

UNIVERSIDAD TECNOLÓGICA DE LA MIXTECA

SOFTWARE PARA LA EVALUACIÓN DE SISTEMAS ÓPTICOS
FORMADOS POR ESPEJOS CÓNCAVOS CÓNICOS USANDO
LA PRUEBA DE RONCHI Y ALGORITMOS GENÉTICOS

TESIS
PARA OBTENER EL TÍTULO DE:
INGENIERO EN FÍSICA APLICADA

PRESENTA:
JORGE LUIS OROPEZA ARELLANO

DIRECTOR DE TESIS:
DR. JORGE GONZÁLEZ GARCÍA
CO-DIRECTOR:
M. T. C. A. MOISÉS EMMANUEL RAMÍREZ GUZMÁN

HUAJUAPAN DE LEÓN, OAXACA
AGOSTO 2024

Agradecimientos

Este trabajo no habría sido posible sin el incondicional apoyo de mis padres, Rosa Arellano Ríos y Moisés Oropeza Espinosa. Su amor y dedicación han sido el pilar fundamental a lo largo de esta carrera. Agradezco profundamente que hayan creído en mí incluso en los momentos más difíciles, cuando las dudas y desafíos parecían insuperables. Su confianza y aliento constante me han dado la fuerza y la determinación necesarias para seguir adelante. Este logro no solo refleja mi esfuerzo y dedicación, sino también el sacrificio y el amor que ustedes han invertido en mí.

A mis hermanos: Alexis, César, Jaqueline y Alan, quiero expresar mi sincero agradecimiento por su constante apoyo y conexión inquebrantable. Siempre han estado dispuestos a extender su mano para ayudarme y, a su vez, saben que siempre pueden contar conmigo. Su apoyo ha sido invaluable y su presencia en mi vida es un verdadero tesoro. Este logro también es un reflejo del amor y la solidaridad que compartimos como familia. Gracias por estar siempre ahí para mí, en cada paso del camino.

Agradezco de corazón a mi familia, a mis abuelas, padrinos, tías y tíos, quienes siempre han creído en mí y me han brindado su apoyo cuando más lo necesitaba. Gracias por esos momentos, consejos y conversaciones que me han ayudado a ser quien soy hoy.

Quiero expresar mi más profundo agradecimiento al Dr. Jorge González García y al M.T.C.A. Moisés E. Ramírez Guzmán, mis directores a lo largo de este trabajo. Su comprensión, paciencia y orientación fueron fundamentales para que este proyecto alcanzara la relevancia que tiene hoy y también para que yo creciera profesionalmente. Gracias a sus valiosas contribuciones, este trabajo no solo enriquece mi formación, sino que también aporta conocimientos significativos para la sociedad.

A mis revisores de tesis, la Dra. Denisse Barreiro, el Dr. Ulises Ramírez y el Dr. Agustín Santiago, les agradezco profundamente por sus valiosos consejos y por las críticas constructivas que ayudaron a mejorar este trabajo. Su orientación enriqueció el contenido de mi tesis y también amplió mi perspectiva sobre futuras investigaciones. Gracias a sus aportaciones, me siento más preparado y motivado para continuar explorando y desarrollando mis ideas en el campo.

Expreso mi gratitud a todos los profesores tanto del que, a lo largo de mi formación, contribuyeron a mi desarrollo académico y personal. Cada clase, cada consejo y cada desafío planteado por ustedes fue fundamental para construir las bases de mi conocimiento y despertar mi curiosidad científica, en especial a Dr. Maxwell Jimenez, Dr. Salomón González, Dr. Hugo D. Sánchez, Dr. Rafael Martínez y el M. Arturo P. Sandoval. Sus enseñanzas han dejado una huella imborrable en mi formación.

A mis amigos y compañeros de carrera, les agradezco sinceramente por su apoyo constante, por las horas compartidas de estudio, y por los momentos de diversión y desafío. Agradezco especialmente a Tachu, Roger, Frida, Carreño, Rubí, Meri, Iris, Juanita, Jeeni, Anita, Pipo, Cristian,

Sanvas, Yeyo, y a todos los demás, cuyo apoyo ha hecho que este camino sea mucho más llevadero y enriquecedor. Nunca me sentí solo durante esta etapa gracias a su compañía. Un agradecimiento especial a J. Francisco Canseco por su amistad y compañerismo; sin su ayuda y aliento, no estaría escribiendo estas líneas.

También quiero agradecer a mis amigos Luis, Tachi, Proco, Daniel, Yara, Alejandra y Noriega, por la amistad que hemos compartido durante tantos años. Su presencia constante y su apoyo incondicional han sido fundamentales en este camino que he recorrido durante esta etapa. Gracias por estar siempre ahí, por cada palabra de aliento, y por acompañarme en este viaje con la misma lealtad y alegría de siempre.

Índice general

1. Introducción	15
1.1. Resumen	15
1.2. Planteamiento del problema	15
1.3. Justificación	16
1.4. Hipótesis	17
1.5. Objetivos	18
1.5.1. Objetivo general	18
1.5.2. Objetivos específicos	18
1.6. Metas	18
1.7. Limitaciones de la tesis	18
2. Óptica geométrica	21
2.1. Introducción	21
2.2. Sistemas ópticos	21
2.3. Frente de onda	23
2.4. Espejos	25
2.4.1. Espejos esféricos	26
2.4.2. Espejos esféricos	26
2.5. Superficies ópticas	28
3. Aberraciones	31
3.1. Introducción	31
3.2. Aberraciones del frente de onda. Aberraciones primarias	31
3.3. Polinomios de aberración del frente de onda.	37
3.4. Polinomios de Zernike.	39
3.5. Aberraciones del frente de onda usando los polinomios de Zernike.	40
4. Evaluación de espejos	45
4.1. Introducción	45
4.2. Pruebas ópticas no interferométricas	45
4.3. Prueba de Ronchi	48
4.3.1. Patrones de Ronchi para superficies esféricas	51
4.4. Birronchigramas.	52

5. Algoritmos evolutivos	53
5.1. Introducción	53
5.2. Optimización	54
5.3. Algoritmos genéticos	56
5.3.1. Terminología	58
5.3.2. Elementos que conforman un Algoritmo Genético	60
5.4. Estrategias evolutivas.	67
5.4.1. Mutación	67
5.4.2. Técnicas avanzadas de adaptación	69
5.4.3. Adaptación de tamaños de paso basados en Frecuencia de éxitos.	70
5.4.4. Estrategia Evolutiva (1+1)	70
5.4.5. Control de covarianzas	71
5.4.6. Estrategia Evolutiva (1+1)-Cholesky-CMA	72
6. Desarrollo del software basado en la prueba de Ronchi para la evaluación de espejos.	75
6.1. Generalidades	75
6.1.1. Introducción	75
6.1.2. Arquitectura general del software	75
6.1.3. Lenguajes y herramientas utilizadas	76
6.1.4. Consideraciones de eficiencia y rendimiento.	78
6.2. Simulación de Birronchigramas.	78
6.3. Programación del método de normalización	80
6.4. Programación del método de correlación de imágenes (Birronchigramas)	85
6.5. Programación del método de optimización	87
6.5.1. Algoritmo Genético	87
6.5.2. Estrategia Evolutiva (1+1)	88
6.5.3. Estrategia (1+1)-Cholesky-CMA	89
6.5.4. Algoritmo de optimización híbrido	90
6.6. Programa integral de evaluación de espejos	91
7. Pruebas de eficiencia	93
7.1. Introducción	93
7.2. Proceso de eficiencia del algoritmo híbrido para el caso 1.	94
7.2.1. Método de selección	94
7.2.2. Tasa de mutación variable	95
7.2.3. Número de individuos y número de generaciones óptimos	98
7.2.4. Optimización de la estrategia evolutiva (1+1).	99
7.2.5. Optimización de la estrategia evolutiva (1+1)-Cholesky-CMA	99
7.2.6. Pruebas del algoritmo híbrido	103
7.3. Proceso de eficiencia del algoritmo híbrido para el caso 2.	104
7.3.1. Parámetros óptimos en algoritmo genético de valores reales	104
7.3.2. Optimización de la estrategia evolutiva (1+1)	105
7.3.3. Pruebas con la estrategia evolutiva (1+1)-Cholesky-CMA	105
7.3.4. Pruebas del algoritmo híbrido	108
7.3.5. Pruebas en rangos de búsqueda.	110

8. Resultados	113
8.1. Caso ideal	115
8.2. Caso experimental	117
8.3. Birronchigrama 1	120
8.4. Birronchigrama 2	120
8.5. Birronchigrama 3	122
9. Discusión de Resultados	125
10. Conclusiones	127

Índice de figuras

1.1.	Arreglo experimental para aplicar la prueba de Ronchi en un espejo cóncavo.	16
1.2.	Patrones de Ronchi generados por diferentes espejos.	16
2.1.	Esquema de un sistema óptico que muestra el paso del rayo principal a través de él.	22
2.2.	Fenómenos de reflexión y refracción.	23
2.3.	Refracción de un rayo que pasa a través de una superficie óptica esférica.	23
2.4.	Rayos de luz y frentes de onda en un medio isotrópico.	24
2.5.	Ondas esféricas que son captados por la pupila de entrada y mapeados en la pupila de salida.	25
2.6.	Vistas en plano bidimensional y tridimensional de un espejo cóncavo y la incidencia de una onda plana en él.	26
2.7.	Esquema de ondas de fuente puntual hacia un espejo esférico y focos conjugados en un espejo cóncavo.	27
2.8.	Representación de varias superficies cónicas y superficies generadas.	29
2.9.	Representación de varias superficies cónicas de acuerdo a su constante de conicidad.	30
3.1.	Esquema de referencia de un sistema óptico.	32
3.2.	Errores de enfoque transversal y longitudinal.	33
3.3.	Descripción gráfica de las aberraciones transversal y longitudinal.	34
3.4.	Astigmatismo: Análisis de la intensidad de rayos incidentes y perfil astigmático.	35
3.5.	Aberración de curvatura de campo en una lente mostrado con ejemplos.	36
3.6.	Proyección de una imagen de acuerdo al tipo de aberración de distorsión.	36
3.7.	Localización de la aberración de Coma de una fuente puntual monocromática.	37
3.8.	Coordenadas polares para el rayo en la pupila de entrada de un sistema óptico.	38
3.9.	Gráficos de superficie de los polinomios de Zernike hasta el orden 10.	43
4.1.	Descripción de la prueba de Hartmann aplicada a una superficie cóncava.	46
4.2.	Descripción de un sensor Shack-Hartmann.	47
4.3.	Descripción de la prueba de la navaja con un ejemplo práctico.	47
4.4.	Sombras obtenidas al realizar la prueba de la alambra en una superficie cóncava.	48
4.5.	Ronchigramas de algunas deformaciones de superficies típicas.	49
4.6.	Diagrama y configuración experimental de la prueba de Ronchi	50
4.7.	Frente de onda W y orientación de la rejilla de Ronchi.	51
4.8.	Rejillas cuadradas para obtener birronchigramas con la prueba de Ronchi.	52
4.9.	Birronchigramas experimentales de diferentes tipos de espejos.	52
5.1.	Ubicación de óptimos globales óptimos locales en un ejemplo de maximización.	56

5.2.	Descripción de óptimos cercanos en un problema de maximización 1-dimensional.	57
5.3.	Ejemplo práctico de la selección natural.	58
5.4.	Descripción gráfica del método de selección por ruleta.	61
5.5.	Ejemplificación del método de selección por rango.	63
5.6.	Dos técnicas populares de cruza en los algoritmos genéticos.	64
5.7.	Diagrama que muestra el funcionamiento del operador mutación.	66
5.8.	Diagrama de flujo de un algoritmo genético simple.	66
5.9.	Elipsoides de mutación que operan dentro del espacio de búsqueda.	68
6.1.	Arquitectura general del software para evaluación de espejos cóncavos.	76
6.2.	Geometría de la prueba de Ronchi usando una rejilla cuadrículada.	78
6.3.	Diagrama de flujo que muestra el proceso de la simulación de un ronchigrama o un birronchigrama.	81
6.4.	Esquema de imágenes que muestran la simulación de birronchigramas con diferentes parámetros.	82
6.5.	Diagrama de flujo que muestra el proceso de normalización de la intensidad de la imagen del patrón del ronchigrama o birronchigrama.	84
6.6.	Esquema de imágenes que muestran la imagen del birronchigrama o ronchigrama experimental (lado izquierdo), y sus respectivas normalizaciones (lado derecho).	86
6.7.	Diagrama de flujo que describe el algoritmo de la estrategia evolutiva (1+1).	89
6.8.	Diagrama de flujo que representa la estrategia evolutiva (1+1)-Cholesky-CMA (caso de minimización).	90
6.9.	Diagrama de flujo de la programación del algoritmo híbrido.	91
6.10.	Diagrama de flujo del software de evaluación de espejos.	92
7.1.	Superficie generada junto con los coeficientes y las variables del lado derecho.	94
7.2.	Comportamiento del método de selección por ruleta. Tasa de mutación fija.	95
7.3.	Gráfico del comportamiento del método de selección por rango. Tasa de mutación fija.	96
7.4.	Gráfico del comportamiento del método de selección por torneo. Tasa de mutación fija.	96
7.5.	Gráfico del comportamiento del método de selección por ruleta. Tasa de mutación variable.	97
7.6.	Gráfico del comportamiento del método de selección por rango. Tasa de mutación variable.	97
7.7.	Gráfico del comportamiento del método de selección por torneo. Tasa de mutación variable.	98
7.8.	Gráficos que muestran el comportamiento del algoritmo genético de valores reales.	100
7.9.	Gráficos que muestran el comportamiento de la estrategia evolutiva (1+1).	101
7.10.	Gráficos que muestran la convergencia de la estrategia evolutiva (1+1)-Cholesky-CMA con diferentes tamaños de paso inicial.	102
7.11.	Imágenes comparativas entre los datos experimentales y los generados con ayuda del algoritmo híbrido que forman la superficie de un molde para lentes elásticas.	103
7.12.	Pruebas de convergencia y rendimiento del algoritmo genético de valores reales.	106
7.13.	Pruebas realizadas para la maximización con la estrategia evolutiva (1 + 1).	107

7.14. Pruebas realizadas para la maximización con la estrategia evolutiva (1+1)-Cholesky-CMA.	109
7.15. Pruebas del algoritmo híbrido cambiando el tamaño del espacio de búsqueda.	111
8.1. Birronchigrama experimental, normalización y evolución de la simulación de un birronchigrama usando el software general de espejos.	114
8.2. Comparación de birronchigramas simulados.	116
8.3. Resultados obtenidos al procesar el birronchigrama 1 con $k = 0$	121
8.4. Resultados al procesar el birronchigrama 1 con $K = -1$	121
8.5. Resultados obtenidos al procesar el birronchigrama 2 con $K = 0$	122
8.6. Resultados obtenidos al procesar el birronchigrama 2 con $K = -1$	122
8.7. Resultados obtenidos al procesar el birronchigrama 3 con $K = 0$	123
8.8. Resultados obtenidos al procesar el birronchigrama 3 con $K = -1$	124

Índice de Tablas

2.1.	Tabla de constantes de conicidad.	29
3.1.	Primeros 28 términos de los polinomios ortonormales circulares de Zernike $Z_j(\rho, \theta)$.	41
3.2.	Polinomios ortonormales de Zernike hasta de cuarto orden en coordenadas rectangulares.	42
5.1.	Selección por ruleta. Valores de ejemplo.	61
5.2.	Tabla de valores donde se evalúa y selecciona una población por medio del método de rango.	62
6.1.	Tabla que muestra los 28 primeros polinomios de zernike, utilizados en el proceso de normalización.	83
6.2.	Tabla comparativa que muestra 3 ejemplos de coeficientes de correlación que fueron evaluados con la función de aptitud programada.	87
7.1.	Valores límite para el espacio de búsqueda en la prueba de maximización.	94
7.2.	Tabla comparativa de los coeficientes publicados y los hallados con el algoritmo híbrido en el caso 1.	104
7.3.	Tabla comparativa para análisis de resultados del algoritmo híbrido.	110
8.1.	Configuración de parámetros utilizados en el algoritmo híbrido para las pruebas con birronchigramas.	113
8.2.	Valores de coeficientes de correlación y sus correspondientes coeficientes de Zernike de la evolución en la optimización de un birronchigrama.	115
8.3.	Tabla de información de parámetros utilizados para evaluar los espejos bajo prueba.	115
8.4.	Coefficientes obtenidos al ejecutar el software de evaluación de espejos con el birronchigrama parab_49R.	117
8.5.	Coefficientes obtenidos al ejecutar el software de evaluación de espejos con el birronchigrama parab_13R.	118
8.6.	Coefficientes obtenidos al ejecutar el software de evaluación de espejos con el birronchigrama parab_27R.	119
8.7.	Coefficientes obtenidos al ejecutar el software de evaluación de espejos con el birronchigrama 1.	120
8.8.	Coefficientes obtenidos al ejecutar el software de evaluación de espejos con el birronchigrama 2.	123
8.9.	Coefficientes obtenidos al ejecutar el software de evaluación de espejos con el birronchigrama 3.	124

Capítulo 1

Introducción

1.1. Resumen

Para la fabricación de lentes, espejos y planos ópticos, además de los diversos métodos de pulido de superficies empleados para la generación de las superficies deseadas de dichos elementos, se requiere probar si la calidad de las superficies generadas, están dentro de los parámetros deseados, a esto se le denomina calidad óptica o pruebas ópticas. Para la determinación de este parámetro se emplean métodos de prueba durante el método de pulido de dichos dispositivos. En el caso particular de la fabricación de espejos cóncavos, además de las pruebas interferométricas, se aplica la prueba conocida como la “prueba de Ronchi”, mediante la cual se miden las aberraciones del frente de onda generados por el espejo.

Para la cuantificación de las aberraciones del frente de onda mediante la prueba de Ronchi, se requiere de un software de evaluación que integre procesamiento digital de imágenes, junto con un método de normalización y un método de optimización, que en su conjunto permitirán realizar una evaluación precisa del espejo.

1.2. Planteamiento del problema

El planteamiento del problema correspondiente a la presente tesis inicia desde la aplicación de la prueba de Ronchi. Para aplicar esta prueba, en este caso se requiere de un dispositivo llamado probador de Ronchi, el cual contiene un led rojo y una base en donde se coloca frente al led una rejilla de líneas claras y oscuras llamada rejilla de Ronchi. Las líneas proyectadas por el led de la rejilla van hacia la superficie del espejo bajo prueba; en la Figura 1.1 se puede observar el arreglo experimental de la prueba de Ronchi utilizado para la cuantificación de las aberraciones de Kingslake [1]. El espejo refleja la proyección de las franjas hacia una abertura colocada arriba del led donde puede observarse directamente con el ojo la imagen generada por el espejo de las franjas proyectadas de la rejilla, esta imagen conlleva la información de las deformaciones del frente de onda generadas por el espejo. La imagen reflejada por el espejo es la imagen de las franjas proyectadas de la rejilla deformadas. La forma o deformación de las franjas determinan la forma de la superficie del espejo, así como sus deformaciones que están ligadas con las deformaciones del frente de onda. A la imagen de las franjas deformadas se le llama patrón de Ronchi; en la Figura 1.2 se muestran algunos de estos patrones.

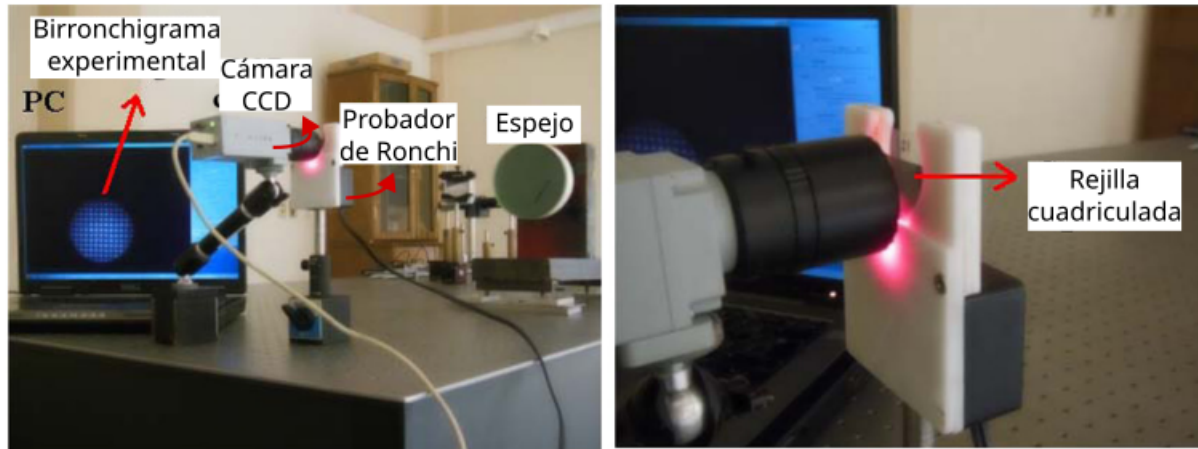


Figura 1.1: a) Arreglo experimental de la prueba de Ronchi aplicada a un espejo cóncavo cónico; b) acercamiento del probador de Ronchi y del detector (Cámara CCD). El experimento fue montado en el laboratorio de óptica de la UTM [2].

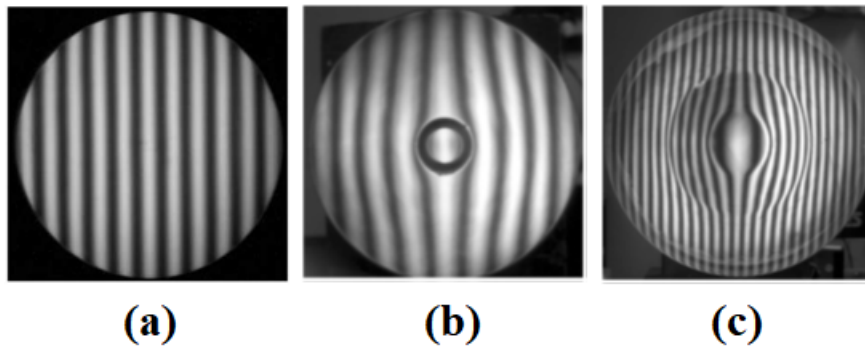


Figura 1.2: Patrones de Ronchi generados por diferentes espejos. a) Espejo esférico, b) espejo elíptico, y (c) espejo esférico en la zona radial externa y parabólico en la zona central [2].

