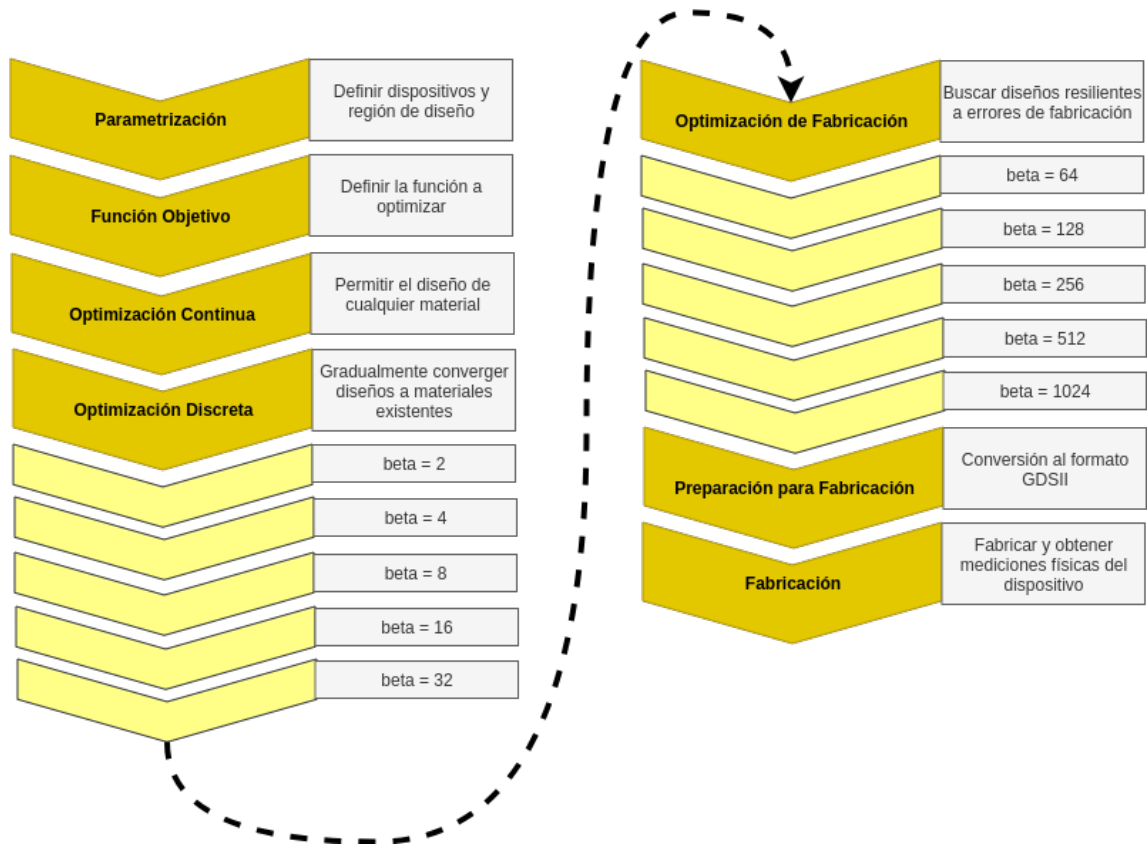


FIGURA 4.1: Metodología del trabajo de investigación

dividida en una cuadrícula de 20×20 . Cada píxel tiene $60nm$ de lado y una permitividad asociada dada por la ecuación 2.3.

4.0.1.2. WDM

En la figura 4.3 se muestra el diseño y dimensiones del WDM a utilizar. Los rectángulos negros representan las guías de onda, estos tienen las mismas dimensiones. La región gris representa la región de diseño, esta región ha sido dividida en una cuadrícula de 20×20 . Cada píxel tiene $60nm$ de lado y una permitividad asociada dada por la ecuación 2.3.

