

UNIVERSIDAD DE INGENIERÍA Y TECNOLOGÍA

CARRERA DE CIENCIA DE LA COMPUTACIÓN



**Comparación de Algoritmos de Optimización usando
Computación de Alto Rendimiento en Dos Dispositivos
Nanofotónicos: *bend-90°* y *2-splitter***

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

22 de octubre de 2021

Índice general

	Pág.
CAPÍTULO 1 Motivación y Contexto	1
1.1 Introducción	2
1.2 Descripción del Problema	5
1.3 Justificación	6
1.4 Objetivos	7
1.5 Aportes	7
CAPÍTULO 2 Marco Teórico	8
2.1 Dispositivos de estudio	8
2.1.1 <i>Bend-90°</i>	8
2.1.2 <i>2-splitter</i>	10
2.2 Diseño inverso	10
2.2.1 Parametrización	10
2.2.1.1 Parametrización por píxeles	10
2.2.1.2 Parametrización por conjuntos de nivel	12
2.2.1.3 Parametrización por segmentos	13
2.2.2 Definición de la función objetivo	13
2.2.3 Simulación	14
2.2.4 Plan de optimización	15
2.2.5 Transformaciones	16

CAPÍTULO 3	Trabajos relacionados	17
3.1	Parametrización por píxeles	17
3.2	Parametrización por conjuntos de nivel	18
3.3	Parametrización por segmentos	18

Índice de tablas

2.1	Evaluación cualitativa de las librerías MEEP y SPINS	15
-----	--	----

Índice de figuras

1.1	Diseños tradicionales y obtenidos a partir de diseño inverso de un <i>bend-90°</i> y <i>2-splitter</i>	4
1.2	<i>Bend-90</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> . . .	5
2.1	Intensidad de campo eléctrico para un <i>bend-90°</i> de radio interno de $0.25 \mu m$	9
2.2	Parametrización por píxeles para un <i>bend-90°</i>	11
2.3	Parametrización por conjuntos de nivel para un <i>bend-90°</i>	12
2.4	Parametrización por 13 segmentos de un <i>2-splitter</i> . Imagen extraída de Prosopio-Galarza et al. (2019)	13

Capítulo 1

Motivación y Contexto

Durante los últimos años distintas áreas de la industria han mostrado interés en la fotónica por el elevado ancho de banda que ofrece en comunicaciones. Sin darnos cuenta, esta tecnología se ha ido integrando a nuestro día a día comenzando con el éxito de los cables ópticos. Probablemente, la creciente popularidad de esta área es debido a que el siguiente paso lógico para mejorar el rendimiento de los dispositivos electrónicos, los cuales están llegando a sus límites físicos, parece ser la incorporación o acondicionamiento de estos con dispositivos fotónicos ([Glick et al., 2018](#), [Lukas Chrostowski, 2010](#)). Sin embargo, la convergencia de estas tecnologías aún presenta muchos desafíos. Particularmente, existen dispositivos como el *bend-90°* y *2-splitter* que aún no logran aplicación industrial aún cuando son fundamentales para una variedad de circuitos fotónicos ([Molesky et al., 2018](#)).

La dificultad de encontrar aplicación industrial en parte se debe que al trabajar en la escala de nanómetros los diseños intuitivos han mostrado un bajo rendimiento y poca flexibilidad para incorporar restricciones de fabricación. Debido a ello se comenzó a aplicar el diseño inverso. Con esta metodología primero se define las propiedades deseadas de un dispositivo y luego se busca una geometría que las satisfaga. De esta manera se ha logrado obtener diseños no intuitivos pero con un buen rendimiento ([Su et al., 2020](#)).

El diseño inverso ha recibido mucha atención en fotónica durante los últimos 20 años ([Molesky et al., 2018](#)). Trabajos como los de [Su et al. \(2020\)](#) han logrado encontrar diseños de un *bend-90°* y *2-splitter* con un buen rendimiento de acuerdo a simulaciones. Luego, investigaciones como las de [Su et al. \(2018\)](#) y [Piggott et al. \(2017\)](#) intentan incorporar 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. Así, han surgido estudios como los de [Elsawy et al. \(2020\)](#), [Gregory et al. \(2015\)](#), [Schneider et al. \(2019\)](#) quienes buscan comparar 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, existe una carencia de estudios de comparación de algoritmos de optimización aplicados a un *bend-90°* y *2-splitter*. Es importante la realización de ellos pues cada dispositivo es una clase distinta de problema ([Molesky et al., 2018](#)). Así, el objetivo principal de este estudio es cubrir esta brecha evaluando el rendimiento y convergencia de algoritmos de optimización usados para estos dispositivos.

El presente trabajo 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 resolver el problema, define los objetivos y señala los aportes del trabajo.

El capítulo 2 desarrolla conceptos y terminología necesaria para entender las siguientes capítulos.