Por lo tanto, el problema consiste en que a partir del patrón de Ronchi, aplicándole un conjunto de procedimientos en forma secuencial, los cuales deben ser programados, incluyendo en sí misma la prueba de Ronchi, (simulación del patrón de Ronchi a partir de parámetros del espejo), se deben calcular las aberraciones del frente de onda, con las cuales queda caracterizado el espejo, es decir, se cuantifica la calidad óptica del mismo. Este conjunto de procedimientos aplicados a la imagen del patrón de Ronchi [1, 2] serán descritos de forma general en el capítulo 6.

1.3. Justificación

Las lentes y los espejos, como dispositivos ópticos, forman parte de muchos desarrollos tecnológicos, en el caso específico de los espejos cóncavos, éstos son dispositivos que forman parte de instrumentos que se usan en diversos ámbitos como son: en el área de la enseñanza para medir la velocidad de la luz, en el cual son ocupados dos espejos [3], en el área de la visión en un robot con capacidad de visión omnidireccional [4] y como sensor omnidireccional catadióptrico, el cual consiste de una cámara y un espejo, dicho sensor le permite al robot rastrear objetos [5]; en la

aplicación del fútbol robótico, los espejos cónicos y esféricos son usados como parte del sistema visión panorámica de 360° [6]; en los sistemas de láseres de descarga de gas, los espejos cónicos son usados como espejos resonadores en el rango de terahertz, su uso permite mejorar la estabilidad de la radiación, simplificar el diseño del láser y reducir la complejidad de su mantenimiento [7]; recientemente se desarrolló un sistema holográfico digital panorámico capaz de obtener información 3D de un objeto cuasicilíndrico mediante el uso de un espejo cónico. El sistema holográfico digital panorámico propuesto es capaz de escanear toda la superficie del objeto para determinar la amplitud y la fase simultáneamente [8]. En el área de la astronomía, propiamente hablando, y aplicada a la enseñanza, los espejos cónicos forman parte del instrumento conocido como telescopio astronómico, el cual es usado para la observación y estudio de los cuerpos celestes [9]. El telescopio, además de ser un instrumento que ha permitido el desarrollo de la astronomía, también es usado como un instrumento de enseñanza y motivación para el estudio de las ciencias exactas. Un ejemplo de esto es el proyecto conocido como “Del universo al aula”, dicho proyecto consistió en que un gran número de secundarias públicas del estado de Puebla y de Oaxaca construyeran un telescopio tipo Newtoniano cuyo espejo primario (cónico parabólico cóncavo) fue fabricado (pulido) y proporcionado por el Laboratorio de Pruebas Ópticas (LPO) del posgrado en Física Aplicada de la Facultad de Ciencias Físico Matemáticas de la BUAP [9], por lo que el LPO tuvo que realizar una producción en serie de estos espejos primarios.

Para la medición de la calidad óptica de los espejos se aplicó una serie de programas desarrollados con los procedimientos matemáticos necesarios para aplicar la prueba óptica conocida como la prueba de Ronchi [2, 9]. Los procedimientos matemáticos aplicados son: 1) métodos de normalización de la irradiancia de patrones de franjas, 2) procesamiento digital de imágenes (método de correlación), 3) métodos de optimización (algoritmos genéticos) y 4) el desarrollo de software de la prueba de Ronchi. La mayoría de estos procedimientos fueron desarrollados como programas independientes, careciendo del desarrollo de un software único eficiente y diseñado para otorgar resultados dentro de lo que se considera “en tiempo real”, beneficiando a la comunidad científica, académica y la sociedad en general.

Por todo lo descrito anteriormente, se justifica el desarrollo de un software eficiente que utilice el mínimo tiempo de cómputo para aplicar la prueba óptica de Ronchi para la medición de la calidad de los espejos cónicos cóncavos, los cuales son aplicados en diversos desarrollos tecnológicos, dentro de los cuales se tienen desarrollos muy actuales como lo muestra [8]. Una de las contribuciones del presente tema de tesis para la cuantificación de la función de aberración del frente de onda, es utilizar los polinomios de Zernike en lugar de los polinomios de Kingslake utilizados en [1].

1.4. Hipótesis

Usando el lenguaje de programación en Python y el método de optimización de algoritmos evolutivos, se desarrollará un software de evaluación de espejos ópticos cóncavos eficiente, que consuma el mínimo tiempo de cómputo, usando la prueba óptica de Ronchi.

1.5. Objetivos

1.5.1. Objetivo general

Desarrollar un programa de evaluación a partir de la prueba óptica de Ronchi para evaluar espejos cóncavos cónicos mediante la cuantificación de las aberraciones de la superficie de espejos cóncavos, usando algoritmos evolutivos para incrementar la eficiencia del proceso de optimización.

1.5.2. Objetivos específicos

- Desarrollar un programa de simulación de la prueba óptica de Ronchi en el lenguaje de programación de Python.
- Realizar los programas de las etapas de los métodos de normalización y correlación en el lenguaje de programación de Python.
- Desarrollar un programa de la etapa de optimización para la evaluación de espejos basado en algoritmos genéticos, programado de manera eficiente en Python.
- Acoplar todas las etapas programadas para generar el software de la evaluación de espejos basado en la prueba de Ronchi.
- Realizar pruebas de eficiencia del software de pruebas ópticas generado para la evaluación de espejos a partir de patrones de Ronchi experimentales de espejos ya conocidos.

1.6. Metas

1. Desarrollo de la teoría necesaria del área de la óptica para el entendimiento y manejo de la prueba óptica de Ronchi.
2. Programa en Python para la simulación de la prueba de Ronchi en sus dos modalidades: simulación de ronchigramas y bironchigramas.
3. Desarrollo y reporte de la teoría necesaria de algoritmos evolutivos como método de optimización.
4. Programa en Python para la aplicación del método de optimización por algoritmos genéticos aplicando la prueba de Ronchi.
5. Programa integral para la evaluación de espejos usando algoritmos genéticos, el método de correlación y la prueba de Ronchi con Python.

1.7. Limitaciones de la tesis

Es importante tener en cuenta que esta tesis se centra específicamente en la evaluación de las aberraciones en la superficie de los espejos de perfil cónico, lo que implica que otros aspectos relacionados con la evaluación de espejos con otras simetrías podrían no ser abordados en profundidad.

Sin embargo, esta delimitación permite un enfoque más preciso y detallado en el objetivo principal de esta tesis, proporcionando así una base sólida para futuros estudios que puedan explorar otros aspectos en este ámbito.

Capítulo 2

Óptica geométrica

2.1. Introducción

La óptica es la ciencia que estudia los orígenes, la propagación y la detección de la luz. En esta definición se entiende por luz no solo la radiación electromagnética visible, sino también la infrarroja y la ultravioleta [10]. La superficie de un objeto luminiscente o que está iluminado externamente se comporta como si constara de un gran número de fuentes puntuales de luz, cada una emitiendo ondas esféricas, es decir, los rayos emanan radialmente en la dirección en la que fluye la energía [11].

En la mayoría de los casos, la función de un sistema óptico es recoger y remodelar una parte del frente de onda incidente, a menudo, con la intención esencial de formar una imagen de un objeto. Observe que los sistemas realizables se caracterizan por no poder recibir toda la luz emitida, pues un sistema acepta solo un segmento del frente de onda. Como resultado, siempre habrá una desviación aparente de la propagación rectilínea incluso en medios homogéneos, las ondas serán difractadas. El grado de perfección alcanzable de un sistema óptico estará limitado por la difracción. A medida que la longitud de onda de la energía radiante disminuye en comparación con las dimensiones físicas del sistema óptico, los efectos de la difracción cobran menos importancia. En el límite cuando la longitud de onda tiende a 0, en los medios homogéneos se produce una propagación rectilínea, definiéndose así el campo ideal de la óptica geométrica. En este caso, ya no sería posible observar el comportamiento específicamente atribuido a la naturaleza ondulatoria de la luz (por ejemplo, la interferencia y la difracción). En muchas situaciones, la gran simplicidad de la aproximación de la óptica geométrica compensa abundantemente su falta de precisión. Resumiendo, la óptica geométrica estudia la manipulación controlada de los frentes de onda (o rayos) por medio de la interposición de cuerpos reflectantes y/o refractantes, despreciando cualquier efecto de difracción [11].

2.2. Sistemas ópticos

El propósito de prácticamente todos los sistemas ópticos de formación de imágenes, es resolver un objeto de tamaño mínimo especificado en un campo de visión deseado. El campo de visión se expresa como la extensión espacial o angular en el espacio del objeto, y el objeto de tamaño mínimo es el elemento de resolución más pequeño que se requiere para identificar o comprender la imagen

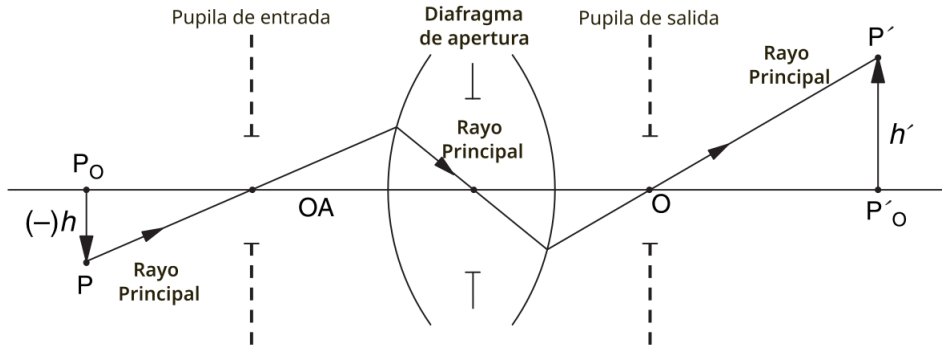


Figura 2.1: Esquema de un sistema óptico. Se muestran la entrada del rayo principal que sale de la imagen del objeto y pasa a través del sistema óptico para formar la imagen [13].

[12].

El diafragma real o lente que con toda intención limita la extensión del haz se llama iris o pupila real. En muchos sistemas, como por ejemplo en los telescopios refractores, el iris es la primera superficie del objetivo, pero en un sistema en general puede ser cualquier lente o diafragma [10]. La pupila de entrada es la imagen de la apertura vista desde la parte frontal del sistema óptico. La imagen de la apertura en el espacio de la imagen se denomina pupila de salida [12]. La Figura 2.1 muestra un esquemático que ejemplifica la idea anterior. Se observa el acceso del rayo principal a través de la pupila de entrada. Este rayo es emitido por un objeto a una altura h en el plano ortogonal al eje óptico. El punto P' es la localización de la imagen del objeto formada a una altura h' .

El principio de Fermat, del cual se basa la óptica geométrica, plantea que un rayo luminoso va de un punto a otro a lo largo de la trayectoria que le toma el menor o mayor tiempo posible, es decir, debe ser extremo o estacionario con respecto al de otras trayectorias. Utilizando este principio las leyes de la reflexión y la refracción son deducidas [10]:

- **Reflexión.** La primera ley de la reflexión dice que el rayo incidente, el rayo reflejado y la normal a la superficie reflectora están en un plano común. La segunda ley dice que la magnitud del ángulo de reflexión es igual a la magnitud del ángulo de incidencia, como lo muestra la figura 2.2a, es decir:

$$\theta_i = \theta_r, \quad (2.1)$$

- **Refracción.** La primera ley de la refracción dice que el rayo incidente, el rayo refractado y la normal a la superficie refractora están en un plano común. Esta ley es también una consecuencia inmediata del principio de Fermat.

La segunda ley, llamada también ley de Snell, se puede deducir con ayuda del camino óptico recorrido por un rayo incidente como se muestra en la Figura 2.2b, y también con el principio de Fermat para concluir que:

$$n_i \sin \theta_i = n_t \sin \theta_t. \quad (2.2)$$

Se infiere que un rayo que entra en un medio de mayor índice de refracción se dobla hacia la normal, y que por el contrario, al entrar en un medio con índice más bajo, el rayo, en lugar de avanzar recto, se doblará alejándose de la normal [11].

La región paraxial de un sistema óptico es una región estrecha cercana al eje óptico, donde los ángulos de los vértices que forman los rayos en la normal a la superficie, los ángulos de incidencia y

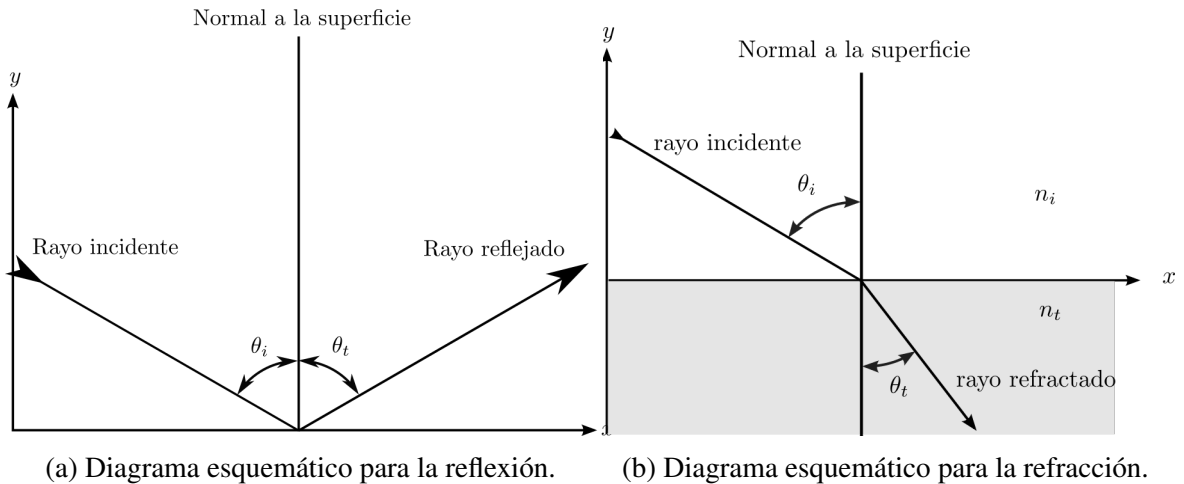


Figura 2.2: Fenómenos de reflexión y refracción.

de refracción se pueden aproximar por sus senos y tangentes. Por ejemplo, se observan en la Figura 2.3 a una superficie óptica esférica refractora que separa a dos medios en donde un rayo incidente forma estos ángulos con respecto a la normal M . Aunque esta región es pequeña, los cálculos basados en relaciones paraxiales son muy útiles debido a su simplicidad y eficacia. La mayoría de los sistemas ópticos prácticos forman imágenes de buena calidad, y las relaciones paraxiales proporcionan una aproximación excelente para describir la formación de imágenes en sistemas bien corregidos [14].

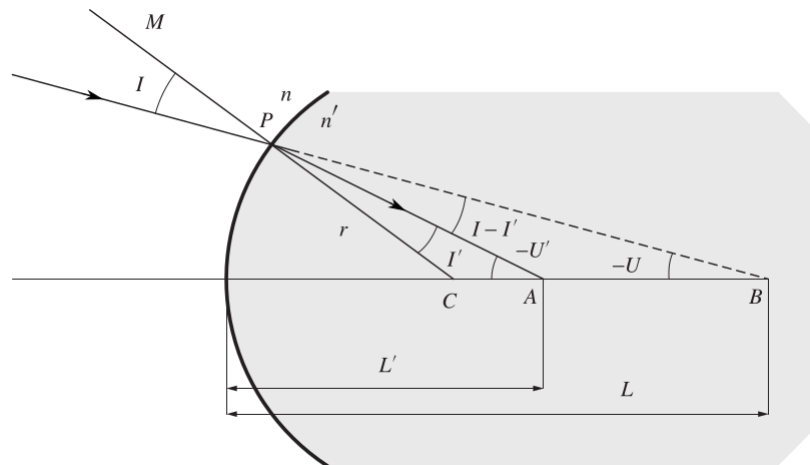


Figura 2.3: Esquema que muestra la refracción de un rayo a través de una superficie óptica esférica [10].

2.3. Frente de onda

Un frente de onda en un haz de luz es una superficie en el espacio en la cual todos los puntos tienen la misma fase instantánea. Otra definición equivalente describe que, un frente de onda es

una superficie de longitud de camino óptico constante a lo largo de la trayectoria de la luz desde un punto luminoso en el objeto. Por lo tanto, podemos imaginar en una onda de luz una serie de superficies en las cuales su perturbación alcanza su máximo en cierto momento; es decir, las crestas de las ondas de luz. Esas superficies son los frentes de onda y la distancia entre dos frentes de onda consecutivos es la longitud de onda λ [15], como se observa en la Figura 2.4.

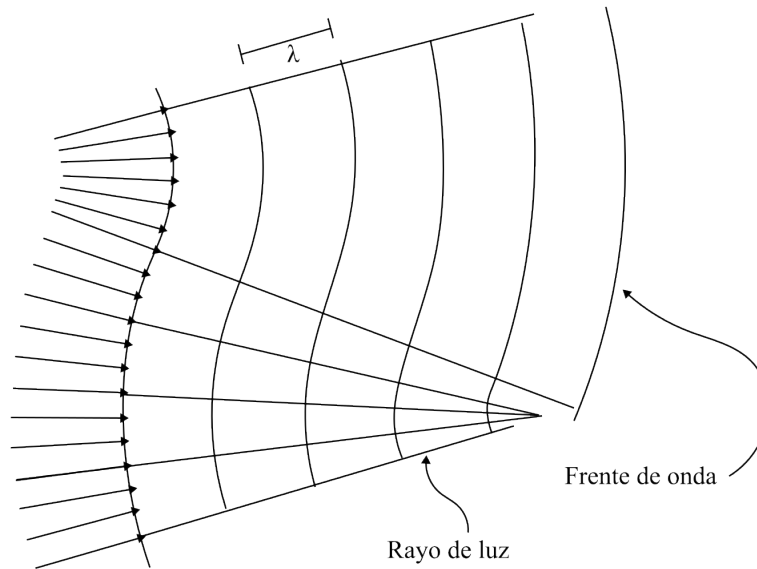


Figura 2.4: Rayos de luz y frentes de onda en un medio isotrópico [15].

Una fuente puntual de luz emite ondas radialmente divergentes, formando en su conjunto un frente de onda esférico, propagándose hacia el exterior desde la ubicación de la fuente. Para un objeto extendido, cada punto en el objeto actúa como una fuente puntual. La intensidad de cada uno de estos puntos puede variar para representar partes brillantes y oscuras en el objeto. Las ondas se propagan hacia el exterior desde la fuente puntual generando un frente de onda esférico perfecto, hasta que interactúan con algún objeto en el entorno. La pupila de entrada de un sistema óptico captura una parte de estos frentes de onda. Dado que la pupila tiene un tamaño finito, recorta la extensión del frente de onda esférico, permitiendo solo que una parte ingrese al sistema óptico. Los frentes de onda recortados que ingresan a la pupila de entrada se mapean en la pupila de salida. En un sistema de imagen ideal, los frentes de onda emergentes de la pupila de salida serían perfectamente esféricos y convergerían hacia un punto ideal en el plano de imagen. Para la colección de puntos que componen un objeto, el sistema de imágenes ideal asignaría cada uno de estos puntos a un frente de onda esférico perfecto con amplitud uniforme en la pupila de salida. El origen de cada una de estas ondas esféricas reside en el plano de la imagen predicho por la ecuación de imagen gaussiana [16].

La Figura 2.5 muestra ondas esféricas desde un punto axial y un punto en el borde del objeto cuando entran en la pupila de entrada E . El objeto está ubicado a una distancia z del plano principal frontal P . La ubicación del plano de la imagen O' a una distancia z' desde el plano principal trasero, P' se determina a partir de la ecuación de imagen gaussiana. Las ondas esféricas recortadas que entran en la pupila de entrada se asignan a la pupila de salida.

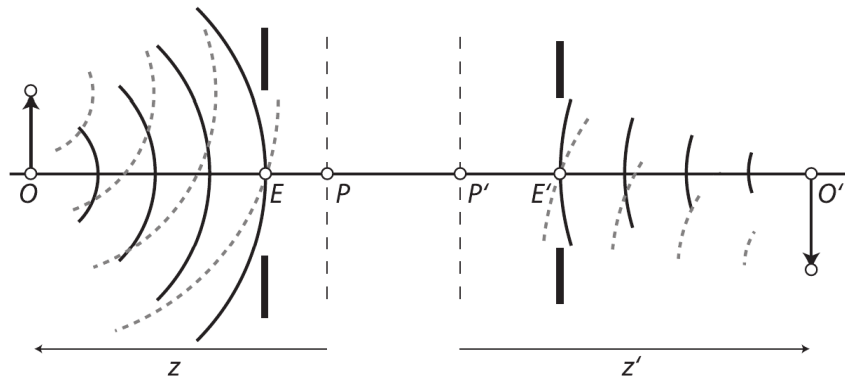


Figura 2.5: Ondas esféricas de diferentes puntos en el objeto que son capturados por la pupila de entrada y mapeados en la pupila de salida [16].

Hay una variedad de efectos ópticos que hacen que los frentes de onda que emergen de la pupila de salida se desvíen de la forma esférica deseada. Primero, la difracción provoca un cambio en la forma del frente de onda cuando éste interactúa con el tope de apertura. Este efecto es el límite fundamental a la capacidad de crear la onda esférica perfecta. En segundo lugar, es posible que los objetos no sean perfectamente planos, por lo que no se asignarán a una imagen plana. A menudo los objetos son tridimensionales y la imagen correspondiente reside en un volumen tridimensional en las proximidades del plano de la imagen gaussiana. El plano de imagen gaussiano captura la proyección de esta imagen tridimensional sobre su superficie bidimensional. Los puntos del objeto pueden enfocarse hacia un punto en el espacio de la imagen, pero este punto no residirá en el plano de la imagen gaussiana. La intersección de los frentes de onda de dichos puntos con el plano de la imagen gaussiana da como resultado un punto de tamaño finito. En tercer lugar, los elementos ópticos que componen el sistema no pueden, en general, preservar la forma esférica del frente de onda tras la refracción o reflexión. A medida que el frente de onda esférico incidente pasa a través de cada elemento, su forma sufre una aberración. Uno de los objetivos del diseñador de lentes es minimizar la desviación neta de estos frentes de onda de su forma esférica ideal equilibrando las aberraciones causadas por múltiples superficies entre sí. Finalmente, hay una variedad de efectos secundarios que contribuyen a la desviación de los frentes de onda esféricos. Estos efectos incluyen: dispersión de la luz, falta de homogeneidad y cambios térmicos en los materiales ópticos, y errores de fabricación y alineación de las superficies ópticas [16].