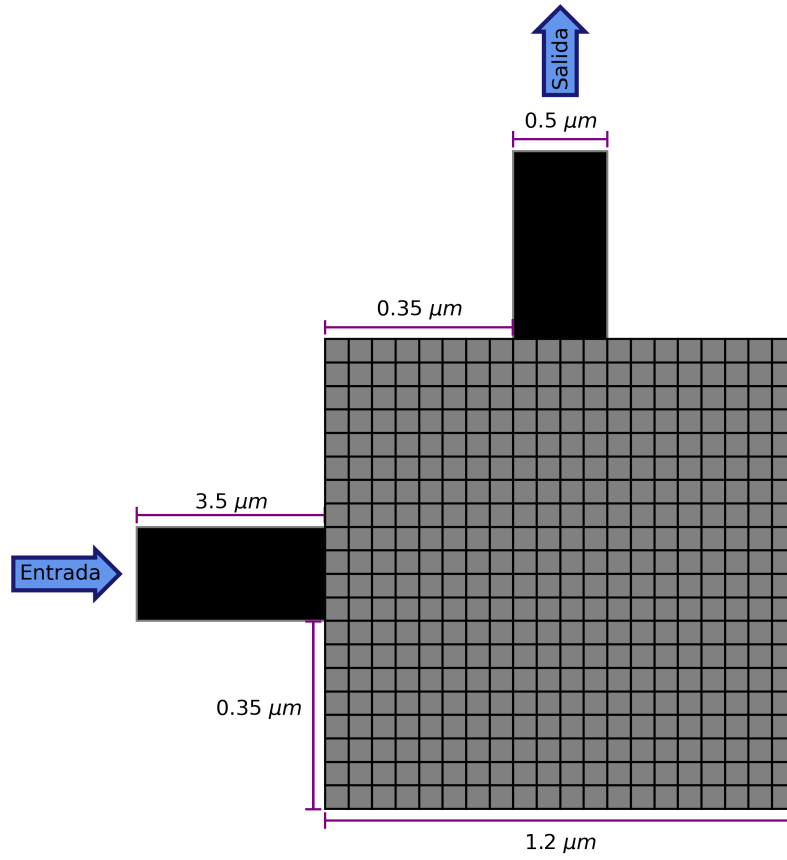


FIGURA 4.2: Dimensiones del diseño del *bend* a utilizar.

4.0.2 Función objetivo

Basándonos en el trabajo de [Su et al. \(2020\)](#), para el *bend* usaremos como función objetivo la ecuación 2.4 y para el WDM la ecuación 2.5.

4.0.3 Optimización

El proceso de optimización se dividirá en tres etapas: (i) optimización continua, (ii) optimización discreta y (iii) optimización de fabricación. Tomando como base a [El-sawy et al. \(2020\)](#), [Gregory et al. \(2015\)](#), [Schneider et al. \(2019\)](#), este proceso se realizará con los siguientes cinco algoritmos: (i) L-BFGS-B, (ii) *covariance matrix adaptation*

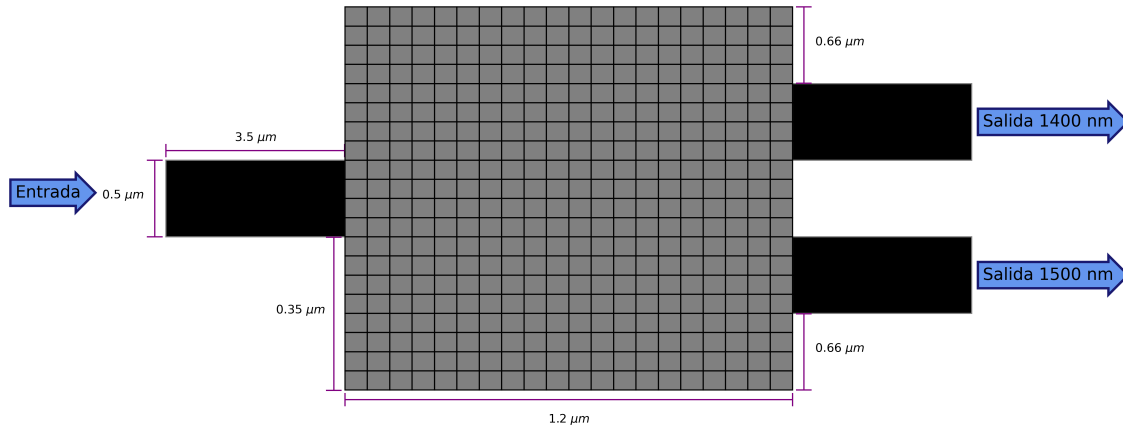


FIGURA 4.3: Dimensiones del diseño del WDM a utilizar.

evolution strategy (CMA-ES), (iii) *particle swarm optimization* (PSO), (iv) algoritmos genéticos (GA), (v) optimización bayesiana (OB).