El capítulo 3 describe trabajos relacionados del estado del arte.

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 comenzando a lograr adaptar para fabricar circuitos fotónicos (Lukas Chrostowski, 2010). Tercero, el elevado ancho de banda que ofrece en comunicaciones digitales ha despertado el interés de cinco grandes sectores de producción: i) centrales de datos, ii) internet de las cosas, iii) sector automotriz, iv) sensores, v) fotónica RF (Glick et al., 2018, Lukas Chrostowski, 2010).

Los dispositivos fotónicos se utilizan en grandes cantidades en los circuitos fotónicos integrados (Lukas Chrostowski, 2010). Estos trabajan en la escala de nanómetros y son diseñados para funcionar bajo ciertas especificaciones. Así, para cumplir los requerimientos deseados existen dos estrategias comunes: diseño tradicional y diseño inverso (Molesky et al., 2018).

En el diseño tradicional se define el dispositivo con geometrías simples que permiten obtener funciones analíticas de sus propiedades físicas. 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-90°* y *2-splitter* que han presentado 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.



(a) *Bend-90°* con diseño tradicional



(b) *Bend-90°* obtenido con diseño inverso. Extraído de ([Su et al., 2020](#))



(c) *2-splitter* con diseño tradicional



(d) *2-splitter*° 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-90°* y *2-splitter*

1.2 Descripción del Problema

Para poder calcular las propiedades físicas de un dispositivo (e.g. campo eléctrico, transmitancia) se debe resolver las ecuaciones de Maxwell. Así, con el objetivo de evaluar las propiedades de cualquier geometría se suele utilizar métodos numéricos como elementos finitos (FEM) y diferencias finitas en el dominio de tiempo (FDTD) (Schneider et al., 2019). Con estos planteamientos se selecciona una región rectangular a optimizar y se la divide en $n \times m$ píxeles como si fuera una imagen, ver Figure 1.2. Luego, a cada píxel se le asocia el número 0 o 1. Típicamente, cero representa la presencia de SiO_2 en la ubicación del píxel y uno la presencia de Si (Molesky et al., 2018).

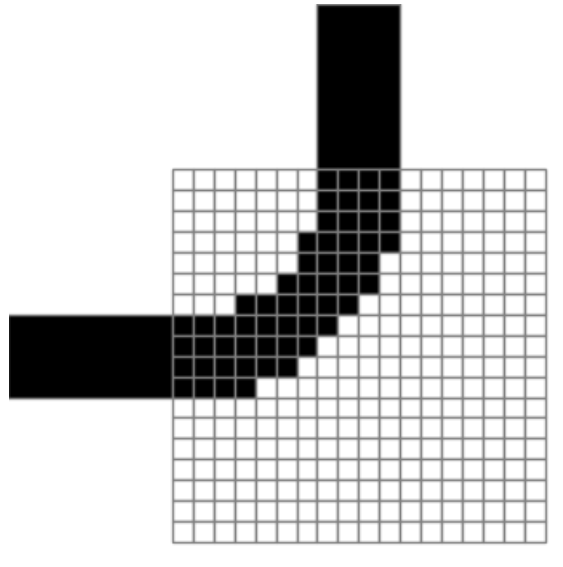


FIGURA 1.2: *Bend-90* con una región de diseño discretizada en 18×18 píxeles. Cada píxel negro representa la presencia de Si y cada píxel blanco de SiO_2

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 diseños.

Por otro lado, [Su et al. \(2020\)](#) con el objetivo de fabricar un dispositivo con buen rendimiento, configuró un espacio de búsqueda de $2^{34 \times 34}$ opciones. Así, se presentan algunas dificultades con el diseño inverso:

1. Es imposible evaluar todas los posibles diseños.
2. Las simulaciones computacionales son muy costosas ([Kudyshev et al., 2020](#)).
3. El espacio de búsqueda es altamente no convexo ([Su et al., 2018](#)).
4. No todos los diseños son fabricables ([Su et al., 2020](#)).
5. Cada dispositivo es una clase distinta de problema ([Molesky et al., 2018](#)).

De este modo, existe una demanda crítica de un *framework* capaz de optimizar dispositivos con un elevado número de parámetros dentro de un espacio de búsqueda no convexo ([Kudyshev et al., 2020](#)). Este es un problema muy grande, por ello en la presente tesis nos centraremos en optimizar un *bend-90°* y un *2-splitter*.

1.3 Justificación

Deste el punto de vista industrial, el *bend-90°* y *2-splitter* se han escogido debido a que han sido estudiados usando diseño inverso desde el 2004, pero aún no han encontrado aplicación masiva en la industria ([Molesky et al., 2018](#)).