2.4. Espejos

Los sistemas de espejos se utilizan cada vez más, especialmente en las regiones del espectro de rayos X, ultravioleta e infrarrojos. Aunque es relativamente sencillo construir un dispositivo reflectante que funcione satisfactoriamente en un rango amplio de frecuencias, lo mismo no puede decirse de los sistemas refractantes [11].

Un espejo puede ser simplemente un trozo de vidrio negro o una superficie de metal finamente pulida. En el pasado, los espejos solían fabricarse recubriendo el vidrio con plata, que se elegía debido a su alta eficiencia en el ultravioleta e infrarrojo. Los recubrimientos evaporados al vacío de aluminio sobre sustratos altamente pulidos se han convertido en el estándar aceptado para espejos

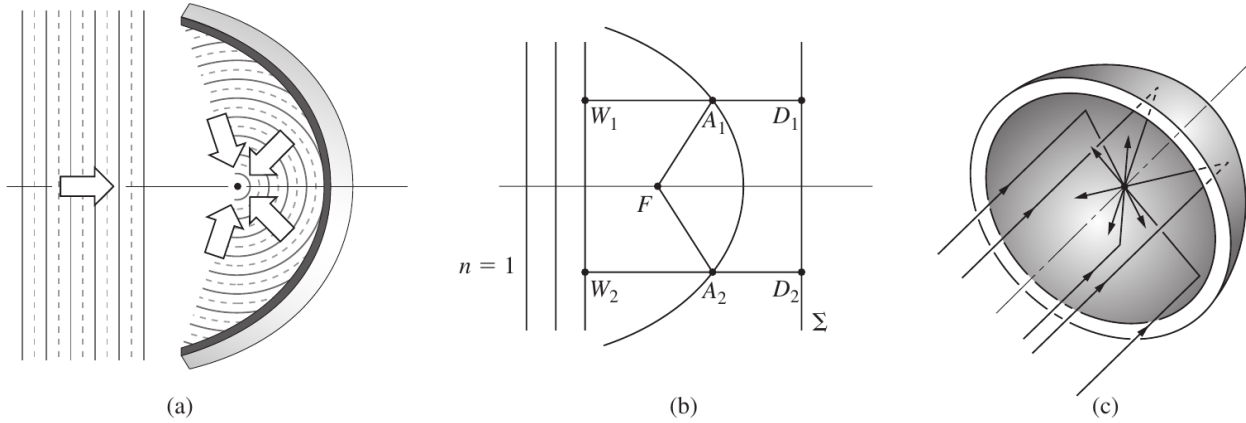


Figura 2.6: (a) Esquema que muestra un espejo cóncavo al que una onda plana incidente se reforma bajo reflexión en una onda esférica convergente. (b) Vista en 2D de la misma onda incidente en un espejo. (c) Vista tridimensional de la misma onda incidente en un espejo esférico [11].

de calidad. A menudo, se aplican recubrimientos protectores de monóxido de silicio o fluoruro de magnesio sobre el aluminio. En aplicaciones especiales (por ejemplo, en láseres), donde no se pueden tolerar ni siquiera las pequeñas pérdidas debido a las superficies de metal, los espejos formados por películas dieléctricas multicapa son indispensables [11].

2.4.1. Espejos esféricos

La base fundamental para el diseño de espejos es el uso del principio de Fermat aplicado a la formación de imágenes en sistemas refractores. Por lo tanto, se determina la configuración que debe tener un espejo si una onda plana incidente debe reformarse mediante reflexión en una onda esférica convergente. Dado que la onda plana finalmente debe converger en el punto F, las longitudes de trayectoria óptica para todos los rayos deben ser iguales; en consecuencia, para los puntos arbitrarios A_1 y A_2 , la longitud de camino óptico (OPL) está dado como [11]:

$$OPL = \overline{W_1A_1} + \overline{A_1F} = \overline{W_2A_2} + \overline{A_2F}, \quad (2.3)$$

tal como como se puede apreciar en la Figura 2.6. Ya que el plano Σ es paralelo a los frentes de onda incidentes tenemos,

$$\overline{W_1A_1} + \overline{A_1D_1} = \overline{W_2A_2} + \overline{A_2D_2}. \quad (2.4)$$

La ecuación 2.3 se satisface para una superficie la cual $\overline{A_1F} = \overline{A_1D_1}$ y $\overline{A_2F} = \overline{A_2D_2}$ o de manera general, una superficie para la cual $\overline{AF} = \overline{AD}$ para cualquier punto A en el espejo. De forma general, $\overline{AF} = e(\overline{AD})$, donde e es la excentricidad de una sección cónica. Los rayos podrían igualmente ser invertidos es decir, una fuente puntual en el foco de un paraboloide resultaría en la emisión de ondas planas desde el sistema [11].

2.4.2. Espejos esféricos

La ecuación paraxial que relaciona los puntos conjugados del objeto y la imagen con los parámetros físicos de un espejo esférico se puede derivar con la ayuda de la Figura 2.7 [11]. Ya

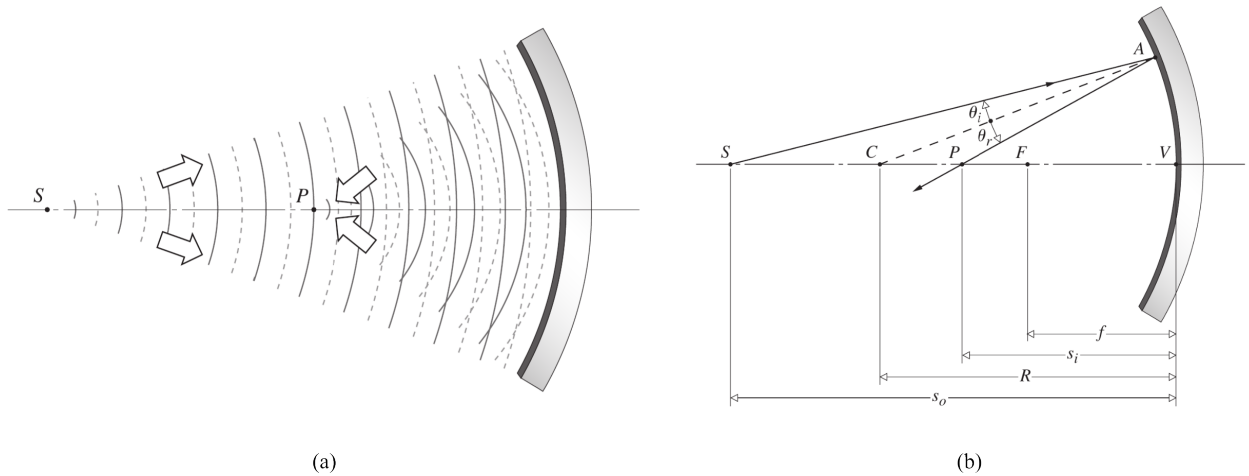


Figura 2.7: (a): Esquema que muestra la dirección de las ondas de una fuente puntual esférica hacia un espejo esférico. (b) Focos conjugados de un espejo cóncavo esférico [11].

que $\theta_i = \theta_r$, el ángulo SAP está dividido por \overline{CA} por consiguiente, divide el lado \overline{SP} del triángulo SAP en segmentos proporcionales a los dos lados restantes, esto es

$$\frac{\overline{SC}}{\overline{SA}} = \frac{\overline{CP}}{\overline{PA}}, \quad (2.5)$$

Además

$$\overline{SC} = s_o - |R| \quad \text{y} \quad \overline{CP} = |R| - s_i,$$

donde s_o (distancia al objeto) y s_i (distancia a la imagen) están a la izquierda siendo, por lo tanto, positivas. R será negativa porque C se halla a la izquierda de V , es decir, la superficie es cóncava. Entonces $|R| = -R$ y

$$\overline{SC} = s_o + R \quad \text{y} \quad \overline{CP} = -(s_i + R).$$

En la región paraxial $\overline{SA} \approx s_o$, $\overline{PA} \approx s_i$, y así la ecuación 2.5 queda definida como:

$$\frac{s_o + R}{s_o} = -\frac{s_i + R}{s_i},$$

o bien

$$\frac{1}{s_o} + \frac{1}{s_i} = -\frac{2}{R}, \quad (2.6)$$

que se denomina la fórmula de los espejos. Es aplicable tanto a espejos cóncavos ($R < 0$) como convexos ($R > 0$). El foco objeto o primario se define por

$$\lim_{s_i \rightarrow \infty} s_o = f_o,$$

y el foco secundario o foco imagen corresponde a

$$\lim_{s_o \rightarrow \infty} s_i = f_i.$$

Por consiguiente, de la ecuación 2.6 se tiene:

$$\frac{1}{f_o} + \frac{1}{\infty} = \frac{1}{\infty} + \frac{1}{f_i} = -\frac{2}{R},$$

lo que implica

$$f_o = f_i = -\frac{R}{2}.$$

Quitando los subíndices en las distancias focales, tenemos

$$\frac{1}{s_o} + \frac{1}{s_i} = \frac{1}{f}, \quad (2.7)$$

f será positiva para espejos cóncavos ($R < 0$) y negativa para espejos convexos ($R > 0$). En éste último, la imagen se forma detrás del espejo y es virtual.

2.5. Superficies ópticas

Los sistemas ópticos tradicionales emplean elementos refractivos y reflectantes para mover la luz del espacio del objeto al espacio de la imagen. Su forma determina la refracción de los rayos cuando pasan a través de las superficies frontal y posterior de la lente. Las superficies reflectantes suelen estar hechas de un sustrato metálico o de vidrio revestido. La forma de la superficie recubierta determina la dirección en la que los rayos se reflejan desde la superficie. Para controlar sistemáticamente la refracción y la reflexión de los rayos, las superficies tienen una forma prescrita y deben variar uniformemente hasta muy por debajo de la longitud de onda de la luz. El incumplimiento de este requisito de homogeneidad introduce dispersión y entrada de luz desde otras fuentes no deseadas hacia el sistema. Según estas observaciones, la forma de la superficie y el material del que está hecha son fundamentales para el rendimiento del sistema óptico [16].

Para el círculo, cualquier eje que pase por su centro es un eje de simetría. Si el círculo se gira alrededor de un eje horizontal, la superficie tridimensional resultante es una esfera. Las superficies ópticas esféricas suelen ser una sección de la esfera. A diferencia de los círculos, las elipses tienen solo dos ejes de simetría correspondientes a los ejes mayor y menor. Si se gira una elipse alrededor de su eje mayor, se forman superficies elipsoidales prolatos. Al girar una elipse alrededor de su eje menor, se crean de manera similar superficies elipsoidales oblatos. La simetría de parábolas e hipérbolas se reduce aún más a un solo eje que pasa por el vértice y el foco de la sección cónica. La rotación alrededor de este eje crea paraboloides e hiperboloides [16]. La Figura 2.8 muestra las secciones cónicas y las superficies generadas al hacer girar estas secciones en sus ejes de simetría.

Las superficies esféricas son muy importantes en los sistemas ópticos. Los de mayor interés son las superficies de simetría rotacional, las cuales están definidas por la siguiente relación, tomando a z como eje de revolución [17]:

$$z = \frac{cS^2}{1 + [1 - (K + 1)c^2S^2]^{1/2}} + A_1S^4 + A_2S^6 + A_3S^8 + A_4S^{10}, \quad (2.8)$$

donde $S^2 = x^2 + y^2$, $c = 1/r$, y r es el radio de curvatura. También A_1, A_2, A_3 y A_4 son constantes de deformación esféricas, K representa la función de excentricidad de una superficie cónica ($K = -e^2$),

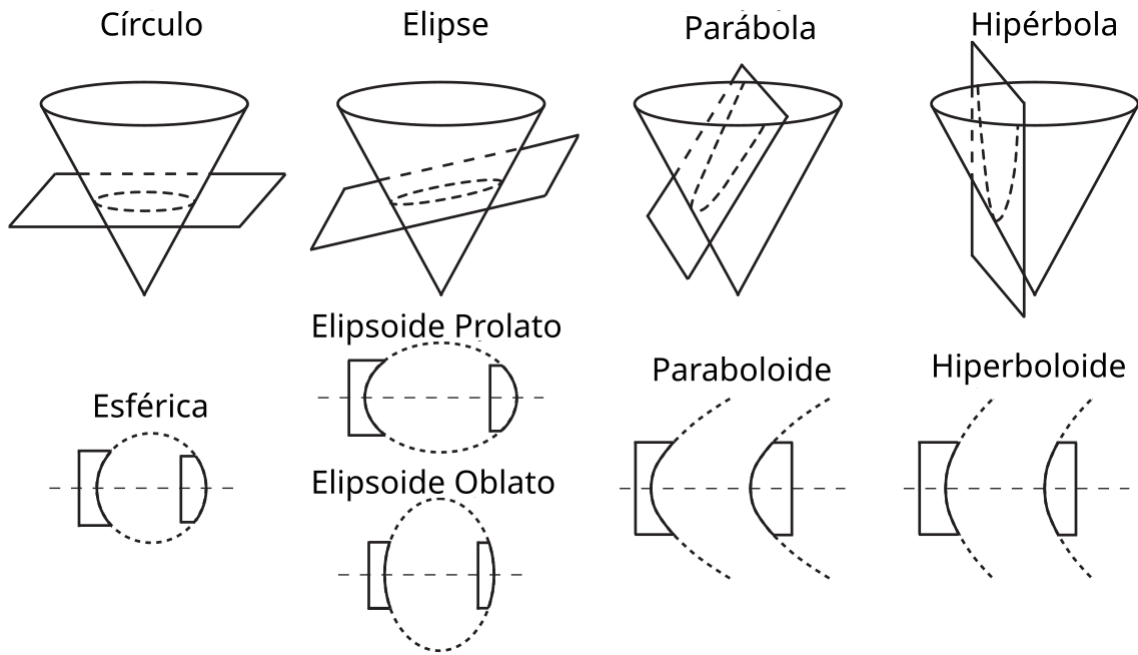


Figura 2.8: Los cuatro tipos de secciones cónicas y las superficies generadas al hacer girar la sección alrededor de sus ejes de simetría. [16].

llamada constante de conicidad. Si las constantes A_i son todas iguales a cero, la superficie es una cónica de revolución, de acuerdo a la tabla 2.1. La ecuación 2.8 es conocida como la ecuación de la sagita.

Tabla 2.1: Valores de la constante de conicidad para superficies cónicas [17].

Tipo de cónica	Constante de conicidad
Hiperboloide	$K < -1$
Paraboloide	$K = -1$
Esferoide prolato o elipsoide	$-1 < K < 0$
Esfera	$K = 0$
Esferoide oblato	$K > 0$

La ilustración de los perfiles de algunas superficies cónicas están descritas en la Figura 2.9.

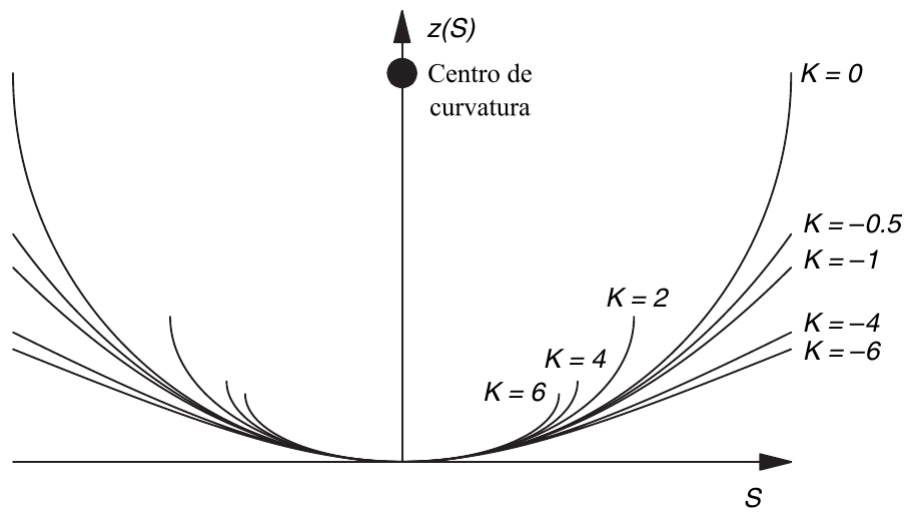


Figura 2.9: Representación de varias superficies cónicas de acuerdo a su constante de conicidad [10].

Capítulo 3

Aberraciones

3.1. Introducción

Por lo general, los sistemas ópticos generadores de imágenes tienen un eje de simetría rotacional, y su pupila puede ser circular o anular, como es el caso de sistemas con espejos. La función de aberración de varios sistemas ópticos puede ser expandida en series de potencias o como un grupo de polinomios ortogonales completos. En el área de la visión, la calidad óptica del ojo humano es común describirla en términos de las aberraciones del frente de onda, representadas por los polinomios de Zernike asociados con sus respectivos coeficientes, de ahí la importancia de usar estos polinomios para cuantificar las aberraciones del frente de onda, no solamente del ojo humano, sino de cualquier sistema óptico [18].

Estos polinomios fueron introducidos por Frits Zernike para evaluar la forma de un espejo circular por medio de la prueba de la navaja o por su método de contraste de fase [19, 20]. Los coeficientes de los términos de la expansión de los polinomios de Zernike para espejos circulares, en coordenadas cartesianas, representan a cada una de las aberraciones del frente de onda generado por un espejo o por un sistema óptico, así como también las aberraciones presentes en la superficie de un espejo.

3.2. Aberraciones del frente de onda. Aberraciones primarias

La óptica geométrica, también llamada teoría de primer orden u óptica Gaussiana, se limita a la formación de imágenes en la región paraxial, lo que otorga simplificaciones en los cálculos y da una aproximación aceptable. Cuando se realiza mediciones en un sistema óptico, fuera de la región paraxial, se observan incongruencias en la formación de la imagen, en comparación que la predicha por la óptica Gaussiana. Estas diferencias se denominan aberraciones [11].

Al clasificar los diferentes defectos de la imagen y entendiendo su comportamiento en cada tipo, el trabajo de determinar las aberraciones de un sistema óptico puede ser simplificado, solo es necesario el trazo de algunos rayos para evaluar cada aberración. Philip L. von Seidel investigó y desarrolló expresiones analíticas para la determinación de las aberraciones primarias. Por esa razón, los defectos primarios de una imagen usualmente se nombran como aberraciones de Seidel, las cuales son aberración esférica, Astigmatismo, Curvatura de campo, Distorsión y Coma [14].

Para el estudio de las aberraciones, considerando las pupilas de entrada y de salida, se considera

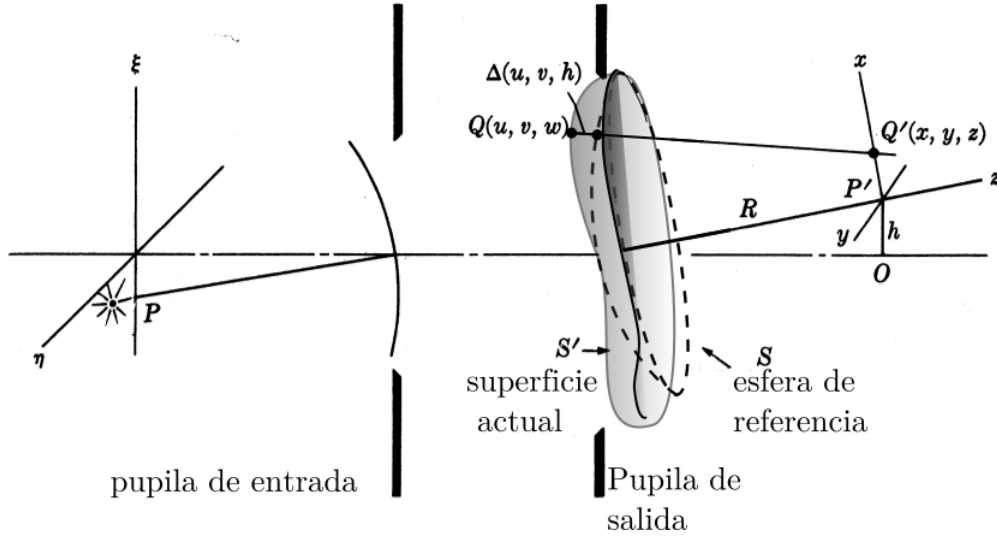


Figura 3.1: Esquema de referencia de un sistema óptico [14].

una esfera centrada en P' (ver Figura 3.1) la cual es la imagen Gaussiana de P , de radio R , que pasa a través del centro de la pupila de salida y por lo tanto a lo largo del rayo principal. Si el camino óptico fuera el mismo para cualquier rayo proveniente de P , pasando a través del sistema y emergiendo desde la pupila de salida, la esfera de referencia representará un camino óptico constante, y todos los rayos convergerán en P' . En la práctica, el camino óptico es diferente, porque la superficie óptica no es ideal o tiene deformaciones, y por lo tanto el camino óptico ya no es constante y la cantidad por la cual se desvían los rayos de una superficie ideal se cuantifica por medio de la función de aberración $\Delta(u, v, h)$. Existen varias representaciones de $\Delta(u, v, h)$ para cuantificar esas deformaciones, una representación consiste en una expansión en serie de potencias en coordenadas polares ρ y ϕ . Por consiguiente, se expresa la función Δ para el caso general como [21]:

$$\begin{aligned} \Delta(\rho, \phi, h) = & a_0 + b_0 h^2 + b_1 \rho^2 + b_2 h \rho \cos \phi + c_0 h^4 + c_1 \rho^4 \\ & + c_2 h^2 \rho^2 \cos^2 \phi + c_3 h^2 \rho^2 + c_4 h^3 \rho \cos \phi \\ & + c_5 h \rho^3 \cos \phi + d_0 h^6 + d_1 \rho^6 + \dots \end{aligned} \quad (3.1)$$

donde

$$\begin{aligned} u = a \rho \cos \phi, & \quad \rho = \sqrt{(u^2 + v^2)/a^2}, & \quad 0 \leq \rho \leq 1, \\ v = a \rho \sin \phi, & \quad \phi = \tan^{-1} v/u, & \quad a = \text{radio de la pupila de salida.} \end{aligned}$$

Se observa también que en la expresión 3.1 los términos b_1 y b_2 representan los ajustes de foco, mientras que las constantes c_1, c_2, c_3 y c_4 representan las aberraciones de Seidel, las cuales también son llamadas aberraciones primarias.