Similar a [Schneider et al. \(2019\)](#), para poder comparar los resultados el proceso se repetirá 6 veces por cada algoritmo y en cada repetición se fijará el valor de la semilla con el fin de asegurar reproducibilidad de los experimentos. Además, en cada etapa de la optimización se realizará un cambio de sus parámetros, en la medida de lo posible, para asegurar ir convergiendo a un resultado óptimo. Adicionalmente, se limitará la cantidad de posibles diseños a simular. Así se podrá hacer una comparación del desempeño de los algoritmos en base a la cantidad de evaluaciones de la función objetivo. Sin embargo, debido a que estos algoritmos trabajan de forma iterativa con sus propias particularidades, se considerará como condición de salida el estar en la primera iteración donde ya se haya realizado la cantidad máxima de simulaciones.

A continuación, se describirá las características de cada etapa del proceso de optimización.

4.0.3.1. Optimización continua

En este primer paso se usará directamente la ecuación 2.3 para buscar dispositivos óptimos de acuerdo al FOM definido. Así, es probable obtener diseños con muy buenos resultados, pero con permitividades no asociadas al SiO_2 o Si . Cada algoritmo comenzará con un diseño inicial aleatorio y se limitará en 5000 la cantidad de posibles diseños a simular. Las simulaciones se realizarán en 2D.

4.0.3.2. Optimización discreta

Bajo la premisa que la anterior etapa obtuvo un buen resultado, cada algoritmo tomará como punto inicial el resultado de la optimización previa y tratará de configurar sus parámetros para realizar la búsqueda principalmente alrededor de ese punto.

Además, se aplicará la ecuación 2.6 al parámetro λ de la ecuación 2.3 para gradualmente terminar mapeando al SiO_2 o Si en cada píxel. Para ello se realizará 5 optimizaciones donde el resultado de una funcionará como punto inicial de la siguiente. En cada una de estas optimizaciones se limitará en 5000 la cantidad de posibles diseños a simular y se tomará como parámetro β de la ecuación 2.6 el valor dado por la siguiente recurrencia:

$$\begin{aligned}\beta_1 &= 2 \\ \beta_n &= \beta_{n-1} \times 2\end{aligned}\tag{4.1}$$

Las simulaciones se realizarán en 2D.

4.0.3.3. Optimización de fabricación

Con la anterior etapa la ecuación 2.6 nos asegura diseños que están convergiendo a dispositivos reales. Ahora, tomando como referencia el trabajo de [Hammond et al. \(2020\)](#) para asegurar un buen desempeño pese a los errores de fabricación, por cada parametrización p se calculará:

- p_{ext} que representará el diseño como si el dispositivo se hubiera dilatado.
- p_{int} que representará el diseño como si el dispositivo se hubiera contraído.

Luego, se definirá como nueva función objetivo la siguiente ecuación:

$$\begin{aligned} F_{obj} &= \omega_1 f_{obj}^2(p) + \omega_2 (1 - |f_{obj}(p_{ext}) - f_{obj}(p_{int})|)^2 \\ \omega_1 &= 0.85 \\ \omega_2 &= 0.15 \end{aligned} \tag{4.2}$$

Así, se buscará un diseño que tenga un buen desempeño y que sea resiliente a posibles errores de fabricación. Además, para asegurar que los resultados sean más congruentes con la realidad las simulaciones se realizarán en 3D con la ayuda de un GPU.

4.0.4 Preparación para fabricación

Para que nuestros diseños puedan ser fabricados, necesitamos representarlo en formato GDSII. Para ello, con los mejores resultados de cada algoritmo usaremos la distribución de la permitividad asociada a su parametrización, $\epsilon(p)$, para obtener la geometría del dispositivo. Así, nuestra tarea se reducirá a aproximar el contorno de esta geometría con polígonos para luego realizar la conversión al formato GDSII. Además, usaremos el

resultado en formato GDSII para simular el funcionamiento del dispositivo y asegurar que siga manteniendo las propiedades deseadas. En caso no se tenga éxito, se volverá a iterar en la conversión

4.0.5 Fabricación

Se mandará a fabricar los mejores resultados hasta cubrir el área dispuesta para fabricación. Luego, se esperará las mediciones de los dispositivos ya fabricados para compararlo con los resultados de las simulaciones computacionales.

Capítulo 5

Resultados Preliminares

En el presente capítulo se muestran resultados preliminares de la investigación. Estos resultados se obtuvieron siguiendo la propuesta descrita en el anterior capítulo, pero llegando solo hasta la optimización continua. Además, los experimentos se realizaron solo con un *bend* utilizando un único algoritmo: CMA-ES. Con las anteriores restricciones se realizaron tres experimentos.