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

- Evaluar y comparar el rendimiento y la convergencia de algoritmos de optimización usadas para optimizar un *bend-90°* y *2-splitter* usando computación de alto rendimiento.
- Fabricar el diseño con mejor rendimiento que se obtenga del *bend-90°* y del *2-splitter* para poder comparar las simulaciones computacionales con las mediciones físicas.

1.5 Aportes

Este trabajo busca brindar una comparativa de las técnicas de optimización más relevantes que se aplican para optimizar un *bend-90°* y un *2-splitter* 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 dos secciones. En primer lugar, se describe las propiedades físicas de interés de un *bend-90°* y *2-splitter*. En segundo lugar, se desarrolla en cinco pasos como utilizar el diseño inverso para optimizar los dispositivos de estudio. En el primer paso se especifica tres estrategias comúnmente usadas para parametrizar la región de diseño. En el segundo paso se muestra como definir la función objetivo para un *bend-90°* y *2-splitter*. En el tercer paso se señala los programas donde se puede simular el comportamiento de los diseños que desarrollemos. En el cuarto paso se exponen como se puede preparar un plan de optimización de un dispositivo fotónico. En el quinto paso se da a conocer transformaciones que se pueden incluir al plan de optimización para incorporar restricciones de fabricación.

2.1 Dispositivos de estudio

2.1.1 *Bend-90°*

Un *bend-90°* es un dispositivo fotónico que se encarga de guiar un haz de ondas para que gire 90 grados.

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-90°* representado por color azul oscuro. Así, se visualiza de forma gráfica que el dispositivo está teniendo considerables pérdidas de energía lo cual es un indicador de un mal rendimiento.

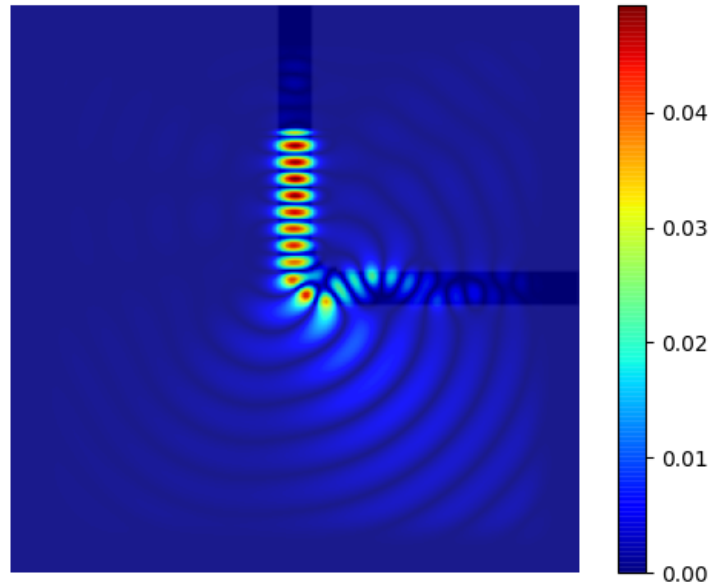


FIGURA 2.1: Intensidad de campo eléctrico para un *bend-90°* de radio interno de $0.25 \mu m$

Por otro lado, para evaluar el desempeño de un *bend-90°* de forma numérica se suele calcular la transmitancia como la relación entre la intensidad del rayo que sale del dispositivo con la intensidad con la que entra. Esto se expresa mediante la ecuación 2.2:

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

2.1.2 2-splitter

Un *2-splitter* 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-90°* se estudia su campo eléctrico y su transmitancia para medir el rendimiento de estos dispositivos.

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 por píxeles

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 por píxeles 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.

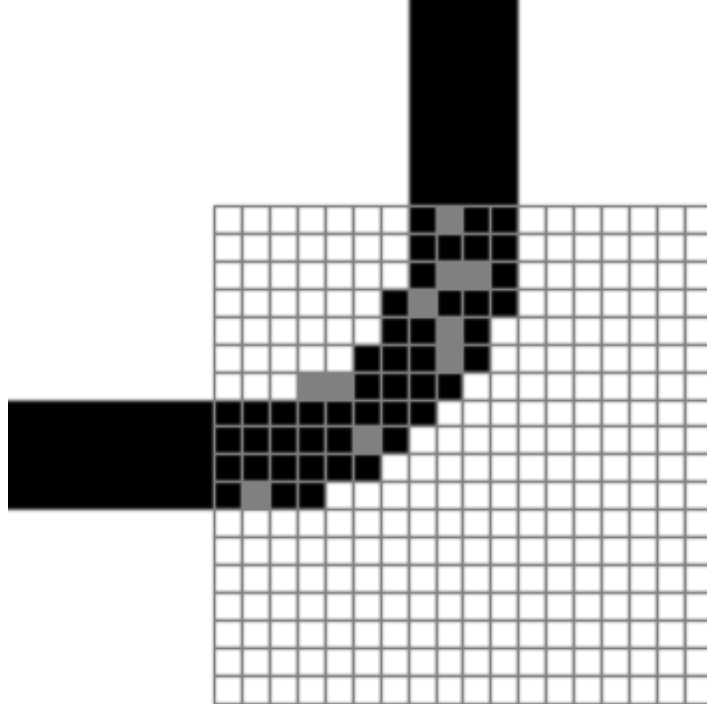


FIGURA 2.2: Parametrización por píxeles para un *bend-90°*

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.