A continuación se describen algunos términos de la expansión, así como los términos que especifican las aberraciones primarias [21]:

Términos constantes. Los términos a_0, b_0 y c_0 , no tienen dependencia con las variables espaciales, y dada su naturaleza indican que se ha planteado condiciones ideales de la esfera de referencia con la óptica gaussiana.

Error de enfoque:

- **Longitudinal.** La función que describe el error longitudinal es $\Delta = b_1 \rho^2$. Los anillos de rayos que describe esta ecuación se enfocan a una distancia δL , con respecto al radio gaussiano, como lo muestra la Figura 3.2b. Lo que provoca que la nueva esfera se centre en un radio fuera del foco y genere el desenfoco de la imagen.

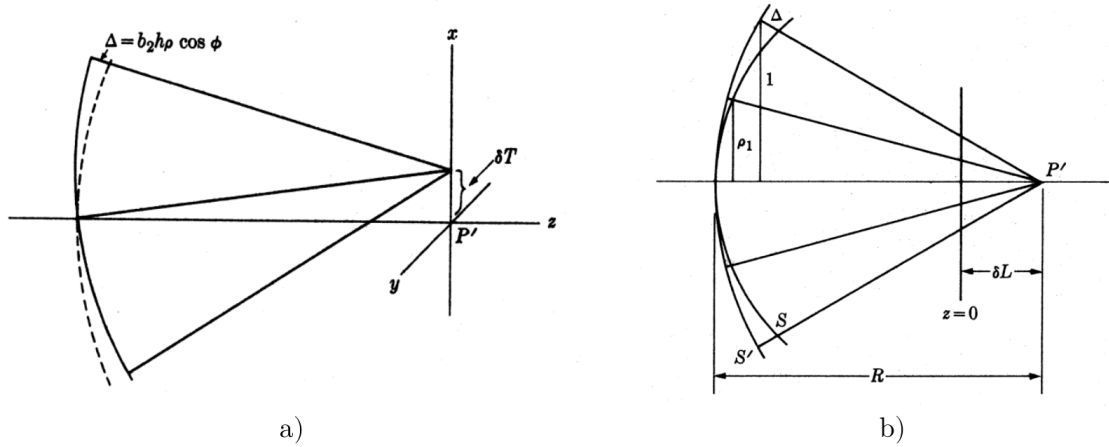


Figura 3.2: Errores de enfoque: a) error de enfoque transversal, b) error longitudinal [21].

- **Transversal.** La ecuación $\Delta = b_2 h \rho \cos \phi = \frac{b_2 h u}{a}$ describe la convergencia de los rayos a una distancia δT del punto imagen gaussiano o rayo principal, de tal manera que se inclina o desvía de manera tangencial al eje óptico, como lo podemos apreciar en la imagen 3.2a.

Los siguientes cinco términos de la serie de expansión de Δ representan las aberraciones clásicas de Seidel de tercer orden [21]:

- **Aberración esférica.** Se utiliza la función de aberración $\Delta = c_1 \rho^4$. La aberración esférica es la más importante de las aberraciones de Seidel o monocromáticas, ya que es la única que afecta a todo el campo, incluyendo las cercanías del eje óptico. Su nombre viene del hecho de que esta aberración se produce aún en las superficies perfectamente esféricas. La posición del foco depende de la altura del rayo sobre la superficie refractora. La envolvente de los rayos refractados forma una curva característica llamada cáustica [10]. La distancia dada por la intersección de uno de estos rayos con el eje óptico y el foco paraxial, se define como aberración esférica longitudinal, mostrada en la Figura 3.3a como δL , mientras que la altura de la imagen en el foco paraxial se le denomina aberración esférica transversal [11]. La deformación que proviene del plano receptor se escribe como

$$\Delta(\rho) = c_1(\rho^4 - 2\mu\rho^2), \quad (3.2)$$

donde

$$\mu = \frac{\delta f}{\delta L}, \quad (3.3)$$

describe la aberración esférica transversal dependiente de los rayos incidentes, δf representa desplazamiento del foco dentro del intervalo $[0, \delta L]$ (figura 3.3b): cuando $\mu = 0$ corresponde

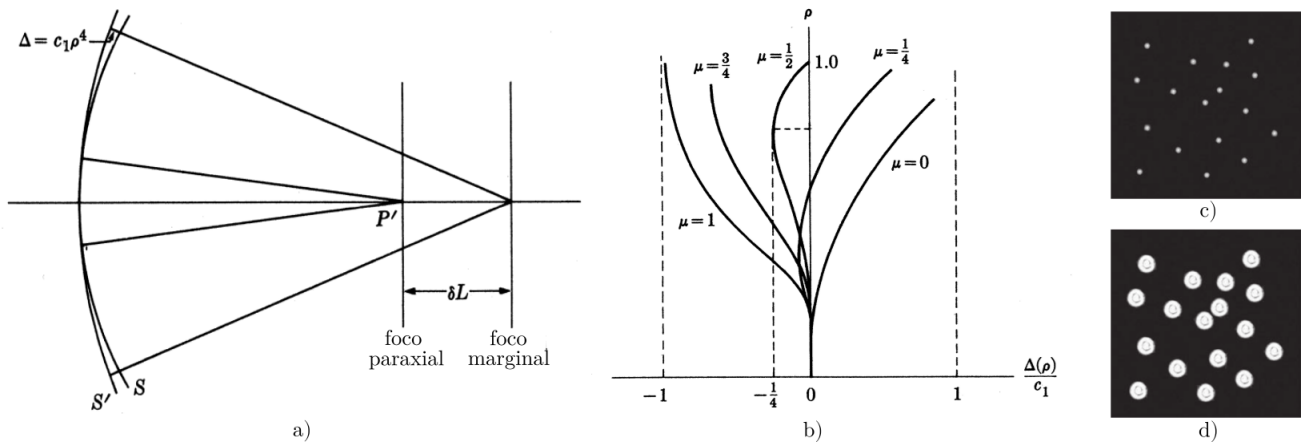


Figura 3.3: a) Aberración esférica longitudinal, b) aberración esférica transversal [21]. c) Imagen estelar sin aberraciones, d) imagen estelar con aberración esférica [10].

al foco paraxial, mientras que para $\mu = 1$ corresponde al foco marginal. En la Figura 3.3c se muestra una imagen de un conjunto de cuerpos celestes sin ninguna aberración, mientras que en la 3.3d la desviación de los rayos por la aberración de esfericidad provoca diversos puntos de convergencia cercanos en cada uno de estos cuerpos [10].

- **Astigmatismo y curvatura de campo.** Consideradas como aberraciones de la misma clase, se definen como:

$$\Delta = c_2 h^2 \rho^2 \cos^2 \phi + c_3 h^2 \rho^2. \quad (3.4)$$

Se reducen los cálculos al expresar la función de aberración en coordenadas cartesianas u, v :

$$\Delta = \frac{h^2}{a^2} [(c_2 + c_3)u^2 + c_3v^2]. \quad (3.5)$$

La aberración llamada Astigmatismo, contribuye a la degradación de la imagen fuera del eje. Esta aberración es la separación entre el punto de convergencia de los rayos meridionales y el punto de convergencia de los rayos sagitales [10], tal que los planos sagital $u = 0$ y tangencial $v = 0$ aparecen como errores focales longitudinales, de modo que el frente de onda posee diferente curvatura en esos dos planos [21].

En el caso de un punto objeto axial, el cono de rayos es simétrico respecto a las superficies esféricas de una lente. No hay necesidad de diferenciar entre planos sagitales y meridionales. Las configuraciones de todos los rayos en todos los planos que contienen al eje óptico son idénticas. Cuando no existe aberración esférica, las distancias de los rayos focales son iguales, y en consecuencia, los rayos llegarán en un frente común. En cambio, el haz oblicuo presente en la aberración astigmática será diferente tanto en el plano sagital y tangencial [11]; como se observa en la Figura 3.4a, esto provoca que la imagen tenga un desenfoque lineal, elíptico o circular, dependiendo de la intensidad en los planos que lo describe.

Si una rueda con rayos, como se muestra en la Figura 3.4b, se usa como objeto y se forma su imagen por medio de una lente con Astigmatismo, se obtiene la imagen en la superficie tangencial y la imagen siguiente a esta en la superficie sagital. En superficies intermedias

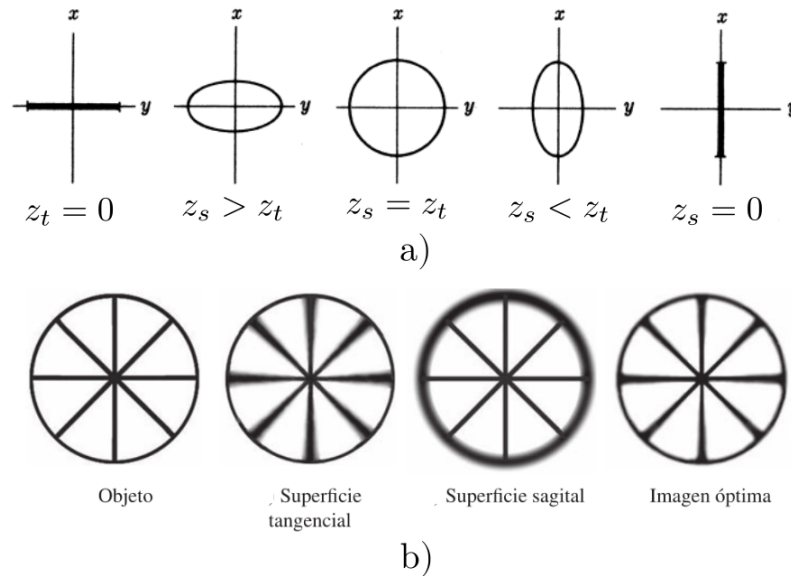


Figura 3.4: Aberración de Astigmatismo. a) Intensidad de rayos incidentes en los planos tangencial y sagital [21]. b) Perfil astigmático en una imagen respecto a su objeto ideal [10].

entre la sagital y la tangencial la imagen presenta una falta de definición general como en el caso de la coma, aunque en general menos pronunciada, como se muestra en el último caso de esta Figura [10].

Suponiendo un sistema óptico sin aberraciones, en él existirá una correspondencia uno a uno entre los puntos de las superficies objeto e imagen (es decir, imágenes estigmáticas). Para un objeto plano, normal al eje, tendrá como imagen un plano solo en la región paraxial. Para aperturas finitas, la curvatura resultante en las imágenes estigmáticas es la manifestación de la aberración primaria conocida como curvatura de campo de Petzval [11]. Esta descripción se muestra de forma gráfica en la Figura 3.5.

- **Distorsión.** Se utiliza la función de aberración

$$\Delta = c_4 h^3 \rho \cos \phi = \frac{c_4 h^3}{a} u. \quad (3.6)$$

Aún si todos los rayos que parten de un punto en el objeto llegaran a un solo punto en el plano focal, podría existir un tipo más de aberración llamado distorsión. La distorsión puede ser positiva (de barril) o negativa. Si el objeto es cuadrado, la imagen tiene una forma de barril cuando la distorsión es positiva, o los lados se curvan hacia adentro si la distorsión es negativa, como se muestra en la Figura 3.6 [10]. La distorsión se manifiesta como una amplificación no constante, es decir que no es la misma para cualquier altura en la imagen. La amplificación disminuye con la altura de la imagen si la distorsión es positiva o aumenta si la distorsión es negativa [10].

- **Coma.** Para la aberración tipo Coma utilizamos la función $\Delta = c_5 h \rho^3 \cos \phi$ [21]. Es una aberración que deteriora la imagen, asociada con un punto objeto apartado del eje aunque sea a una distancia corta. Su origen radica en el hecho de que los planos principales en las lentes

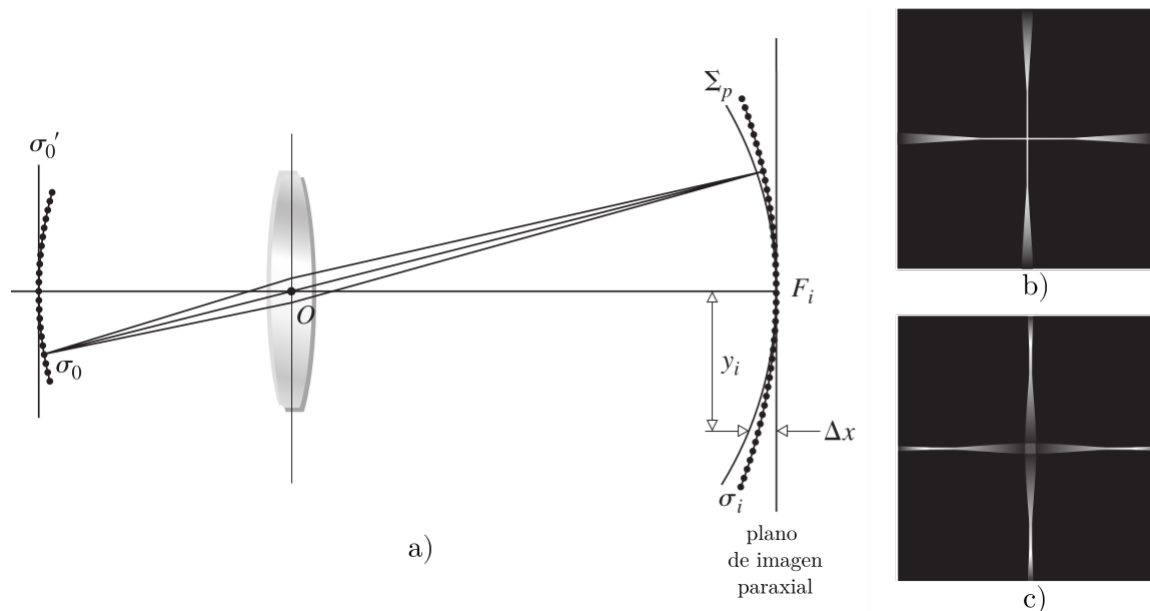


Figura 3.5: Curvatura de campo: a) cuando el objeto corresponde a σ_0' , la imagen corresponderá a la superficie Σ_p . b) La imagen formada en una pantalla plana cerca del plano imagen paraxial estará enfocada tan solo en su centro. c) Al acercar la pantalla a la lente, se enfocarán los extremos [11].

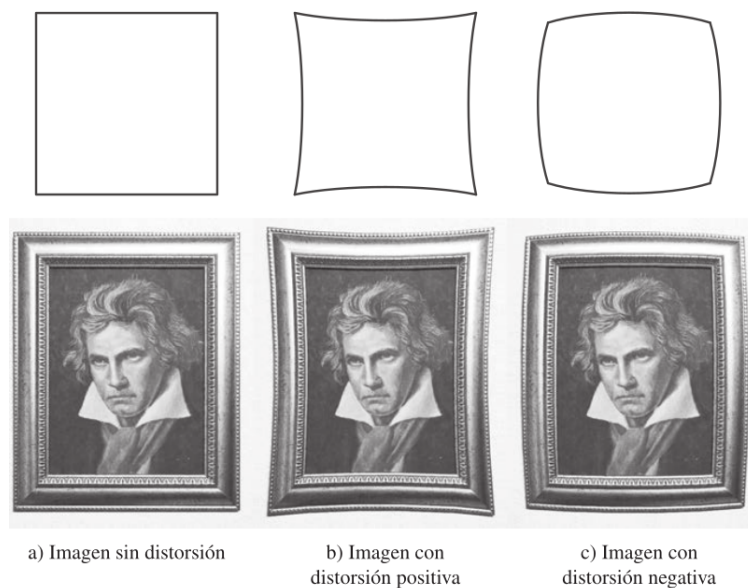


Figura 3.6: Proyección de una imagen de acuerdo al tipo de aberración de distorsión [10].

gruesas pueden ser considerados solamente en la región paraxial, ya que realmente son superficies curvas principales. Cuando no existe aberración esférica, un haz de rayos paralelos serán enfocados en el punto del foco imagen desde el vértice posterior. Pero las distancias focales efectivas y, por lo tanto, los aumentos laterales, serán diferentes para aquellos rayos que crucen la zona de la lente que se encuentra fuera del eje. Cuando el punto imagen está sobre el eje óptico, esta situación no tiene mayor trascendencia, pero cuando el haz es oblicuo, y el

punto imagen está fuera del eje, la Coma es evidente. La dependencia de M_T (amplificación transversal) con respecto a h , la altura del rayo en la lente, puede verse en la Figura 3.7a. Vemos cómo los rayos meridionales que cruzan las orillas de la lente llegan al plano imagen más cerca del eje que los rayos que se encuentran en las proximidades del rayo principal. En estas circunstancias, el aumento más pequeño está asociado con los rayos marginales, que formarán la imagen más pequeña; en este caso la Coma será negativa. La Coma de la Figura 3.7b será positiva, ya que los rayos marginales se enfocarán a una distancia más alejada del eje. En la Figura 3.7c aparecen dibujados varios rayos oblicuos a partir de un objeto fuera del eje S . Obsérvese que cada cono circular de rayos, cuyos puntos finales forman un anillo sobre la lente, proyectan una imagen en lo que H. Dennis Taylor denominó un círculo comático sobre Σ_i . En este caso corresponde a una Coma positiva, y por consiguiente, cuanto más grande sea el anillo sobre la lente, mayor será la distancia que separa el círculo comático del eje [11].

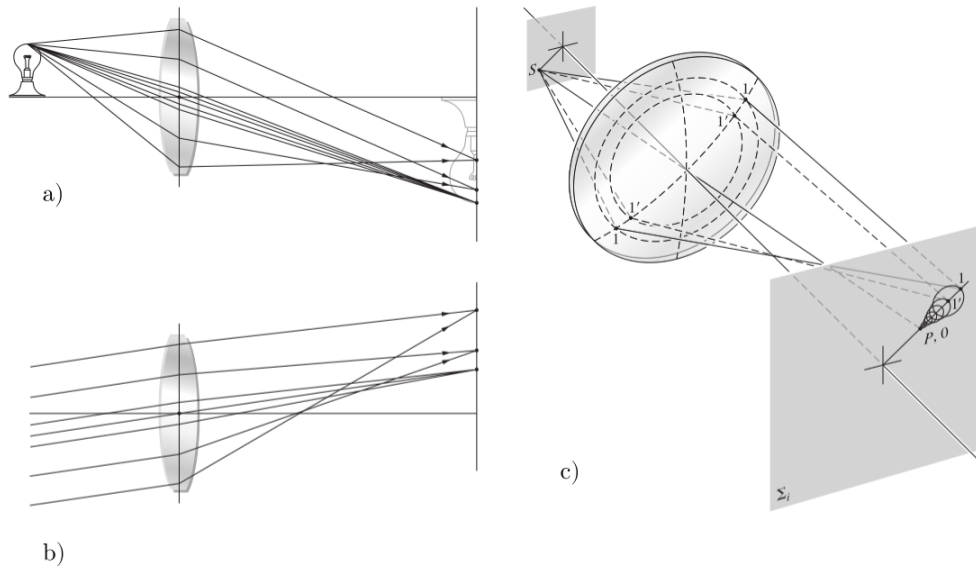


Figura 3.7: Localización geométrica de la aberración de Coma de una fuente puntual monocromática. La región central de las lentes forma un punto imagen en el vértice del cono [11].