5.1 Experimento 1

Como primer experimento se comenzó configurando un *bend* con las características descritas en la figura 4.2. Luego, se procedió a realizar la optimización. Para la optimización continua se limitó la cantidad de simulaciones a 5000 y para cada una de las 5 etapas de la optimización discreta se limitó en 5000 simulaciones.

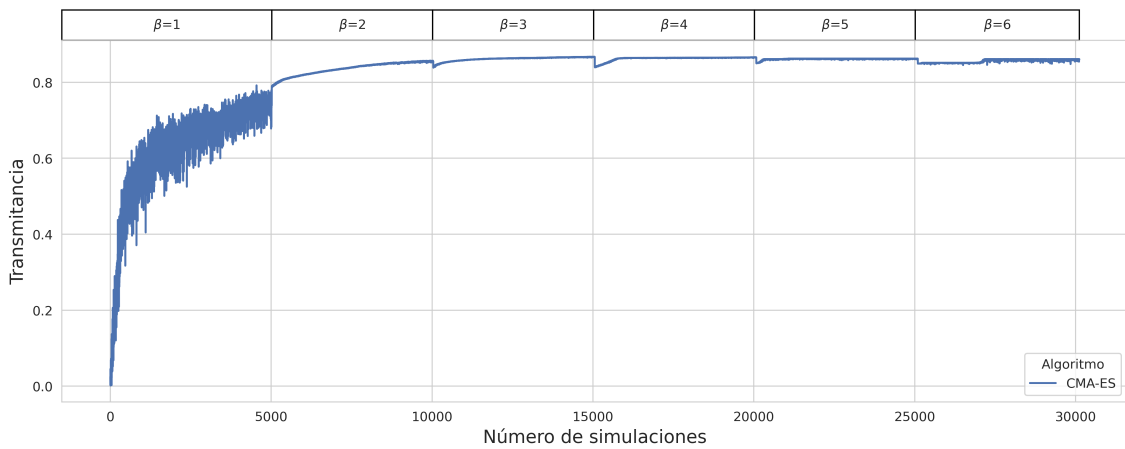


FIGURA 5.1: Gráfica del número de simulaciones vs transmitancia producto de optimizar un *bend* usando CMA-ES en el experimento 1.

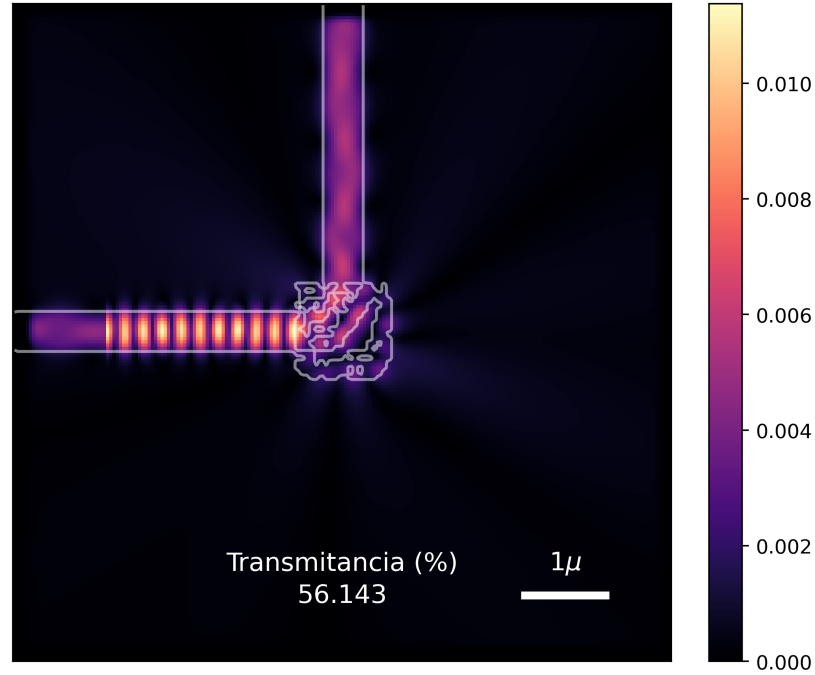


FIGURA 5.2: Intensidad de campo eléctrico y transmitancia del diseño obtenidos en el experimento 1 simulado bajo una resolución de 40 nm . La línea blanca ubicada en la parte inferior derecha indica la escala de $1\text{ }\mu\text{m}$ en la figura.