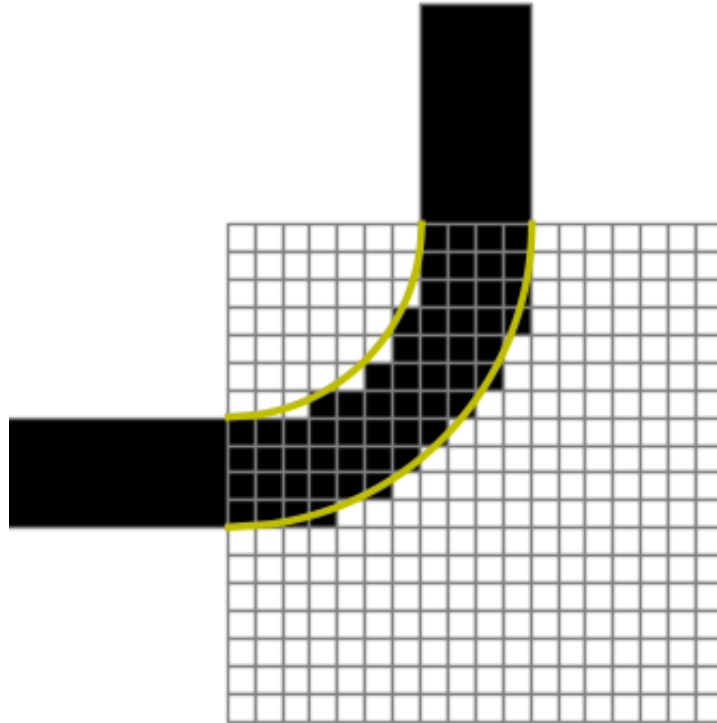


FIGURA 2.3: Parametrización por conjuntos de nivel para un *bend-90°*

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. En la figura 2.4 podemos observar como [Prosopio-Galarza et al. \(2019\)](#) utiliza esta parametrización con 13 segmentos para diseñar un 2 – *splitter*.

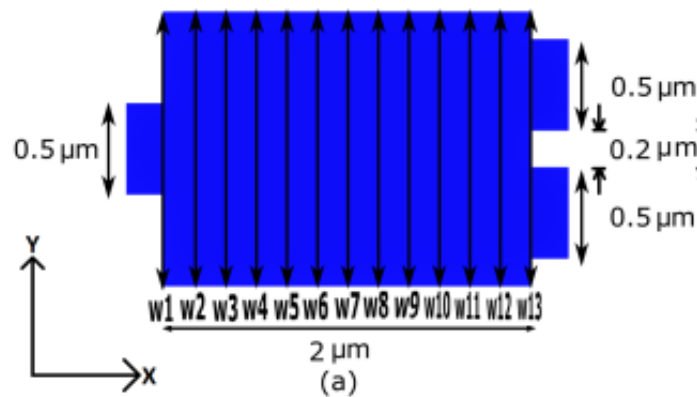


FIGURA 2.4: Parametrización por 13 segmentos de un 2-*splitter*. Imagen extraída de [Prosopio-Galarza et al. \(2019\)](#)

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 por píxeles 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 función 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-90°*

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

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

2. *2-splitter*

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 representa que se 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

2.2.4 Plan de optimización

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

1. Parametrización por píxeles

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 segmentos. 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.
- 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 por píxeles, una forma se 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. 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.

Capítulo 3

Trabajos relacionados

En el presente capítulo se resume los puntos principales de trabajos relacionados al problema de investigación. Para ello dividiremos el contenido en tres secciones. En primer lugar, mostraremos trabajos que han usado parametrización por píxeles. En segundo lugar, veremos investigaciones que utilizan parametrización por conjuntos de nivel. En tercer lugar, describiremos publicaciones que emplearon parametrización por segmentos.

3.1 Parametrización por píxeles

[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 *bend-90°*. La definición de su función objetivo es como la descrita en el marco teórico, 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 $L - BFGS - B$ y MMA 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 5 GPU, 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.

3.2 Parametrización por conjuntos de nivel

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.

3.3 Parametrización por segmentos

Prosopio-Galarza et al. (2019) utiliza esta estrategia 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 *2-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*.

REFERENCIAS BIBLIOGRÁFICAS

- 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.
- 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.
- 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*.

- 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.
- 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).
- 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.