3.3. Polinomios de aberración del frente de onda.

Siguiendo la deducción de D. Malacara y Z. Malacara [15], de forma general y sin asumir ninguna simetría, la forma del frente de onda puede representarse por el polinomio:

$$W(x,y) = \sum_{i=0}^k \sum_{j=0}^i c_{ij} x^i y^{i-j}, \quad (3.7)$$

incluyendo términos de aberraciones de alto orden, donde k es el grado de este polinomio. En coordenadas polares, se define

$$x = \rho \sin \theta, \quad y = \rho \cos \theta, \quad (3.8)$$

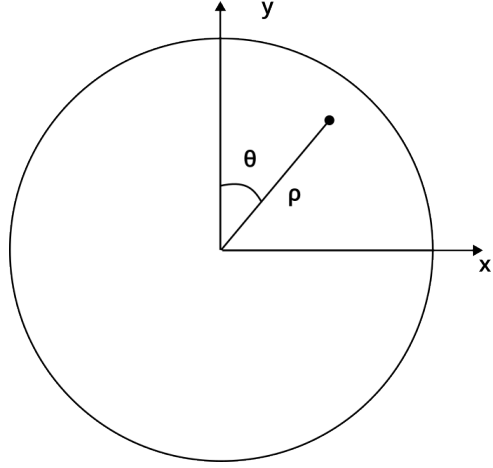


Figura 3.8: Coordenadas polares para el rayo en la pupila de entrada de un sistema óptico [15].

donde el ángulo θ es medido con respecto al eje y , como se ve en la Figura 3.8. Por lo tanto, la forma del frente de onda se reescribe como:

$$W(\rho, \theta) = \sum_{n=0}^k \sum_{l=0}^n \rho^n (a_{nl} \cos^l \theta + b_{nl} \sin^l \theta), \quad (3.9)$$

donde los términos $\cos \theta$ y $\sin \theta$ describen los componentes simétricos y antisimétricos del frente de onda, respectivamente. Sin embargo, no todos los posibles valores de n y l están permitidos. Para tener una función de un solo valor, se debe satisfacer la condición:

$$W(\rho, \theta) = W(-\rho, \theta + \pi), \quad (3.10)$$

con esta condición, tanto n y l deben ser ambos par o impar. Si esta expresión del frente de onda es transformado a coordenadas cartesianas $W(x, y)$, se convertirá en una serie infinita, a menos que $l \leq n$. Ahora, si nos limitamos al caso de un frente de onda producido por un sistema óptico axialmente simétrico, con un objeto puntual desplazado a lo largo del eje y , el frente de onda es simétrico respecto al plano tangencial o meridional, obteniendo

$$W(\rho, \theta) = \sum_{n=0}^k \sum_{l=0}^n \rho^n a_{nl} \cos^l \theta. \quad (3.11)$$

Si se incluye la expresión de la altura de la imagen h y se imponen las condiciones

$$W(\rho, \theta, h) = W(\rho, -\theta, h), \quad (3.12)$$

por la simetría en el plano meridional y

$$W(\rho, \theta, h) = W(\rho, \theta + \pi, -h), \quad (3.13)$$

por la simetría rotacional del sistema de lentes cercano al eje óptico, en la expresión del frente de onda, se puede mostrar que solo tenemos los términos de la forma:

$$\rho^2, h\rho \cos \theta, h^2 \quad (3.14)$$

y sus productos. Una consecuencia interesante es que la suma de las potencias de ρ y h siempre es un número par. Cuanto mayor sea este número, mayor será el orden de aberración. El frente de onda puede ser representada por una combinación lineal de estas aberraciones, con los términos:

$${}_k W_{nl} \rho^n h^k \cos^l \theta, \quad (3.15)$$

donde k es la potencia de la altura de la imagen h , n , es la potencia de la apertura ρ , y l es la potencia de $\cos \theta$, con $l \leq n$. Así, obtenemos el frente de onda $W(\rho, \theta, h)$, como la suma de uno o más términos, o bien, $\Delta(\rho, \phi, h)$ como se expresó en la ecuación 3.1. Cada uno de los términos que se pueden obtener tienen nombre, pero no para todos los de alto orden. Hopkins [22] propuso de manera general los siguientes nombres:

- Aberraciones esféricas: términos independientes de θ ($k = 0$).
- Aberraciones comáticas: Términos con potencias impares de $\cos \theta$ (k impar).
- Aberraciones astigmáticas: términos con potencias pares de $\cos \theta$ (k par).

Un caso particular de un sistema óptico centrado considerando únicamente aberraciones primarias, la dependencia de altura de la imagen no se toma en cuenta y la fase del campo se desprecia, el polinomio de aberraciones del frente de onda puede ser escrita de manera compacta como lo describe Kingslake (1925-1926) y también estudiado por A. Cordero Dávila y J. González García [9]:

$$W(\rho, \theta) = F + E\rho \cos \theta + D\rho^2 + C\rho^2(1 + 2\cos^2 \theta) + B\rho^3 \cos \theta + A\rho^4, \quad (3.16)$$

siendo A el coeficiente de aberración esférica, B el coeficiente de coma, C corresponde al coeficiente de Astigmatismo, D es el coeficiente de Defoco. E se considera como Tilt en el eje x (si el desplazamiento de la imagen ocurre a lo largo del eje y) y por último F , que es el término constante o de pistón.

3.4. Polinomios de Zernike.

La función de aberración de un sistema con una pupila de salida circular se puede expandir en términos de un conjunto completo de polinomios circulares de Zernike $R_n^m(\rho) \cos m\theta$ [17], que son ortogonales sobre el círculo unitario de la forma

$$W(\rho, \theta) = \sum_{n=0}^{\infty} \sum_{m=0}^n c_{nm} Z_n^m(\rho, \theta), \quad (3.17)$$

donde c_{nm} es la expansión de coeficientes que dependen de la locación del punto objeto, n y m son enteros positivos incluyendo cero, $n - m \geq 0$ y pares, también considerar que

$$Z_n^m(\rho, \theta) = [2(n+1)/(1 + \delta_{m0})]^{1/2} R_n^m(\rho) \cos m\theta, \quad (3.18)$$

es un polinomio ortonormal de Zernike. δ_{ij} es la delta de Kronecker y

$$R_n^m(\rho) = \sum_{s=0}^{(n-m)/2} \frac{(-1)^s (n-s)!}{s! (\frac{n+m}{2} - s)! (\frac{n-m}{2} - s)!} \rho^{n-2s}, \quad (3.19)$$

es un polinomio de grado n en ρ que contiene los términos $\rho^n, \rho^{n-2}, \dots$, y ρ^m . Los polinomios radiales del círculo $R_n^m(\rho)$ son pares e impares en ρ , dependiendo si n o m es par o impar. Note que $R_n^m(1) = 1$, que $R_n^m(\rho) = \rho^n$ y que $R_n^m(0) = \delta_{m0}$; para $n/2$ par y $-\delta_{m0}$ para $n/2$ impar.

Los polinomios de Zernike son ortogonales de acuerdo a las siguientes expresiones:

$$\frac{1}{\pi} \int_0^1 \int_0^{2\pi} Z_n^m(\rho, \theta) Z_{n'}^{m'}(\rho, \theta) \rho d\rho d\theta = \delta_{mm'} \delta_{nn'}, \quad (3.20)$$

$$\int_0^{2\pi} \cos m\theta \cos m'\theta d\theta = \pi(1 + \delta_{m0}) \delta_{mm'}, \quad (3.21)$$

y

$$\int_0^1 R_n^m(\rho) R_{n'}^m(\rho) \rho d\rho = \frac{1}{2(n+1)} \delta_{nn'}. \quad (3.22)$$

Los coeficientes de expansión de Zernike están dados por:

$$c_{nm} = \frac{1}{\pi} \int_0^1 \int_0^{2\pi} W(\rho, \theta) Z_n^m(\rho, \theta) \rho d\rho d\theta. \quad (3.23)$$

Los polinomios circulares de Zernike son únicos porque son polinomios en dos variables ρ y θ , también son ortogonales sobre un círculo unitario, son invariantes respecto a la rotación de los ejes coordenados cercanos al origen e incluyen un polinomio para cada valor de n y m permitidos [17].

3.5. Aberraciones del frente de onda usando los polinomios de Zernike.

La función de aberración para un sistema sin un eje de simetría rotacional consistirá tanto en términos de $\cos \theta$ como en términos de $\sin \theta$. Esto también es válido para las aberraciones resultantes de errores de fabricación, así como aquellas introducidas por la turbulencia atmosférica. En tales casos, la función de aberración se puede expandir en términos de polinomios de círculo de Zernike ortogonales $Z_j(\rho, \theta)$ de la forma [17]:

$$W(\rho, \theta) = \sum_j a_j Z_j(\rho, \theta), \quad (3.24)$$

donde a_j son los coeficientes de expansión. Los polinomios pueden ser escritos como

$$Z_j(\rho, \theta) = \sqrt{2(n+1)} R_n^m(\rho) \cos m\theta, \quad m \neq 0 \quad j \text{ par}, \quad (3.25)$$

$$Z_j(\rho, \theta) = \sqrt{2(n+1)} R_n^m(\rho) \sin m\theta, \quad m \neq 0 \quad j \text{ impar}, \quad (3.26)$$

$$Z_j(\rho, \theta) = \sqrt{n+1} R_n^0(\rho), \quad m = 0 \quad (3.27)$$

donde $R_n^m(\rho)$ son los polinomios radiales dados por la ecuación 3.19. n y m son enteros positivos, (incluyendo cero) y $n - m \geq 0$ e impar. El índice n representa la potencia más grande de ρ en el polinomio, y m se define como *frecuencia azimutal*. El índice j es un número de ordenamiento de polinomios y es una función de n y m . Los polinomios ortonormales y la relación entre los índices

j , n y m están dados en la tabla 3.1, mientras que en la Figura 3.9 se muestran los frentes de onda generados por cada aberración de hasta orden 10. Los polinomios están ordenados de tal manera que un j par corresponde a un polinomio simétrico que varía como $\cos m\theta$, mientras que un j impar corresponde a un polinomio antisimétrico que varía como $\sin m\theta$. El número de términos para un valor dado de n es $n + 1$. El número de términos de cierto orden n , incluyéndolo, está dado por

$$N_n = (n + 1)(n + 2)/2. \quad (3.28)$$

Tabla 3.1: Primeros 28 términos de los polinomios ortonormales circulares de Zernike $Z_j(\rho, \theta)$. Los índices j , n y m son llamados números polinomiales, grado radial, y frecuencia azimutal, respectivamente.

j	n	m	$Z_j(\rho, \theta)$	Nombre de la aberración
1	0	0	1	Piston
2	1	1	$2\rho \cos \theta$	Tilt en x
3	1	1	$2\rho \sin \theta$	Tilt en y
4	2	0	$\sqrt{3}(2\rho^2 - 1)$	Defoco
5	2	2	$\sqrt{6}\rho^2 \sin 2\theta$	Astigmatismo primario a 45°
6	2	2	$\sqrt{6}\rho^2 \cos 2\theta$	Astigmatismo primario en 0°
7	3	1	$\sqrt{8}(3\rho^3 - 2\rho) \sin 2\theta$	Coma primaria en y
8	3	1	$\sqrt{8}(3\rho^3 - 2\rho) \cos 2\theta$	Coma primaria en x
9	3	3	$\sqrt{8}\rho^3 \sin 3\theta$	
10	3	3	$\sqrt{8}\rho^3 \cos 3\theta$	
11	4	0	$\sqrt{5}(6\rho^4 - 6\rho^2 + 1)$	Esférica primaria
12	4	2	$\sqrt{10}(4\rho^4 - 3\rho^2) \cos 2\theta$	Astigmatismo secundario a 0°
13	4	2	$\sqrt{10}(4\rho^4 - 3\rho^2) \sin 2\theta$	Astigmatismo secundario a 45°
14	4	4	$\sqrt{10}\rho^4 \cos 4\theta$	
15	4	4	$\sqrt{10}\rho^4 \sin 4\theta$	
16	5	1	$\sqrt{12}(10\rho^5 - 12\rho^3 + 3\rho) \cos \theta$	Coma secundaria en x
17	5	1	$\sqrt{12}(10\rho^5 - 12\rho^3 + 3\rho) \sin \theta$	Coma secundaria en y
18	5	3	$\sqrt{12}(5\rho^5 - 4\rho^3) \sin 3\theta$	
19	5	3	$\sqrt{12}(5\rho^5 - 4\rho^3) \cos 3\theta$	
20	5	5	$\sqrt{12}\rho^5 \cos 5\theta$	
21	5	5	$\sqrt{12}\rho^5 \sin 5\theta$	
22	6	0	$\sqrt{7}(20\rho^6 - 30\rho^4 + 12\rho^2 - 1)$	Esférica secundaria
23	6	2	$\sqrt{14}(15\rho^6 - 20\rho^4 + 6\rho^2) \sin 2\theta$	Astigmatismo terciario a 45°
24	6	2	$\sqrt{14}(15\rho^6 - 20\rho^4 + 6\rho^2) \cos 2\theta$	Astigmatismo terciario a 0°
25	6	4	$\sqrt{14}(6\rho^6 - 5\rho^4) \sin 4\theta$	
26	6	4	$\sqrt{14}(6\rho^6 - 5\rho^4) \cos 4\theta$	
27	6	6	$\sqrt{14}\rho^6 \sin 6\theta$	
28	6	6	$\sqrt{14}\rho^6 \cos 6\theta$	

Para un valor dado de n , el número de polinomios N_n representa el valor más grande de j . Dado que el número de polinomios con el mismo valor de n pero diferentes valores de m es igual a $n + 1$,

el valor más pequeño de j para un valor dado de n es $N_n - n$. Para un valor dado de n y m , hay dos valores de j , $N_n - n + m - 1$ y $N_n - n + m$. El valor de j con $m = 0$ es $N_n - n$. Por ejemplo, para $n = 5$, $N_n = 21$ y $j = 21$ representan los polinomios de $\sin 5\theta$. El número correspondiente a los polinomios de $\cos 5\theta$ es $j = 20$. Los dos polinomios con $m = 3$ por ejemplo, tienen valores de 18 y 19, representan los polinomios de $\cos 3\theta$ y los de $\sin 3\theta$, respectivamente.

Para un valor dado de j , n está dado por:

$$n = [(2j - 1)^{1/2} + 0.5]_{\text{entero}} - 1, \quad (3.29)$$

donde el subíndice entero implica un valor entero del número dentro de los corchetes. Una vez n conocida, el valor de m está dado por

$$m = \begin{cases} 2\{[2j + 1 - n(n + 1)]/4\}_{\text{entero}} & \text{para } n \text{ par} \\ 2\{[2(j + 1) - n(n + 1)]/4\}_{\text{entero}} - 1 & \text{para } n \text{ impar} \end{cases} \quad (3.30)$$

Los polinomios ortonormales circulares de Zernike $z_j(x, y)$ en coordenadas cartesianas (x, y) , los cuales están normalizados en una pupila de radio a se muestran en la tabla 3.2. El subíndice n en las coordenadas se omiten por simplicidad.

Por lo tanto, en el presente tema de tesis para la evaluación de los espejos cóncavos cónicos (por medio del método de cuantificación de las aberración del frente de onda) serán utilizados los polinomios de Zernike de cuarto orden en coordenadas cartesianas para describir la función de aberración W (o función de error) de la superficie del espejo, es decir, la función de aberración de la superficie se representará tal como la ecuación 3.24, con los términos escritos en coordenadas cartesianas. La tabla 3.2 muestra los primeros 15 polinomios de Zernike con el nombre de cada término.

Tabla 3.2: Polinomios ortonormales de Zernike hasta de cuarto orden en coordenadas rectangulares [17].

Número de término	Nombre de la aberración	Polinomio ortonormal de Zernike
z_1	Piston	1
z_2	Tilt en y	$2x$
z_3	Tilt en x	$2y$
z_4	Defoco	$\sqrt{3}(2x^2 + 2y^2 - 1)$
z_5	Astigmatismo primario a 45°	$2\sqrt{6}xy$
z_6	Astigmatismo primario a 0°	$\sqrt{6}(x^2 - y^2)$
z_7	Coma primaria en y	$\sqrt{8}y(3x^2 + 3y^2 - 2)$
z_8	Coma primaria en x	$\sqrt{8}x(3x^2 + 3y^2 - 2)$
z_9	Trefoil en y	$\sqrt{8}y(3x^2 - y^2)$
z_{10}	Trefoil en x	$\sqrt{8}x(x^2 - 3y^2)$
z_{11}	Esférica primara	$\sqrt{5}(6(x^2 + y^2)^2 - 6(x^2 + y^2) + 1)$
z_{12}	Astigmatismo secundario en 0°	$\sqrt{10}(4x^4 - 4y^4 - 3x^2 + 3y^2)$
z_{13}	Astigmatismo secundaria a 45°	$2\sqrt{10}(4x^3 - 4xy^3 - 3xy)$
z_{14}	Quatrefoil vertical	$\sqrt{10}[(x^2 + y^2) - 8x^2y^2]$
z_{15}	Quatrefoil oblicuo	$4\sqrt{10}xy(x^2 - y^2)$

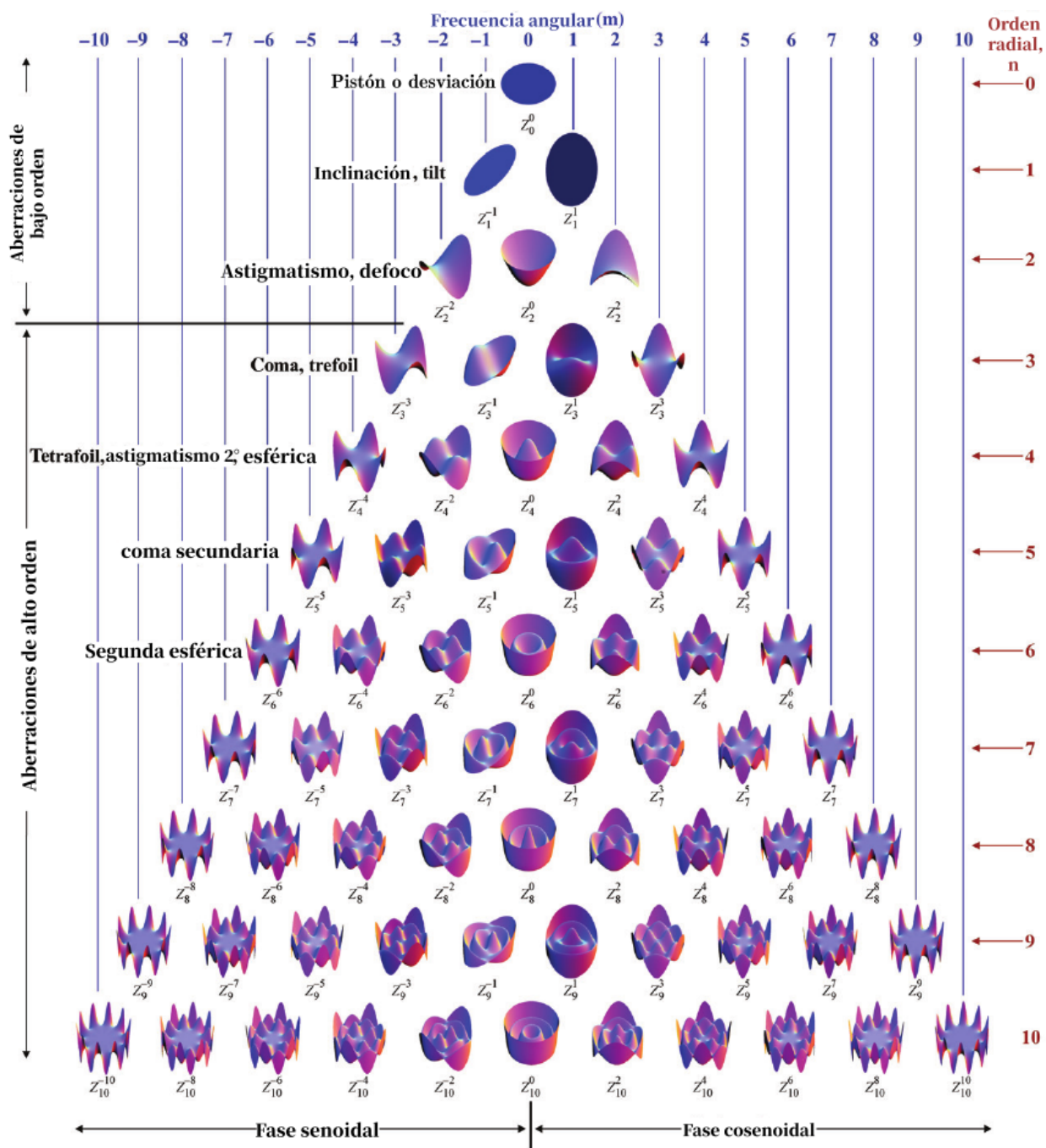


Figura 3.9: Gráficos de superficie de los polinomios de Zernike hasta el orden 10 [23].

Capítulo 4

Evaluación de espejos

4.1. Introducción

Una amplia variedad de características de los elementos ópticos, necesitan ser medidos de la manera más sencilla para garantizar que éstos operen de forma deseada. Por ejemplo, para las ventanas planas, las superficies deben ser planas y paralelas una con otra. Para una lente, los radios de curvatura de sus dos superficies deben coincidir con los valores establecidos dentro de cierta tolerancia. Además, el índice de refracción y el espesor de la lente también deben cumplir con los valores objetivo para que la lente tenga la potencia deseada. Los elementos ópticos suelen combinarse en un sistema óptico y su alineación entre sí es crucial para maximizar el rendimiento. Las lentes comerciales de elementos múltiples a menudo se proporcionan sin detalles sobre su diseño y el usuario no tiene acceso a los elementos individuales para verificar su desempeño. En tales casos, la capacidad de medir los puntos cardinales del conjunto de lentes, así como sus aberraciones del frente de onda, es útil para comprender las propiedades ópticas de la lente compuesta [16].

Las técnicas ópticas de medición son conocidas como las más precisas en la actualidad, además de ser rápidas, de no contacto y no invasivas. En años recientes, el uso de las pruebas ópticas han aumentado drásticamente en un rango amplio de aplicaciones, desde determinar la topografía de paisajes hasta verificar la rugosidad en las superficies pulidas [24].

Los instrumentos ópticos utilizados miden información que es transportada por la luz. El interés radica en la luz misma o en alguna modificación realizada por la interacción con un objeto en estudio. Las características de la luz susceptibles de medición son intensidad, fase, color, polarización, coherencia y direccionalidad. Entre otras cosas, hacer evaluaciones ópticas nos permiten medir y caracterizar sistemas de imágenes, evaluando datos marginales, realizar diagnósticos de haz, tal como la detección del frente de onda y realizar calibraciones radiométricas [25].