En la figura 5.1 se muestra la transmitancia de cada uno de los diseños evaluados. Podemos observar que en la etapa de la optimización continua, $\beta = 1$, es cuanto mayor crecimiento se logra. Luego, en los distintos pasos dentro de la optimización discreta ($\beta = 2, 3, 4, 5, 6$) el crecimiento es más lento. Sin embargo, es destacable el hecho que aún al incrementar el valor de β , la simulación sigue logrando explorar diseños con valores de transmitancia similares. Para lograr esto se comenzó ejecutando el algoritmo con un valor de $\sigma = 0.3$ en la optimización continua y luego se uso $\sigma = 0.01$ en la optimización discreta.

En total el experimento duró 24 horas con 30 minutos y en promedio cada simulación tomó 2.788 segundos. Esto se debió a que se utilizó una resolución de 100 nm . Pero, como el *bend* que definimos utilizaba un tamaño de píxel de 60 nm , las simulaciones no estaban logrando simular de forma correcta todos los detalles de los diseños.

En la figura 5.2 se muestra el diseño obtenido con este experimento, pero simulado con una resolución de $40nm$. Como se puede observar, hay una reducción a prácticamente la mitad de la transmitancia que se estaba obteniendo al usar una resolución de $100nm$.

5.2 Experimento 2

Con el anterior experimento se logró determinar la importancia de utilizar una adecuada resolución. Así, se procedió a trabajar con $40nm$ de resolución.

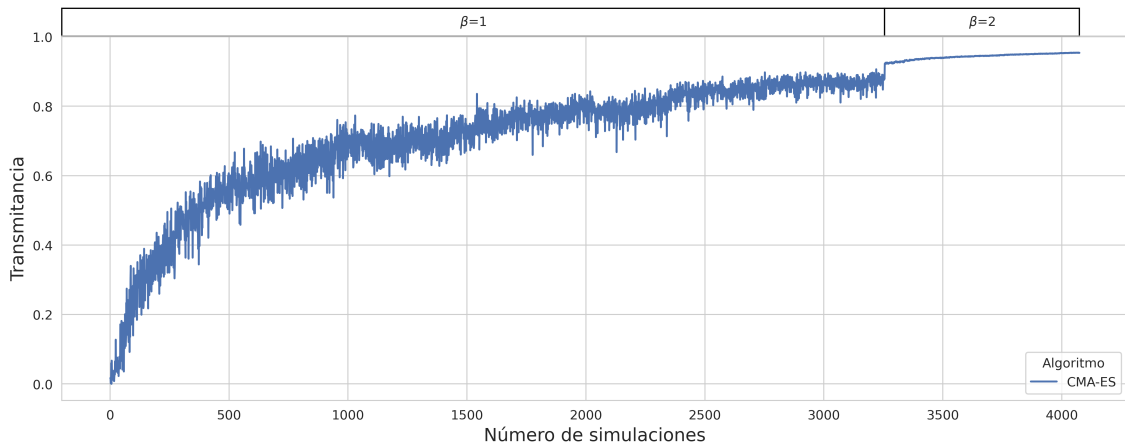


FIGURA 5.3: Gráfica del número de simulaciones vs transmitancia producto de optimizar un *bend* usando CMA-ES en el experimento 2.

En total el experimento duró 4 días con 16 horas y en promedio cada simulación tomó 60.543 segundos. Esto se debe al haber puesto la resolución en $40nm$.

Como se puede observar en la figura 5.3, la optimización continua se realizó por alrededor de 3200 simulaciones y solo se avanzó un poco en la optimización discreta. Esto se debe al tiempo que estaba tomando la optimización. Sin embargo, aún cuando se realizaron menos simulaciones se obtuvieron mejores resultados que en el experimento 1 y con un mayor grado de confianza debido a la resolución usada. En la figura 5.4 se puede corroborar lo destacable de estos resultados. Sin embargo, debido a la reducción en la cantidad de simulaciones, la optimización no logró llegar a fases con valores de β mayores

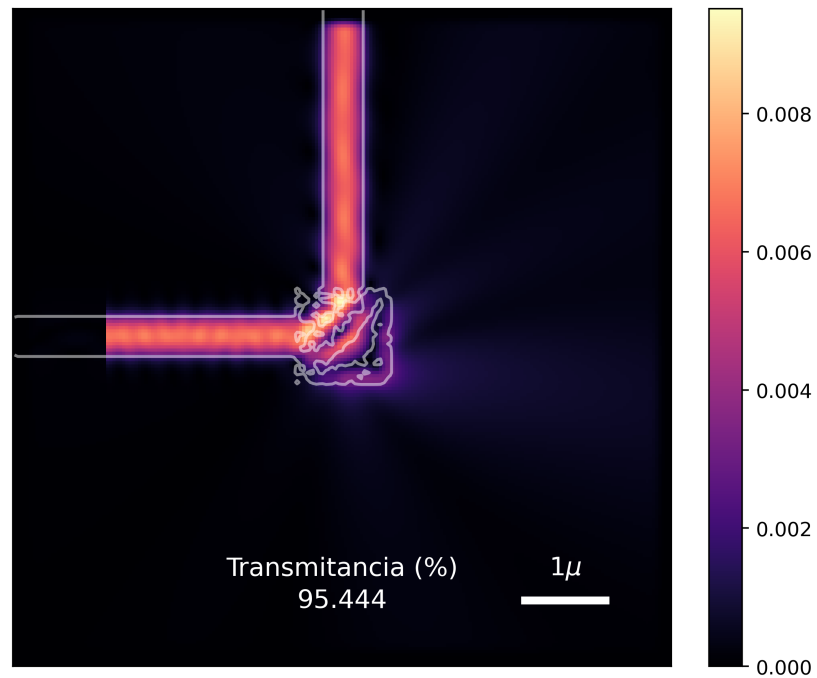


FIGURA 5.4: Intensidad de campo eléctrico y transmitancia del diseño obtenidos en el experimento 2 simulado bajo una resolución de 40 nm .

a 2, por ello los diseños aún no han convergido a valores reales. Esto se puede apreciar en la figura 5.5 donde las regiones grises representan la permitividad de un material no conocido.

Al resultado de este experimento se le sometió a una discretización forzada (acercar la permitividad de cada píxel a la del Si u SiO_2 de acuerdo a cual valor estuviera más cerca). El diseño resultante se muestra en la figura 5.6, se observa que hay una reducción en el valor de la transmitancia. A pesar de ello, los resultados son mejores que el experimento 1, esto nos lleva a suponer que de haber logrado ejecutarse la misma cantidad de iteraciones que en el experimento 1, se podría haber logrado un dispositivo discretizado con una transmitancia alrededor del 95 %.



FIGURA 5.5: Distribución de la permitividad eléctrica del diseño del experimento 2.

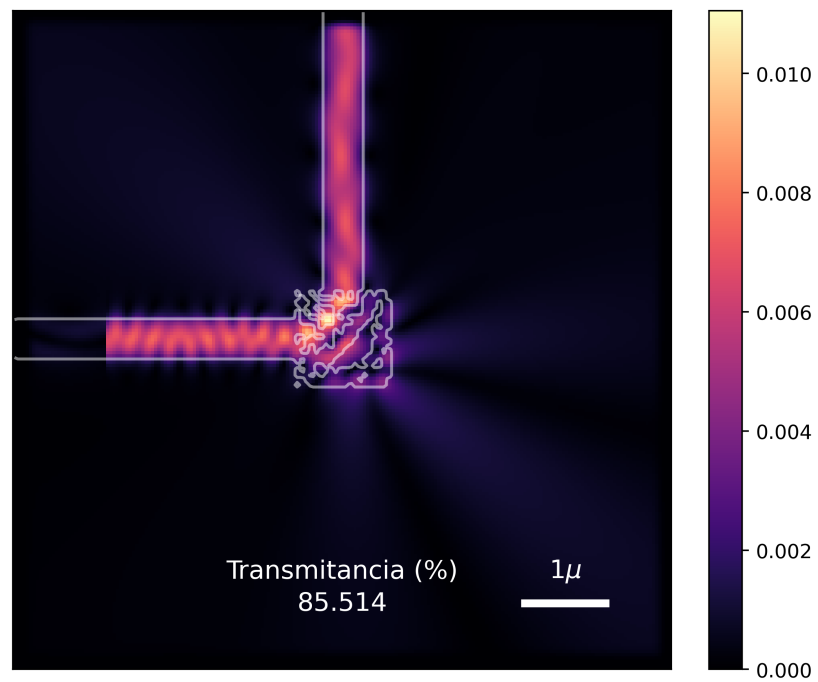


FIGURA 5.6: Intensidad de campo eléctrico y transmitancia del diseño obtenidos en el experimento 2 simulado bajo una resolución de 40 nm tras ser discretizado.