4.2. Pruebas ópticas no interferométricas

Las pruebas no interferométricas del error del frente de onda proporcionan un medio rápido para determinar la calidad del sistema. Estas pruebas normalmente se realizan en dos casos. El primer caso es la fabricación de una superficie óptica. La luz de una fuente se refleja desde la superficie y se analiza el frente de onda resultante. Las aberraciones en este frente de onda reflejado están estrechamente relacionadas con la forma de la superficie. Los errores al esmerilar y pulir la Figura

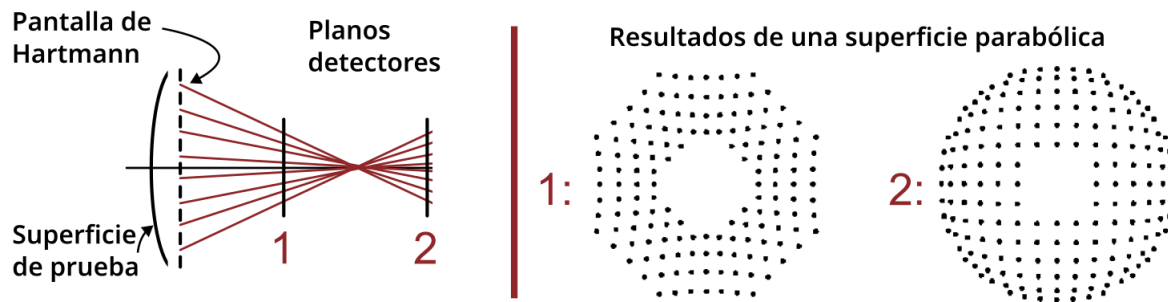


Figura 4.1: Prueba de Hartmann aplicada a una superficie cóncava [26].

correcta en la superficie darán lugar a aberraciones. En consecuencia, este tipo de pruebas pueden proporcionar retroalimentación para el proceso de fabricación que permite un trabajo adicional de la superficie para eliminar errores de forma. Estos procedimientos de prueba de frente de onda también se pueden utilizar para evaluar el rendimiento de los sistemas ópticos. En estos casos, normalmente se utiliza una fuente de luz colocada en la posición adecuada dentro del sistema óptico y que emite en el rango de longitudes de onda para el cual el sistema está diseñado. La alineación del sistema de prueba se puede ajustar para medir individualmente ángulos de campo particulares. El frente de onda generado del sistema se mide en la pupila de salida y se compara con su rendimiento esperado. Las desviaciones del frente de onda esperado sugieren problemas con el centrado, la inclinación y la potencia de los elementos individuales del mismo. A menudo, este tipo de prueba se utiliza para determinar si el sistema cumple con alguna especificación mínima de rendimiento [16].

Las siguientes técnicas son métodos comunes de pruebas ópticas no interferométricas que miden la pendiente del error de frente de onda en lugar del propio error de frente de onda, que es lo que la mayoría de los interferómetros miden [26]:

Prueba de Hartmann. La prueba clásica de Hartmann utiliza una placa con una matriz de agujeros cerca de una superficie de prueba iluminada. Se colocan placas fotográficas o matrices de detectores en el haz convergente, como se muestra en la Figura 4.1. Si se utiliza un solo detector, se deben conocer las posiciones de los agujeros en la pantalla y la separación entre la pantalla y el plano del detector. Si se utilizan dos placas, solo se debe conocer la distancia entre los dos planos de los detectores [26].

La prueba más común utilizada en la actualidad es la prueba *Shack-Hartmann*, donde los agujeros en la pantalla son reemplazados por pequeñas lentes, creando una matriz de lentes. Un esquema de su funcionamiento, se puede apreciar en la Figura 4.2. Por lo general, el frente de onda de prueba transmitido o reflejado incide en la matriz de lentes [26]. Un detector se coloca en el plano focal posterior de las lentes. Una onda plana producirá una matriz uniforme de puntos, mientras que cualquier desviación de pendiente en el frente de onda desplazará los puntos. Las ubicaciones de los puntos se utilizan para encontrar la pendiente del frente de onda en el plano de las lentes. El rango dinámico y la precisión de esta prueba varían según la longitud focal de las lentes individuales y el número de lentes en la matriz. Esta prueba se utiliza para pruebas ópticas y en óptica adaptativa para la corrección atmosférica activa en telescopios [26].

Prueba de la navaja. También conocida como prueba de Foucault. Este test es uno de los más antiguos y comunes para determinar aberraciones longitudinales y transversales. Se coloca el borde de una navaja cerca del foco y se hace pasar a través de la imagen de una fuente puntual o

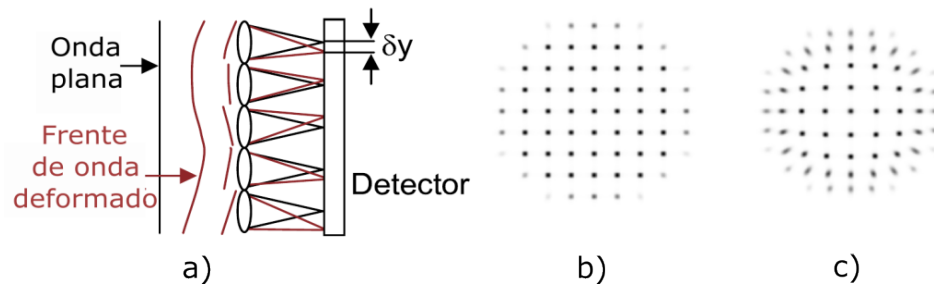


Figura 4.2: a) Configuración de un sensor Shack-Hartmann [26], b) Un frente de onda plano incidente en el sensor Shack-Hartmann produce una rejilla uniforme de puntos separado. c) Cuando están presentes las aberraciones, los puntos se desplazan, codificando la pendiente local del frente de onda incidente [16].

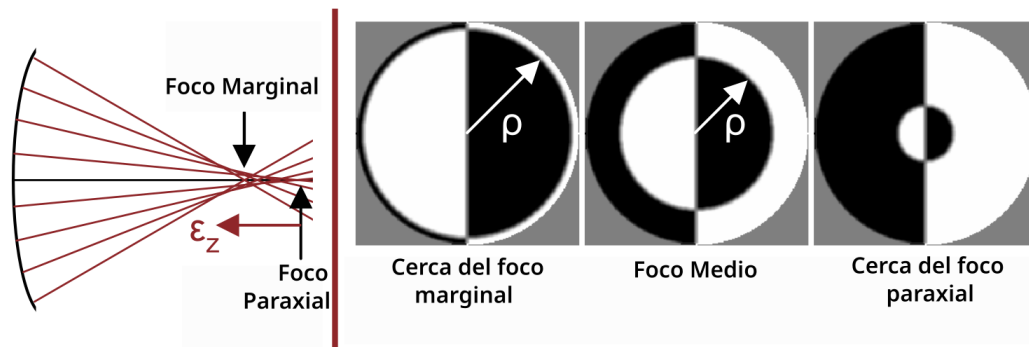


Figura 4.3: Detección de la aberración esférica usando la prueba de la navaja [26].

una rendija. La sombra resultante, observada por el ojo o en una pantalla, proporciona información sobre las aberraciones del sistema [26].

Cuando se introduce un filo de navaja, cortando el haz iluminado que forma la imagen, un observador (colocado detrás de la imagen) ve un patrón de sombra que aparece sobre la superficie uniformemente iluminada del espejo. En el caso de un espejo esférico “perfecto”, cuando se introduce el filo de la navaja dentro del foco (hacia el espejo), el patrón de sombra consiste en una región oscura y una región brillante, claramente separadas como se muestra en la Figura 4.3. A medida que se mueve el filo de la navaja, la región oscura parece moverse a través del espejo en la misma dirección en la que se mueve la navaja [26].

La prueba de la navaja se utiliza principalmente para medir el enfoque zonal de diferentes partes de una superficie óptica para que el óptico pueda determinar las partes de la superficie que están más altas o más bajas [26].

Prueba del alambre. La prueba del alambre es un método estrechamente relacionado con la prueba de Foucault. En lugar de un borde afilado, se utiliza un alambre delgado como máscara. En este caso, la transmisión de luz es cero solo a lo largo de la línea recta del alambre [16].

La prueba del alambre es inferior para obtener datos cualitativos, pero es superior para obtener datos cuantitativos, ya que el alambre produce una sombra simétrica. Es más fácil determinar el centro de la sombra en este caso [26].

La principal ventaja de esta prueba sobre la prueba de la navaja, cuando se usa con la superficie óptica a evaluar, es la capacidad del primero de restringir la acción de bloqueo a una región muy

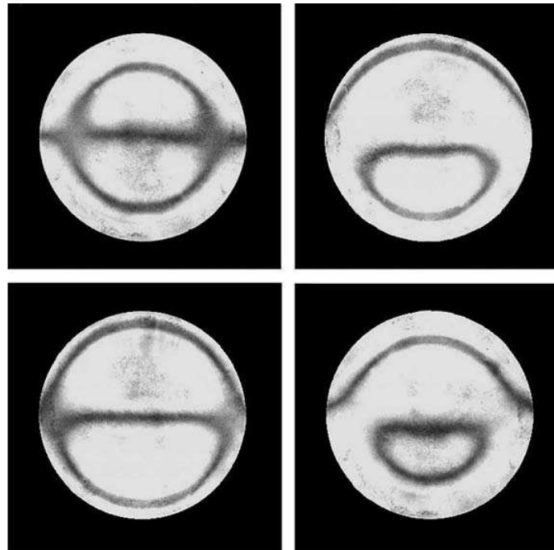


Figura 4.4: Algunas sombras obtenidas por medio de la prueba del alambre [17].

estrecha en el anillo que está siendo evaluado. Además, los efectos de difracción son simétricos con respecto al anillo. Como resultado, la medición de la intersección de las normales con el eje óptico es más precisa, y de esta manera se pueden medir las desviaciones entre los valores experimentales de las intersecciones y los valores teóricos. En otras palabras, se determinan las aberraciones de las normales. Además, la prueba del alambre tiene la ventaja de que las mediciones de las intersecciones de las normales con el eje óptico se refieren al valor teórico de la intersección del primer anillo paraxial. Esta característica puede resultar extremadamente útil al probar una muestra óptica con un agujero central [17]. En la Figura 4.4 son mostrados algunos patrones obtenidos por la prueba del alambre.

El principal inconveniente del test del alambre es que el ancho del patrón de sombra a veces oculta pequeños desplazamientos de toda la sombra mientras el alambre se desplaza a lo largo del eje óptico [17].

El concepto de la prueba del alambre se expande a la prueba de Ronchi, que utiliza una rejilla de Ronchi como máscara. La máscara que consiste en una serie de bandas alternantes transparentes y opacas (u oscuras) [16]. Las ventajas prácticas de las rejillas de difracción por reflexión, es su bajo costo, predictibilidad matemática, cobertura de longitudes de onda, rango de tamaños y resolución, son tan grandes que en gran medida han reemplazado a los prismas como elementos de dispersión espectral preferidos. La alta fidelidad de los modernos métodos de replicación ha permitido la producción en masa de cualquier rejilla. Sin embargo, con las rejillas uno debe aceptar un comportamiento de eficiencia más complejo [27]. En la siguiente sección se expandirá más la prueba de Ronchi y sus propiedades.

4.3. Prueba de Ronchi

Desde su descubrimiento y aplicación en la evaluación de superficies ópticas, la prueba de Ronchi ha sido ampliamente usada tanto de forma cualitativa como cuantitativa, ya que es fácil de realizar y también de interpretar las observaciones experimentales [17].

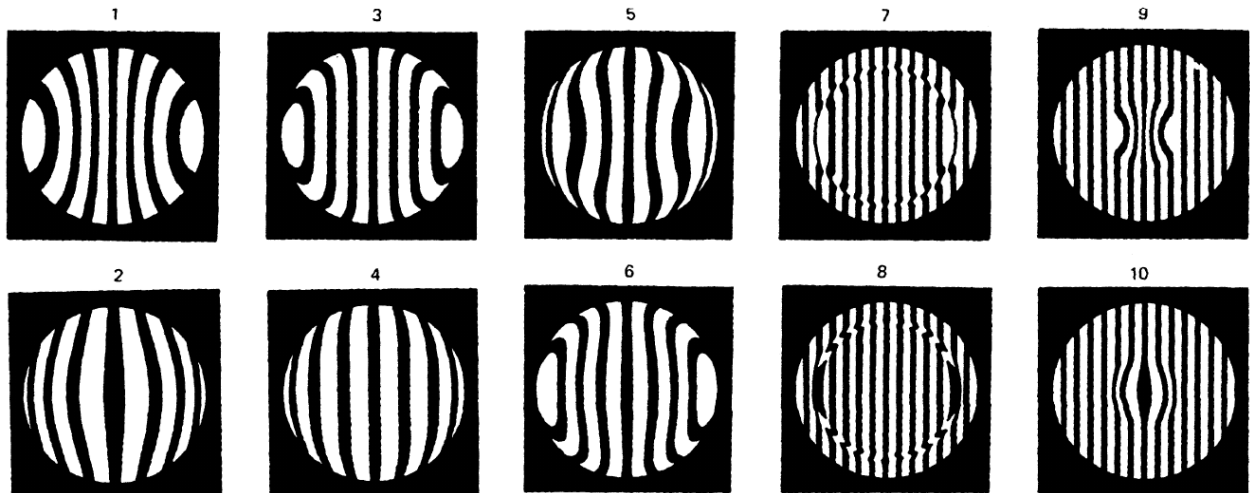


Figura 4.5: Ronchigramas de algunas deformaciones de superficies típicas [17].

La prueba de Ronchi es uno de los métodos más simples para evaluar y medir las aberraciones de un sistema óptico. Descubierta por Vasco Ronchi en 1923, al colocar una rejilla cerca del centro de curvatura de un espejo, la imagen de la rejilla se superpone en ella misma, produciendo un patrón debido a la combinación de las franjas. Ya que la forma de la combinación de estas franjas dependen de las aberraciones del espejo, V. Ronchi aplicó este fenómeno hacia la evaluación de la calidad de los espejos [17].

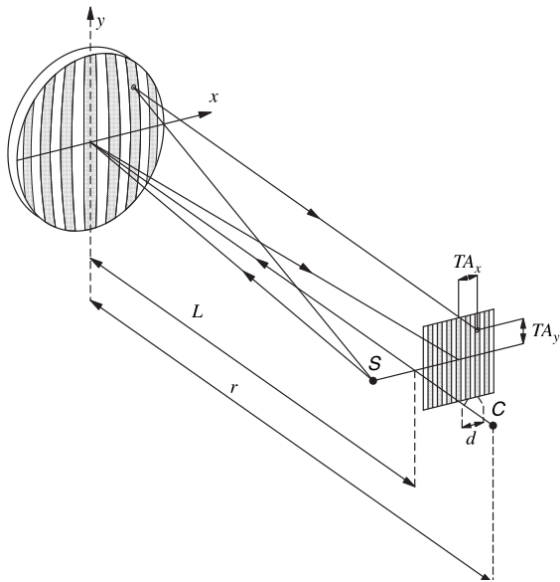
La prueba de Ronchi utiliza una rejilla de baja frecuencia, llamada “Rejilla de Ronchi”, en lugar del filo de navaja en una prueba de Foucault o del alambre en la prueba de alambre. Para un espejo con superficie esférica, el patrón observado son líneas rectas, reduciéndose el número de franjas observadas a medida en que la rejilla se acerca al foco [26].

La prueba de Ronchi tiene dos modelos equivalentes, una de ellas es la geométrica, que interpreta las franjas como sombras producidas por las rejillas; y el modelo físico, en el que ahora las franjas se producen debido a la difracción y la interferencia. Cuando la frecuencia de la rejilla no es alta, los dos modelos llegan al mismo resultado [17].

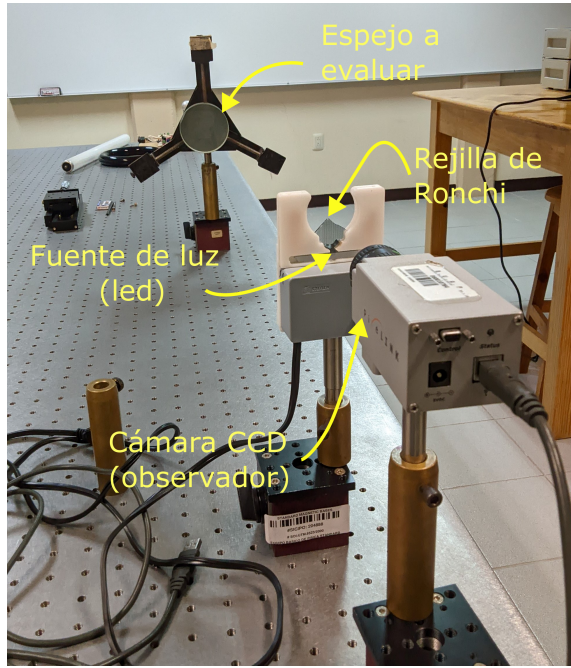
Los frentes de onda o deformaciones del espejo pueden ser determinados por el ronchigrama. Se puede tener ideas cualitativas acerca de las deformaciones del espejo que pueden ser obtenidas fácilmente por este método. Se puede observar en la Figura 4.5 algunos posibles patrones.

Un esquema geométrico de la prueba de Ronchi se muestra en la Figura 4.6a. Consta de una fuente puntual de luz S , donde los rayos viajan hacia el espejo de prueba. Algunos de estos rayos pasan a través de la rejilla, localizada justo en el mismo eje que la fuente de luz. El observador debe colocarse detrás de la rejilla justo alrededor del eje del foco del espejo. Todo este sistema debe estar cerca del centro de curvatura del espejo, denotado por C . La configuración experimental de esta prueba se muestra en la Figura 4.6b, realizada en las instalaciones del laboratorio de Física de la Universidad Tecnológica de la Mixteca.

La prueba de Ronchi mide la aberración transversal TA de forma directa, como se aprecia en 4.6a. En esta figura, tanto el objeto como la imagen están en el eje óptico, por lo que la aberración transversal se mide desde el eje y se puede observar que incluye desenfoque, así como otras aberraciones. La aberración de onda se define en la pupila de salida del sistema óptico bajo prueba,



(a)



(b)

Figura 4.6: (a) Esquema geométrico de la prueba de Ronchi [17]. (b) Configuración experimental de la prueba de Ronchi realizada en la Universidad Tecnológica de la Mixteca.

utilizando una fórmula proporcionada por Rayces [17]:

$$\frac{\partial W}{\partial x} = -\frac{TA_x}{r-W}; \quad \frac{\partial W}{\partial y} = -\frac{TA_y}{r-W}, \quad (4.1)$$

para propósitos prácticos y resultados muy precisos pueden ser obtenidos si se aproxima la fórmula 4.1 como:

$$\frac{\partial W}{\partial x} = -\frac{TA_x}{r}; \quad \frac{\partial W}{\partial y} = -\frac{TA_y}{r}, \quad (4.2)$$

donde r es la distancia desde la pupila de salida o la superficie a evaluar hasta la rejilla de Ronchi. Si el Defoco generado por la posición de la rejilla de Ronchi es pequeña, esta distancia se puede aproximar por el radio de curvatura del frente de onda. Así, si se asume una rejilla de Ronchi con espaciado d entre las líneas adyacentes de la rejilla, para un punto en el espacio (x, y) en la m -ésima franja, se puede reescribir, de forma general:

$$\frac{\partial W}{\partial x} \cos \varphi - \frac{\partial W}{\partial y} \sin \varphi = -\frac{md}{r}, \quad (4.3)$$

donde se asume que las líneas de la rejilla fueron inclinadas a un ángulo φ con respecto al eje y , como lo muestra la Figura 4.7. Esta es la fórmula básica para el modelo geométrico de la prueba de Ronchi [17].

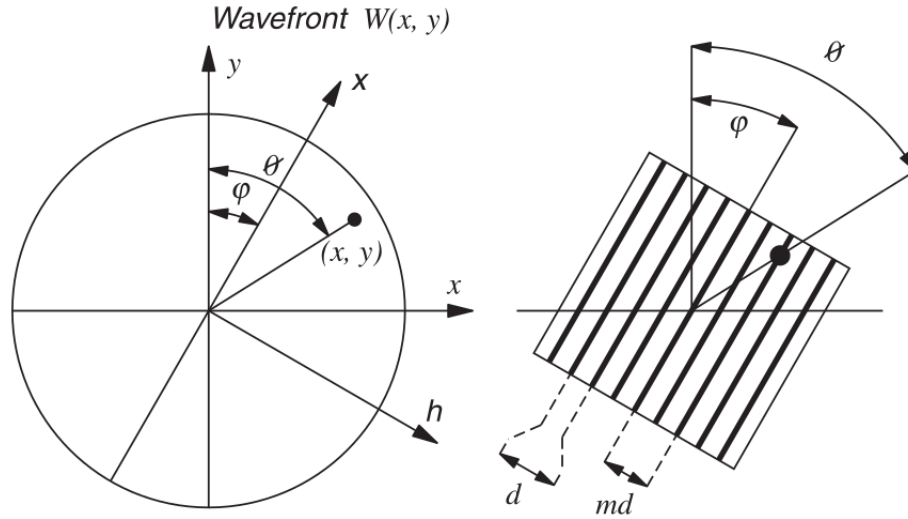


Figura 4.7: Frente de onda W y orientación de la rejilla de Ronchi [17]

4.3.1. Patrones de Ronchi para superficies esféricas

La prueba de Ronchi es útil para evaluar superficies esféricas, incluyendo espejos de gran tamaño para telescopios astronómicos. Usando la ecuación 4.3 con $\varphi = 0$ se calcula el ronchigrama ideal para cualquier superficie esférica al asumir la relación aproximada [17]:

$$z(x, y) - z_0(x, y) = 2W(x, y), \quad (4.4)$$

donde z es la sagita para la superficie esférica y z_0 es para la esfera osculadora. Este método, sin embargo, es solo una aproximación. Una alternativa o procedimiento más preciso, es el trazo de rayos usando las leyes de reflexión. Sin embargo, Sherwood (1958) y Malacara (1965) mostraron de forma independiente (y con diferentes métodos) que, cuando se usa la configuración de la Fig. 4.6a, la aberración transversal TA en el plano de la rejilla de Ronchi está dado por [17]:

$$TA(S) = \frac{(l+L-2z) \left[1 - \left(\frac{dz}{dS} \right)^2 \right] + 2 \frac{dz}{dS} \left[S - \frac{(l-z)(L-z)}{S} \right]}{\frac{l-z}{S} \left[1 - \left(\frac{dz}{dS} \right)^2 \right] + 2 \frac{dz}{dS}}, \quad (4.5)$$

donde S es la distancia desde el eje óptico a un punto en el espejo. También se puede ver en la Figura 4.6a que

$$TA = \frac{md}{\sin \theta}. \quad (4.6)$$

El ronchigrama se obtiene asignando muchos valores de S y luego calculando θ para diferentes valores de m . En general, se desea obtener el ronchigrama sobre una superficie plana, paralela y cercana al centro de curvatura del espejo, ya que esto es lo que se obtiene cuando se toma una fotografía. En la mayoría de los casos, el error introducido al considerar las franjas sobre la superficie del espejo es muy pequeño; pero cuando el radio de curvatura del espejo es pequeño en comparación con su diámetro, el error se vuelve importante.