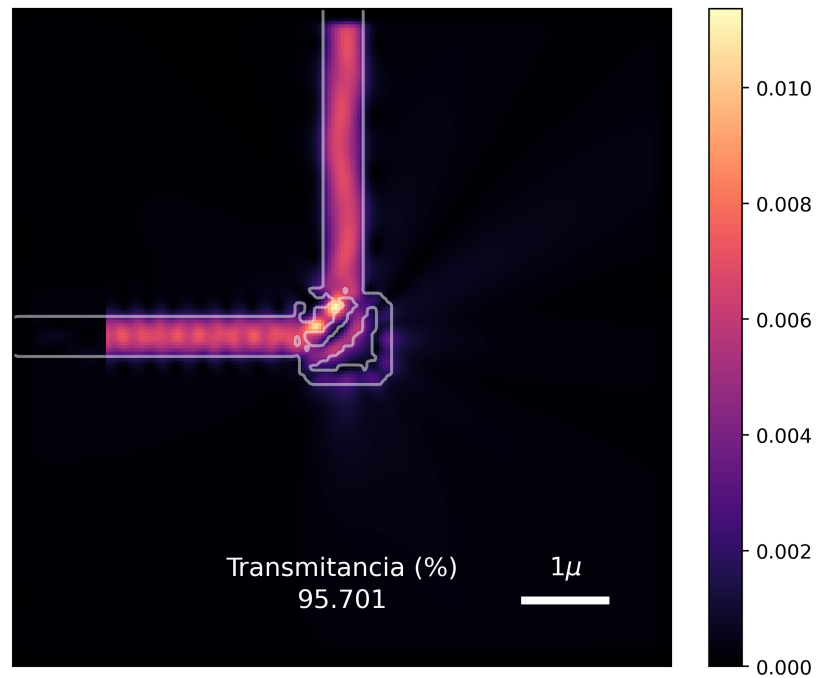


FIGURA 5.7: Intensidad de campo eléctrico y transmitancia del diseño obtenidos en el experimento 3 simulado bajo una resolución de 40 nm .

5.3 Experimento 3

Como tercer experimento se analizaron los resultados de las últimas evaluaciones realizadas en el anterior experimento. Producto de ello, se encontró un diseño que ya estaba prácticamente discretizado y que mantenía una transmitancia del 95.701 %, además que presentaba curvas suaves que presumiblemente no tendrían dificultades para ser fabricadas. Este diseño se puede apreciar en la figura 5.7.

A partir de los experimentos realizados se observa que se está logrando el objetivo de la tesis. Cuando se logre completar las siguientes etapas de la metodología, principalmente las simulaciones en 3D y restricciones de fabricación, parece razonable el suponer que se obtendrán resultados favorables.

REFERENCIAS BIBLIOGRÁFICAS

- Anderson, E., González, J., Gazman, A., Azevedo, R., and Bergman, K. (2018). Optically connected and reconfigurable gpu architecture for optimized peer-to-peer access. In *Proceedings of the International Symposium on Memory Systems, MEMSYS '18*, page 257–258, New York, NY, USA. Association for Computing Machinery.
- Bogaerts, W. and Chrostowski, L. (2018). Silicon photonics circuit design: Methods, tools and challenges. *Laser & Photonics Reviews*, 12(4):1700237.
- Elsawy, M. M., Lanteri, S., Duvigneau, R., Fan, J. A., and Genevet, P. (2020). Numerical Optimization Methods for Metasurfaces. *Laser and Photonics Reviews*, 14(10):1–17.
- Glick, M., Kimmerling, L. C., and Pfahl, R. C. (2018). A Roadmap for Integrated Photonics. *Optics and Photonics News*, 29(3):36.
- Goodfellow, I. J., Vinyals, O., and Saxe, A. M. (2015). Qualitatively characterizing neural network optimization problems. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*.
- Gregory, M. D., Martin, S. V., and Werner, D. H. (2015). Improved Electromagnetics Optimization. *IEEE Antennas and Propagation Magazine*, 57(june):48–59.
- Hammond, A. M., Oskooi, A., Johnson, S. G., and Ralph, S. E. (2020). Robust topology optimization of foundry-manufacturable photonic devices: An open-source fdtd toolbox. In *Frontiers in Optics / Laser Science*, page FTh1C.4. Optical Society of America.

- Hansen, N. (2016). The CMA Evolution Strategy: A Tutorial.
- Hohenester, U. (2020). *Nano and Quantum Optics: An Introduction to Basic Principles and Theory*. Graduate Texts in Physics. Springer International Publishing, 1st ed. 2020 edition.
- Hughes, T. W. and Fan, S. (2016). Plasmonic circuit theory for multiresonant light funneling to a single spatial hot spot. *Nano Letters*, 16(9):5764–5769.
- Kudyshev, Z. A., Kildishev, A. V., Shalaev, V. M., and Boltasseva, A. (2020). Machine learning-assisted global optimization of photonic devices. *Nanophotonics*, 10(1):371–383.
- Lukas Chrostowski (2010). *Silicon Photonics Design: From Device to System*.
- Ma, Y., Zhang, Y., Yang, S., Novack, A., Ding, R., Lim, A. E.-J., Lo, G.-Q., Baehr-Jones, T., and Hochberg, M. (2013). Ultralow loss single layer submicron silicon waveguide crossing for soi optical interconnect. *Opt. Express*, 21(24):29374–29382.
- Malheiros-Silveira, G. N. and Delalibera, F. G. (2020). Inverse design of photonic structures using an artificial bee colony algorithm. *Applied Optics*, 59(13):4171.
- Malkiel, I., Mrejen, M., Nagler, A., Arieli, U., Wolf, L., and Suchowski, H. (2018). Plasmonic nanostructure design and characterization via Deep Learning. *Light: Science and Applications*, 7(1).
- Molesky, S., Lin, Z., Piggott, A. Y., Jin, W., Vucković, J., and Rodriguez, A. W. (2018). Inverse design in nanophotonics. *Nature Photonics*, 12(11):659–670.
- Oskooi, A. F., Roundy, D., Ibanescu, M., Bermel, P., Joannopoulos, J. D., and Johnson, S. G. (2010). Meep: A flexible free-software package for electromagnetic simulations by the FDTD method. *Computer Physics Communications*, 181(3):687–702.

- Piggott, A. Y., Petykiewicz, J., Su, L., and Vučković, J. (2017). Fabrication-constrained nanophotonic inverse design. *Scientific Reports*, 7(1):1–7.
- Prosopio-Galarza, R., De La Cruz-Coronado, J., Hernandez-Figueroa, H. E., and Rubio-Noriega, R. (2019). Comparison between optimization techniques for Y-junction devices in SOI substrates. *Proceedings of the 2019 IEEE 26th International Conference on Electronics, Electrical Engineering and Computing, INTERCON 2019*, pages 1–4.
- Schneider, P. I., Garcia Santiago, X., Soltwisch, V., Hammerschmidt, M., Burger, S., and Rockstuhl, C. (2019). Benchmarking Five Global Optimization Approaches for Nano-optical Shape Optimization and Parameter Reconstruction. *ACS Photonics*, 6(11):2726–2733.
- Shen, Y., Harris, N. C., Skirlo, S., Prabhu, M., Baehr-Jones, T., Hochberg, M., Sun, X., Zhao, S., Larochelle, H., Englund, D., and Soljačić, M. (2017). Deep learning with coherent nanophotonic circuits. *Nature Photonics*, 11(7):441–446.
- Shen, Y., Meng, X., Cheng, Q., Rumley, S., Abrams, N., Gazman, A., Manzhosov, E., Glick, M. S., and Bergman, K. (2019). Silicon photonics for extreme scale systems. *J. Lightwave Technol.*, 37(2):245–259.
- Song, W. and Xie, K. (2008). Optimal design of a multi-mode interference splitter based on SOI. *Optoelectronics Letters*, 4(2):92–95.
- Su, L., Piggott, A. Y., Sapra, N. V., Petykiewicz, J., and Vučković, J. (2018). Inverse Design and Demonstration of a Compact on-Chip Narrowband Three-Channel Wavelength Demultiplexer. *ACS Photonics*, 5(2):301–305.
- Su, L., Vercruysse, D., Skarda, J., Sapra, N. V., Petykiewicz, J. A., and Vučković, J. (2020). Nanophotonic inverse design with SPINS: Software architecture and practical considerations. *Applied Physics Reviews*, 7(1).

- Vuckovic, J. (2019). From inverse design to implementation of practical (quantum) photonics (Conference Presentation). In Soci, C., Sheldon, M. T., and Agio, M., editors, *Quantum Nanophotonic Materials, Devices, and Systems 2019*, volume 11091. International Society for Optics and Photonics, SPIE.
- Yang, J. and Fan, J. A. (2017). Topology-optimized metasurfaces: impact of initial geometric layout. *Opt. Lett.*, 42(16):3161–3164.
- Zhang, J., Bi, S., and Zhang, G. (2021). A directional Gaussian smoothing optimization method for computational inverse design in nanophotonics. *Materials and Design*, 197:109213.