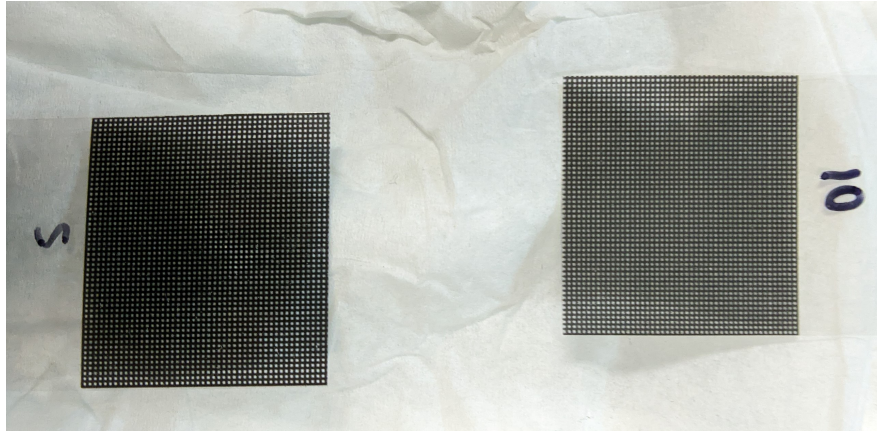
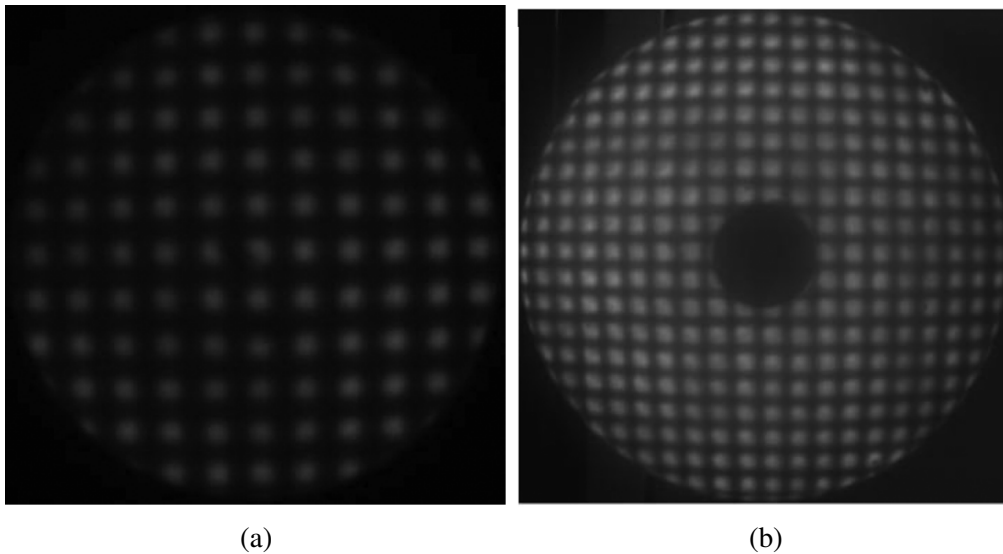


Figura 4.8: Rejillas cuadriculadas para obtener birronchigramas con la prueba de Ronchi.



(a)

(b)

Figura 4.9: (a). Birronchigrama experimental obtenido de un espejo parabólico. (b). Birronchigrama experimental obtenido de un espejo elíptico [2].

4.4. Birronchigramas.

Un birronchigrama es la imagen obtenida después de aplicar la prueba de Ronchi a una superficie como se explicó anteriormente, con la diferencia de que la rejilla utilizada es cuadrada, como se muestra en la Figura 4.8. El uso de esta rejilla dará como resultado patrones, como los mostrados, por ejemplo, en la Figura 4.9.

Capítulo 5

Algoritmos evolutivos

5.1. Introducción

El término *mejor solución* implica que hay más de una solución y que las soluciones no tienen el mismo valor. La definición de *mejor* es relativa al problema en cuestión, su método de solución y las tolerancias permitidas. Por lo tanto, la solución óptima depende de la persona que formula el problema. La educación, las opiniones, los sobornos (entendidos como influencias inapropiadas) y la cantidad de sueño son factores que influyen en la definición de lo que es *mejor*. Algunos problemas tienen respuestas o raíces exactas, y por lo tanto, *lo mejor* tiene una definición específica [28].

Anteriormente, el diseño de ingeniería se basaba en gran medida en experimentos prácticos, lo que implicaba crear sistemas reales para realizar pruebas. Esto significa que el modelo de diseño era físico, como construir un avión real para estudiar su aerodinámica en un túnel de viento. Este enfoque era complicado, costoso y lento. Con la llegada de las computadoras, los ingenieros adoptaron la simulación por computadora para analizar aspectos de sistemas reales. Esto eliminó la necesidad de modelos físicos en la etapa de diseño, ahorrando tiempo y costos. Por ejemplo, ya no se requería construir modelos físicos para estudiar la aerodinámica, como un túnel de viento para un avión. Además, esta simulación permitió explorar y descubrir nuevas características del problema, facilitando la optimización de los parámetros de diseño. Sin embargo, a pesar del uso de la computadora, los diseñadores todavía tenían que ajustar manualmente los parámetros del problema [29].

Más adelante, las personas comenzaron a usar algoritmos computacionales para que la computadora encuentre soluciones óptimas en modelos simulados. Esto llevó al desarrollo de campos como el diseño asistido por computadora, donde la computadora manipula los parámetros con poca intervención humana. Los Algoritmos Evolutivos (AE) también se volvieron populares para optimizar problemas [29].

Algunos de los primeros ejemplos de algoritmos evolutivos aparecieron en la literatura en la década de 1950. Sin embargo, tres desarrollos que tuvieron lugar en la década de 1960 han tenido el impacto más directo y duradero en el campo en la actualidad. En la Universidad Técnica de Berlín, Ingo Rechenberg y Hans-Paul Schwefel desarrollaron un algoritmo al que llamaron *estrategias evolutivas* (ES) con el fin de resolver problemas de optimización de parámetros de valores reales complejos. Al mismo tiempo, Larry Fogel y sus colegas en la Universidad de California en Los Ángeles (UCLA) desarrollaron una técnica que llamaron "programación evolutiva" (EP) para

automatizar el proceso de construcción de máquinas de estados finitos que controlaban el comportamiento de agentes inteligentes [30].

5.2. Optimización

La optimización se puede concebir como el proceso de intentar encontrar la mejor solución posible a un problema, generalmente en un tiempo limitado. Por lo que se puede decir que un problema de optimización es simplemente un problema en el que hay varias soluciones posibles y alguna forma clara de comparación entre ellas, de manera que éste existe si y solo si se dispone un conjunto de soluciones candidatas diferentes que pueden ser comparadas [31].

Los ingenieros comúnmente se enfrentan a las tareas de diseñar y operar sistemas para cumplir o superar objetivos específicos y al mismo tiempo cumplir con numerosas limitaciones impuestas al diseño y operación. La optimización es la búsqueda organizada de dichos diseños y modos de funcionamiento. Determina el conjunto de acciones o elementos que se deben implementar para lograr sistemas optimizados. En el caso más simple, la optimización busca el valor máximo o mínimo de una función objetivo correspondiente a variables definidas en un rango o espacio factible [32].

Un modelo de optimización de un solo objetivo incorpora varias expresiones matemáticas que incluyen una función objetivo y restricciones de la siguiente manera [32]:

$$\text{Optimizar } f(X), \quad X = (x_1, x_2, \dots, x_i, \dots, x_n) \quad (5.1)$$

sujeto a

$$g_j(X) < b_j, \quad j = 1, 2, \dots, m \quad (5.2)$$

$$x_i^{(L)} \leq x_i \leq x_i^{(U)}, \quad i = 1, 2, \dots, N, \quad (5.3)$$

en la cual $f(X)$ es la función objetivo; X es un grupo de variables de decisión x_i , que constituye una posible solución al problema de optimización; x_i es la i -ésima variable de decisión, N es el número de variables de decisión que determina la dimensión del problema de optimización; $g_j(X)$ es la j -ésima restricción; b_j es la constante de la j -ésima condición; m es el número total de restricciones; $x_i^{(L)}$ es el límite inferior de la variable i -ésima de decisión; y $x_i^{(U)}$ es el límite superior de la i -ésima variable de decisión.

Terminología.

Los siguientes conceptos son fundamentales para abordar un problema con alguna técnica de optimización [32]:

1. **Función objetivo.** También llamada función de costo. Constituye la meta de un problema de optimización. Esa meta debe ser maximizada o minimizada al escoger variables o variables de decisión que satisfagan todas las restricciones del sistema.
2. **Variables de decisión.** Un problema de optimización que deber ser resuelto a través de alguna técnica de optimización, depende de ciertos valores que se consideran en la entrada. El

rango de esas variables son definidas y ciertas combinaciones de ellas generarán una solución óptima. En los problemas de optimización el rango de las variables de decisión se encuentran entre el límite inferior y superior. Estas variables forman un espacio de decisión continuo.

3. **Soluciones al problema de optimización.** Cada función objetivo es expresado en términos de variables de decisión. Cuando hay solo una variable, el problema se dice que es uno-dimensional, mientras que los problemas con dos o más variables se denominan como N -dimensional. Un problema N -dimensional es expresado en términos de uno o más grupos de soluciones, tal que cada solución tienen N variables de decisión.
4. **Espacio de decisión.** El grupo de variables de decisión que satisfagan las restricciones de un problema de optimización se denomina como espacio factible de decisión. En un problema N -dimensional, cada solución posible es una variable N -vector con N elementos. Cada elemento de este vector se denomina como variable de decisión.
5. **Restricciones.** Cada problema de optimización puede tener dos tipos de restricciones. El primer tipo restringe de forma directa el valor posible de las variables de decisión. Y la segunda forma puede estar escrita en términos de fórmulas.
6. **Variables de estado.** Son variables dependientes las cuales sus valores cambian tanto como las variables de decisión lo hacen.
7. **Óptimo local y global.** Un óptimo local, se refiere a la mejor solución que tienen la función objetivo en su vecindad. En un problema de optimización 1-dimensional una variable de decisión factible X^* es un óptimo local de un problema de maximización si se cumple la condición:

$$f(X^*) \geq f(X), \quad X^* - \varepsilon \leq X \leq X^* + \varepsilon. \quad (5.4)$$

Para un problema de minimización, la condición para un óptimo local se define como

$$f(X^*) \leq f(X), \quad X^* - \varepsilon \leq X \leq X^* + \varepsilon, \quad (5.5)$$

donde X^* es un óptimo local y ε es la distancia limitada entre el vecino cercano al óptimo local X^* .

Un óptimo global es la mejor solución en el espacio de decisión. Un problema de optimización 1-dimensional con variable de decisión X y función objetivo $F(X)$, el valor X^* es un óptimo global para un problema de maximización si para cualquier variable de decisión X se cumple que

$$f(X^*) \geq f(X), \quad (5.6)$$

y en un problema de minimización se tiene que

$$f(X^*) \leq f(X). \quad (5.7)$$

8. **Soluciones cercanas al óptimo.** Un óptimo cercano tiene un valor muy próximo pero inferior al óptimo global (ver Figura 5.1). En algunos problemas de ingeniería, alcanzar el óptimo

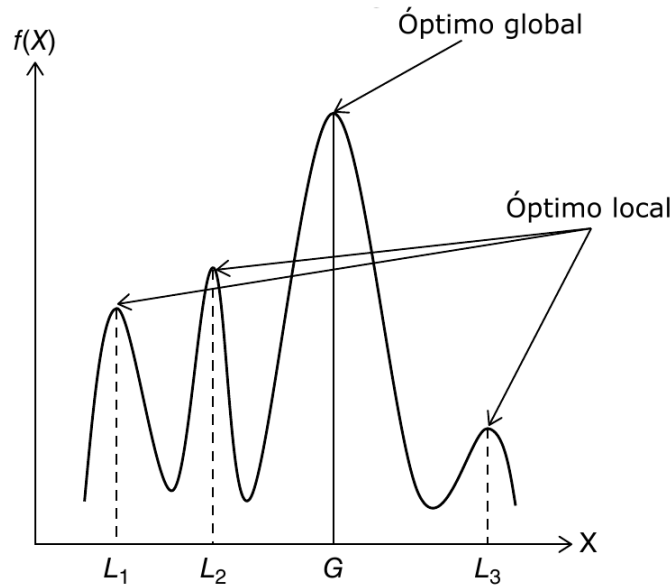


Figura 5.1: L_1 , L_2 y L_3 son óptimos locales, y G denota el óptimo global con el valor más grande de la función objetivo. El espacio de decisión puede ser unimodal o multimodal. En una superficie unimodal, solo hay un punto extremo, mientras que en un problema multimodal hay varios extremos en la superficie. En un problema unimodal, existe un único óptimo local que también es el óptimo global. Por otro lado, un problema multimodal puede incluir varios óptimos locales y globales [32].

global absoluto es extremadamente difícil o a veces imposible debido a la complejidad inherente del problema o al método utilizado para resolverlo. O también puede que acercarse al global óptimo puede ser computacionalmente prohibitivo. Un óptimo cercano es satisfactorio al resolver varios problemas del mundo real. La aproximación del óptimo cercano al óptimo global depende del problema que se esté resolviendo y del juicio del analista. Se aprecia en la Figura 5.2 el concepto del óptimo cercano en un problema de maximización.

Hay varias formas de algoritmos evolutivos que utilizan la evolución simulada como un mecanismo para resolver problemas que pueden expresarse como un problema de búsqueda u optimización, tales como los algoritmos genéticos, estrategias evolutivas, programación evolutiva, programación genética, entre otros. Los algoritmos evolutivos en el que se enfocarán este trabajo serán los algoritmos genéticos y las estrategias evolutivas.

5.3. Algoritmos genéticos

Los algoritmos genéticos fueron la primera forma de algoritmos evolutivos que fueron ampliamente aceptados a lo largo de varias disciplinas, desde investigación de operaciones a la inteligencia artificial. Hoy en día, los algoritmos genéticos y otros algoritmos evolutivos son usados de forma habitual como herramientas de búsqueda y optimización para aplicaciones científicas y de ingeniería [30].

Los algoritmos genéticos fueron ampliamente desarrollados por John Holland y sus estudiantes en las décadas de 1960, 1970 y 1980. El término de “algoritmos genéticos” entra en uso común con

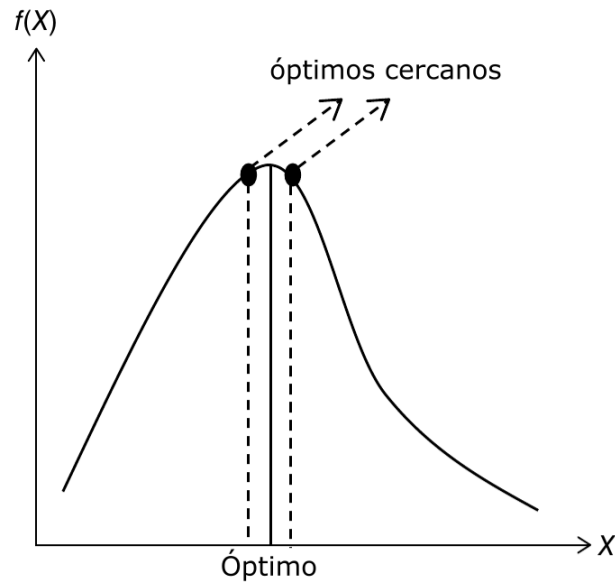


Figura 5.2: Óptimos cercanos en un problema de maximización 1-dimensional [32].

la publicación de Ken De Jongs para su disertación de doctorado [30].

Existen muchas maneras de que los algoritmos evolutivos puedan simular la evolución como mecanismo para resolver problemas, donde estos problemas pueden ser expresados como una búsqueda o un problema de optimización. Comparado con otros algoritmos evolutivos, los AG dan énfasis en la combinación de interacciones de selección, recombinación y mutación aplicado a un genotipo. Este énfasis todavía perdura en algunas formas de algoritmos genéticos. Sin embargo la hibridación de algoritmos genéticos con búsquedas locales también es muy común [30]. La idea básica detrás de un algoritmo genético (AG) es el principio Darwiniano de supervivencia de los organismos mejor adaptados, amenazados por depredadores y por peligros en su ambiente. Los miembros de esta población tienen mayor probabilidad de supervivencia que otros al adaptarse dentro de sus condiciones del entorno y su descendencia puede heredar sus rasgos y aprender habilidades de ellos, produciendo una mejor adaptación en las futuras generaciones [32].

La meta de los AG es encontrar soluciones aceptables de un problema manipulando una población de soluciones candidatas. Cada miembro de la población es típicamente representado por un solo cromosoma, el cual puede ser una simple cadena de caracteres de ceros y unos o tan complejos como un programa completo de computadora. Los AG evalúan la aptitud de cada individuo en la población usando una función de aptitud. Para mejorar soluciones pobres, soluciones mejores son escogidas aleatoriamente con un mecanismo de selección [33].

Los AG son estocásticos, lo que hace que sean confiables y capaces de estimar un óptimo global para un problema dado es el proceso de mantener las mejores soluciones en cada generación usándolas para implementar otras. Entonces toda la población comienza a mejorar, generación por generación. La cruce entre individuos da a lugar al aprovechamiento del área de soluciones generada entre los dos padres dados [34].

También este algoritmo se beneficia de la mutación de genes. Este operador genera aleatoriamente cambios en los genes de los cromosomas, los cuales mantienen la diversidad en la población e incrementa el comportamiento de exploración del AG. Similarmente en la naturaleza, el operador

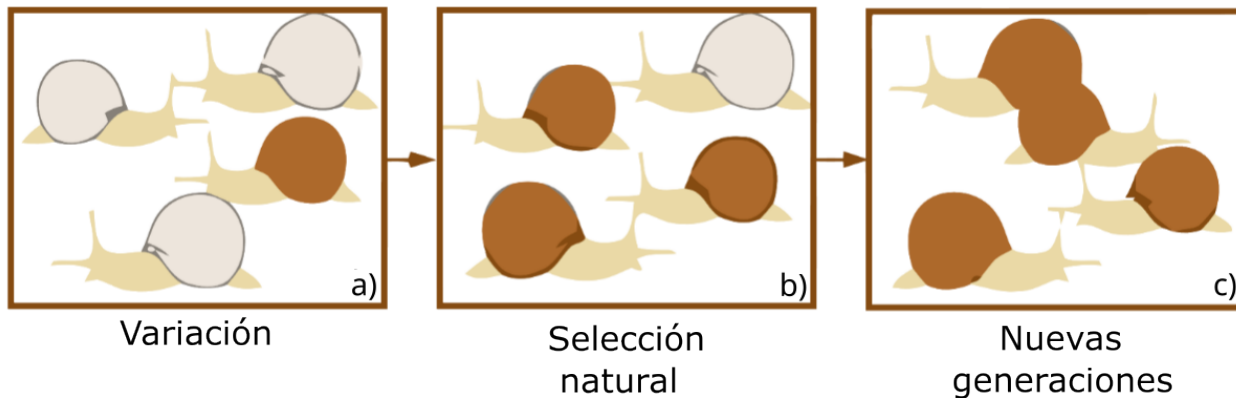


Figura 5.3: a) Algunos individuos muestran variaciones. Estas variaciones muestran ventajas sobre otros. b) Los mejor adaptados son favorecidos y seleccionados. c) La variación incrementa en frecuencia entre la población.

de mutación puede darnos sustancialmente una mejor solución y promover otras soluciones hacia el óptimo global [34]. Un ejemplo de este efecto se aprecia en la Figura 5.3, en donde la selección natural actúa modificando genes de forma que esa mutación favorece a la población total y se transmite de generación en generación.

5.3.1. Terminología

La palabra "genético" viene del griego *gènesis* y significa *crecer*. A. Q. H. Badar [35] describe los siguientes conceptos biológicos los cuales nos ayudarán a entender mejor a los AG.

- **Cromosoma.** La información genética de cualquier organismo vivo está guardado en sus cromosomas. Los cromosomas en sí están compuestos por partes más pequeñas llamados genes.
- **Genes.** Los genes son responsables de determinar las características de un individuo. Un gen representa una característica/variable de entrada independiente.
- **Conjunto.** El conjunto determina todas las diferentes combinaciones posibles que pueden formar los genes. Cuanto mayor sea el tamaño del conjunto de genes, mejores serán las variaciones disponibles para las generaciones más nuevas.
- **Genoma.** El conjunto de todos los genes de una especie en particular se llama genoma. Se preocupa más por la disposición de los genes. En términos biológicos, el cromosoma y el genoma pueden ser diferentes; sin embargo, en los algoritmos genéticos, se consideran iguales.
- **Mitosis.** En la mitosis, la información genética es copiada desde los padres a la descendencia.
- **Meiosis.** Combinación de información diferente de los padres para formar un nuevo individuo/descendencia.
- **Fenotipo.** Es el término usado para representar valores de la función objetivo.

En cada iteración de los AG, los cromosomas o soluciones tienen que pasar a través de los siguientes procesos (específicamente con los AG):

1. **Selección.** Método usado para seleccionar cierto número de individuos, a los que llamaremos padres, de un conjunto de posibles soluciones disponibles para la cruce/reproducción.
2. **Cruza.** Es un proceso que genera descendencia a través de la cruce de los cromosomas de los padres.
3. **Mutación.** Esta etapa es usada para introducir variaciones en un cromosoma existente por medio de cambios en algunos genes.

El AG binario puede funcionar con codificaciones binarias, pero la función de costo requiere variables continuas. Siempre que se evalúa la función de costo, primero se debe decodificar el cromosoma. Un ejemplo de un cromosoma codificado binario que tienen N_{var} variables, cada una codificada con $N_{genes} = 10$ bits es [28]:

$$cromosoma = \left[\underbrace{1111001001}_{gen1} \underbrace{0011011111}_{gen2} \dots \underbrace{0000101001}_{genN_{var}} \right].$$

Sustituyendo cada gen en este cromosoma nos otorga un arreglo de una versión cuantizada de las variables. Este cromosoma tiene un total de $N_{bits} = N_{genes} \times N_{var}$ bits.

En el caso de un AG de valores reales, se define un cromosoma como un arreglo de varias variables a ser optimizadas. Si el cromosoma tiene N_{var} variables (un problema N-dimensional de optimización) dados por $p_1, p_2, \dots, p_{N_{var}}$ entonces los cromosomas se escriben como un arreglo de $1 \times N_{var}$ elementos, tal que [28]:

$$cromosoma = [p_1, p_2, p_3, \dots, p_{N_{var}}]. \quad (5.8)$$

En este caso, los valores de las variables son representados como números de punto flotante. Cada cromosoma tiene un costo que se calcula al evaluar la función de costo f en las variables $p_1, p_2, p_3, \dots, p_{N_{var}}$:

$$costo = f(cromosoma) = f(p_1, p_2, p_3, \dots, p_{N_{var}}). \quad (5.9)$$

¿Qué sucede si intenta resolver un problema en el que los valores de las variables son continuos y desea conocerlos con toda la precisión de la máquina? En tal problema cada variable requiere muchos bits para representarla. Si el número de variables es grande, el tamaño del cromosoma también lo será. Por supuesto, los 1 y 0 no son la única forma de representar una variable. En principio, se podría utilizar cualquier representación imaginable para codificar las variables. Cuando las variables se cuantizan de forma natural, el AG binario encaja muy bien. Sin embargo, cuando las variables son continuas, es más lógico representarlasm mediante números de punto flotante. Además, dado que el AG binario tiene su precisión limitada por la representación binaria de variables, el uso de números de punto flotante permite fácilmente la representación con precisión de la máquina. El AG continuo también tiene la ventaja de requerir menos almacenamiento que el AG binario porque un único número de punto flotante representa la variable en lugar de números enteros de N bits. Es inherentemente más rápido que el AG binario, porque los cromosomas no tienen que ser decodificados antes de la evaluación de la función de costo [28].

5.3.2. Elementos que conforman un Algoritmo Genético

Población inicial

Los AG inician con una población aleatoria. Esta población puede ser generada a partir de una distribución aleatoria Gaussiana para incrementar la diversidad. Esta población debe incluir múltiples soluciones, los cuales representan cromosomas de individuos. El principal objetivo en el paso de inicialización es el que las soluciones cercanas del espacio de búsqueda sea lo más uniforme posible para incrementar la diversidad de la población y tener mejores probabilidades para encontrar regiones prometedoras de soluciones óptimas [29].

Selección

La selección natural es la principal inspiración de este componente para el algoritmo genético (AG). En la naturaleza, los individuos más aptos tienen una mayor probabilidad de obtener alimento y aparearse. Esto hace que sus genes contribuyan más en la producción de la siguiente generación de la misma especie [29]. Inspirándose en esta idea simple, el algoritmo genético puede utilizar algunos de los siguientes métodos de selección para crear la siguiente generación proporcional a sus valores de aptitud.

Selección por ruleta. En la selección por el método de la ruleta, cada individuo elegible es asignado con una probabilidad. La probabilidad de la i -ésimo cromosoma se evalúa usando [35]:

$$p(i) = \frac{f(i)}{\sum_{j=1}^n f(j)}, \quad (5.10)$$

donde $f(i)$ es el valor de la función objetivo para el i -ésimo cromosoma y n representa el número de cromosomas disponibles para la selección. Se necesita generar un número aleatorio de acuerdo con la distribución de probabilidad dada por los $p(i)$ [36]:

1. Se definen las puntuaciones acumuladas de la siguiente forma:

$$\begin{aligned} q_0 &:= 0, \\ q_i &:= p_1 + \dots + p_i \quad (\forall i = 1, \dots, n). \end{aligned} \quad (5.11)$$

2. Se genera un número aleatorio $a \in [0, 1]$.
3. Se selecciona al individuo i que cumpla:

$$q_{i-1} < a < q_i. \quad (5.12)$$

Este proceso se repite para cada individuo que se desee seleccionar.

Para ejemplificar el método, se consideran las probabilidades descritas en la tabla 5.1. la suma de funciones objetivo se calcula y se presenta en la última fila. En este caso se encuentra que la suma es 100. Luego, dividiendo todos los valores de la función objetivo por esta suma, se obtiene la probabilidad de cada cromosoma. La suma de los valores de la función objetivo puede diferir, pero la suma de probabilidades siempre será 1. En base a las probabilidades anteriores se forma una ruleta como en la Figura 5.4a. Los cromosomas obtienen su asignación en la ruleta en función del valor o

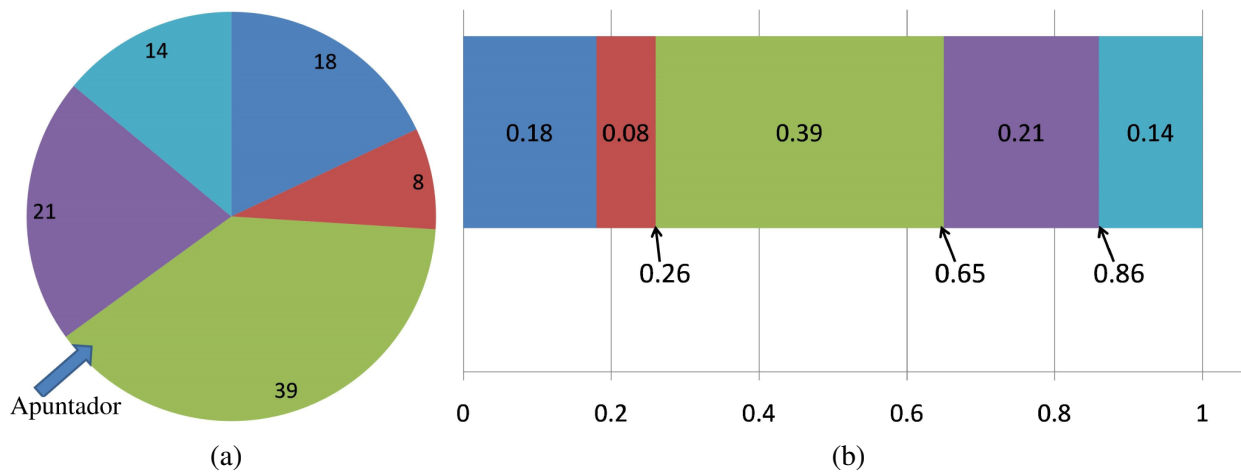


Figura 5.4: (a): Rueda de la ruleta vista en forma real. (b): Representación de la ruleta con los datos de la tabla 5.1 [35].

probabilidad de su función objetivo. Los valores presentados en la Ruleta están en porcentajes, es decir, sobre 100. La asignación de la rueda de la Ruleta se puede representar prácticamente como en la Figura 5.4b. Con esta representación, las probabilidades de todos los cromosomas se distribuyen en rangos de valores específicos. Para el cromosoma 1 el rango de valores es de 0 a 0.18, mientras que para el cromosoma 2 el rango es de 0.18 a 0.26. Para el cromosoma 3, el rango de números es de 0.26 a 0.65 con una diferencia de 0.39 y así sucesivamente. Esto distribuye los rangos entre los cromosomas en función de sus probabilidades. Se genera un número aleatorio entre 0 y 1 y se selecciona en el grupo de apareamiento el cromosoma en cuyo rango se encuentra el número aleatorio [35].

Tabla 5.1: Selección por ruleta. Valores de ejemplo.

Núm. cromosoma	Valor de la función	Probabilidad
1	18	0.18
2	8	0.08
3	39	0.39
4	21	0.21
5	14	0.14
suma	100	1

Selección por torneo. Se elige aleatoriamente una pequeña muestra de la población, de ella se selecciona al individuo de mejor valor de adaptación. Normalmente se utiliza un tamaño pequeño para la muestra, de 2 o 3 individuos. El proceso se repite hasta contemplar el número de individuos que se desee seleccionar. Esta selección puede realizarse de forma determinista o probabilística [36].

- En la selección por torneo determinista se selecciona aleatoriamente a un número p de individuos. Después entre los individuos seleccionados se elige el mejor, es decir, el de mejor valor de adaptación.

- El caso probabilístico es similar, excepto porque, una vez seleccionados los p individuos, en lugar de escoger siempre al mejor, la selección se hace con una cierta probabilidad. Si un número, que se genera aleatoriamente entre 0 y 1, es mayor que un cierto umbral especificado como un parámetro del algoritmo, entonces se elige al mejor, en otro caso al peor.

Selección por rango. En el método de selección por rango [35], los cromosomas elegibles para entrar en el grupo de apareamiento se organizan según su valor de función objetivo. El cromosoma con el mejor valor de función objetivo obtiene el rango más alto, mientras que el peor cromosoma obtiene el rango 1. El rango más alto es igual al número de cromosomas elegibles para entrar en el grupo de apareamiento, asumamos que sea n .

La probabilidad de cada cromosoma se evalúa por medio de la ecuación:

$$p(i) = \frac{\text{rango}(i)}{n \times (n - 1)}. \quad (5.13)$$

En la tabla 5.2, siendo un ejercicio de minimización, los cromosomas se ordenan en orden ascendente según los valores de la función objetivo. No es necesario convertir el problema de minimización en uno de maximización. Los rangos permanecerán iguales incluso cuando los cromosomas cambien, es decir, el cromosoma de mayor rango siempre tendrá una probabilidad del 0.25 y el cromosoma de menor rango siempre tendrá 0.05, y esto se aplica a todos los demás cromosomas en el grupo de apareamiento. Una vez que las probabilidades están asociadas con los cromosomas, entonces los cromosomas aleatorios pueden ser seleccionados de manera similar al método de selección de Ruleta. La distribución de los cromosomas se presentan en la Figura 5.5. De acuerdo

Tabla 5.2: Tabla de valores en donde se evalúa y selecciona una población por medio del método de rango [35].

Núm. Cromosoma	Valor de f	Rango	Probabilidad
3	0.025641026	5	0.25
4	0.047619048	4	0.2
1	0.055555556	3	0.15
5	0.071428571	2	0.1
2	0.125	1	0.05

con este esquema, el número aleatorio puede variar de 0 a 0.75, ya que la suma de todas las probabilidades es 0.75. El rango del primer cromosoma es de 0 a 0.25, para el segundo será de 0.26 a 0.45, y así sucesivamente.

Selección Natural. Se decide qué cromosomas de la población son lo suficientemente aptos para sobrevivir y posiblemente reproducir descendencia en la próxima generación. Los costos numéricos (evaluaciones de los individuos) y los cromosomas asociados se clasifican de acuerdo al criterio de maximización o minimización. Los no aptos desaparecen. Este proceso de selección natural debe ocurrir en cada iteración del algoritmo para permitir que la población de cromosomas evolucione a lo largo de las generaciones hacia los miembros más aptos según la función de costo. No todos los sobrevivientes se consideran lo suficientemente aptos para aparearse. De los N cromosomas dados en una generación, solo los $N' < N$ mejor adaptados se mantienen para aparearse y el resto se descarta para dejar espacio a las nuevas descendencias [28].

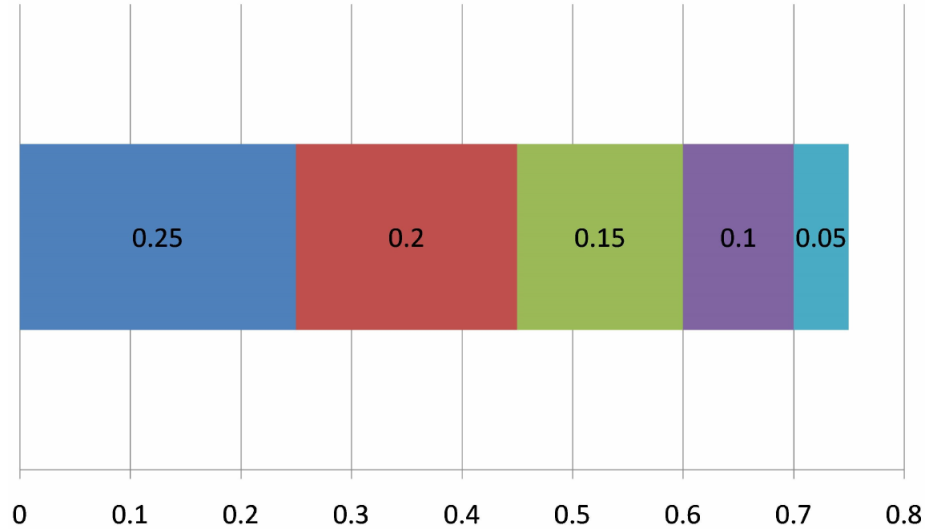


Figura 5.5: Ejemplificación del método de selección por rango [35].

Cruza

Después de seleccionar a los individuos mediante un operador de selección, estos individuos deben emplearse para crear la nueva generación. En la naturaleza, los cromosomas en los genes de un macho y una hembra se combinan para producir un nuevo cromosoma [29]. Los métodos de selección descritos anteriormente se utilizan para seleccionar cromosomas para el apareamiento. La cruce se utiliza para combinar las características de dos o más cromosomas para producir los cromosomas hijos [35]. En la literatura, existen diferentes técnicas para el operador de cruce.

Cruce monopunto. En el cruce de un solo punto, los cromosomas parentales se dividen en dos en un punto dado. Luego, las partes correspondientes de los cromosomas se intercambian para formar dos cromosomas hijos a partir de dos cromosomas parentales. El punto de cruce puede ser fijo, elegido al azar, basado en alguna función u otra técnica adecuada [35].

Cruce de doble punto. En el cruce de doble punto, en cambio, hay dos puntos de cruce y sólo se intercambian los cromosomas entre los puntos [29].

La Figura 5.6 muestra ambos tipos de cruce de forma esquematizada.

El problema con los métodos de cruce de punto es que no se introduce nueva información: cada valor continuo que fue iniciado aleatoriamente en la población inicial se propaga a la siguiente generación, solo en combinaciones diferentes. Aunque esta estrategia funciona bien para representaciones binarias, en el caso de las variables reales continuas, y en este continuo, simplemente estamos intercambiando dos puntos de datos. Estos enfoques dependen totalmente de la mutación para introducir nuevo material genético. Para el caso del AG de números reales, los métodos de cruce resuelven este problema al encontrar formas de combinar los valores de las variables de los dos padres en nuevos valores de variables en la descendencia. Un solo valor de variable descendiente, p_{nuevo} , proviene de una combinación de los dos valores de variable descendiente correspondientes [28]:

$$p_{nuevo} = \beta p_{mn} + (1 - \beta) p_{dn}, \quad (5.14)$$

donde β es un número aleatorio en el intervalo $[0, 1]$, p_{mn} es la n -ésima variable en el cromosoma madre y p_{dn} es la n -ésima variable en el cromosoma padre. Este método de mezcla combinan

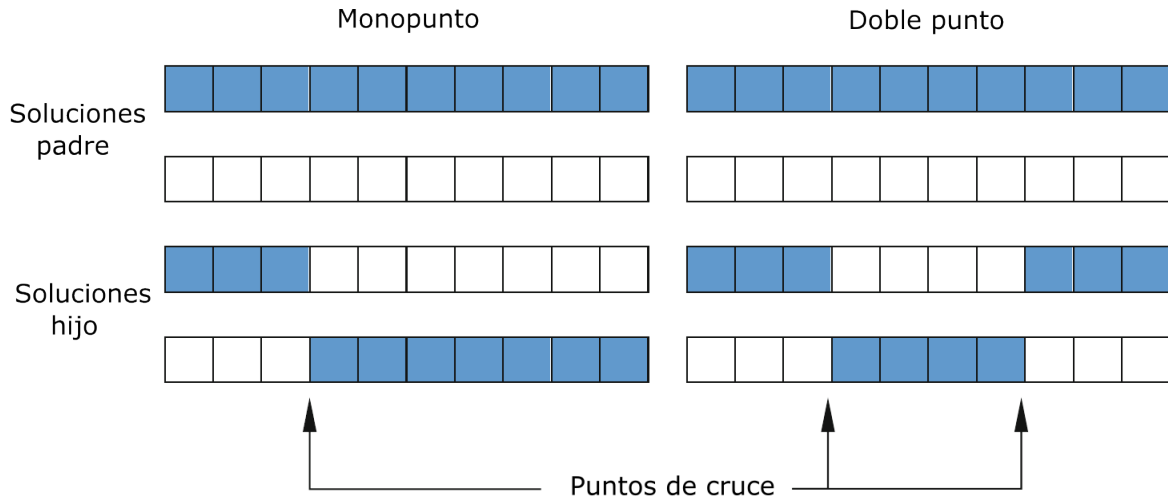


Figura 5.6: Dos técnicas populares de cruce en los AG: cruce de un punto o monopunto y cruce de doble punto. En el cruce de un punto, los cromosomas de dos soluciones parentales se intercambian antes y después de un solo punto. En el cruce de doble punto, hay dos puntos de cruce y solo se intercambian los cromosomas entre los puntos [29].

eficazmente la información de los dos padres y eligen valores de las variables entre los valores encuadrados por los padres; sin embargo, no permiten la introducción de valores más allá de los extremos ya representados en la población. Para lograr esto se requiere un método de extrapolación. El más simple de estos métodos es el cruce lineal. En este caso, tres hijos son generados de dos padres de la siguiente manera:

$$\begin{aligned}
 p_{nuevo1} &= 0.5p_{mn} + 0.5p_{dn}, \\
 p_{nuevo2} &= 1.5p_{mn} - 0.5p_{dn}, \\
 p_{nuevo3} &= -0.5p_{mn} + 1.5p_{dn}.
 \end{aligned}
 \tag{5.15}$$

Cualquier variable fuera de los límites se descarta a favor de las otras dos. Luego, se eligen los dos mejores descendientes para propagarse. Por supuesto, el factor 0.5 no es el único que se puede usar en este método. La cruce heurística es una variación donde algún número aleatorio, β , se elige dentro del intervalo $[0, 1]$ y las variables de los descendientes está dado por

$$p_{nuevo} = \beta(p_{mn} - p_{dn}) + p_{mn}.
 \tag{5.16}$$

Variaciones de este enfoque incluyen elegir cualquier número de variables para modificar y generar diferentes valores de β para cada variable. Este método también permite generar descendencia fuera de los valores de las dos variables parentales.

El método utilizado en el algoritmo genético de este trabajo está basado en una combinación del método de extrapolación con el método de cruce, mismo que usa R. L. Haupt y S. E. Haupt para desarrollar su algoritmo [28]. Se quiere encontrar una manera de imitar de cerca las ventajas del esquema de apareamiento de los algoritmos genéticos binarios. Comienza seleccionando al azar una variable en el primer par de padres para que sea el punto de cruce

$$\alpha = \text{round}\{\text{rand} * N_{\text{var}}\},
 \tag{5.17}$$

donde α es un número aleatorio que no rebase del tamaño del número de genes del individuo, la función *round* se refiere a redondear el producto de un número aleatorio $rand \in (0, 1)$ por el número de variables N_{var} . Entonces se define

$$\begin{aligned} padre_1 &= [p_{m1}, p_{m2}, \dots, p_{m\alpha}, \dots, p_{mN_{var}}], \\ padre_2 &= [p_{d1}, p_{d2}, \dots, p_{d\alpha}, \dots, p_{dN_{var}}], \end{aligned} \quad (5.18)$$

donde m y d son subíndices para diferenciar a la madre y al padre. Luego, las variables seleccionadas se combinan para formar nuevas variables que aparecerán en los descendientes:

$$\begin{aligned} p_{nuevo1} &= p_{m\alpha} - \beta [p_{m\alpha} - p_{d\alpha}] \\ p_{nuevo2} &= p_{d\alpha} - \beta [p_{m\alpha} - p_{d\alpha}] \end{aligned} \quad (5.19)$$

donde β es también un número aleatorio entre 0 y 1. El paso final es completar la cruce con el resto de los cromosomas, es decir:

$$\begin{aligned} descendencia_1 &= [p_{m1}, p_{m2}, \dots, p_{nuevo1}, \dots, p_{dN_{var}}] \\ descendencia_2 &= [p_{d1}, p_{d2}, \dots, p_{nuevo2}, \dots, p_{mN_{var}}] \end{aligned} \quad (5.20)$$

Si se selecciona la primera variable de los cromosomas, solo se intercambian las variables a la derecha de la variable seleccionada. Si se selecciona la última variable de los cromosomas, solo se intercambian las variables a la izquierda de la variable seleccionada. Este método no permite variables descendientes fuera de los límites establecidos por los padres, a menos que se cumpla que $\beta > 1$.

Mutación

La mutación es el último operador evolutivo, en el cual uno o varios genes son alterados después de crear soluciones hijos. La tasa de mutación se establece baja en el AG porque tasas de mutación altas convierten el AG en una búsqueda aleatoria primitiva. El operador de mutación mantiene la diversidad de la población al introducir otro nivel de aleatoriedad. De hecho, este operador evita que las soluciones se vuelvan similares y aumenta la probabilidad de evitar soluciones locales en el AG [29]. La Figura 5.7 muestra de forma gráfica el proceso de mutación, después de ocurrida la cruce.

Siguiendo el procedimiento de R. L. Haupt y S. E. Haupt [28], se elige una tasa de mutación que se definirá como m_r . Luego se multiplicará por el número total de variables que pueden ser mutados. Después se eligen números aleatorios para seleccionar la ubicación en las filas y columnas de las variables que se van a mutar. Una variable mutada es reemplazada por una nueva variable aleatoria elegida dentro del rango. Por ejemplo, si se usa un porcentaje de mutación m_r del 20%, y sean 7 individuos de 2 variables cada uno, se tiene $0.20 \times 7 \times 2 \approx 3$ mutaciones. Los siguientes pares serán aleatoriamente seleccionadas:

$$\begin{aligned} m_{row} &= [4, 4, 7], \\ m_{col} &= [1, 2, 1]. \end{aligned}$$

Siendo m_{row} los valores en las posiciones de fila y m_{col} los valores en las posiciones de columna. El primer par aleatorio es (4, 1), tal que el valor en la fila 4 y la columna uno de la matriz de población se reemplaza por un número aleatorio entre los límites establecidos por el problema.

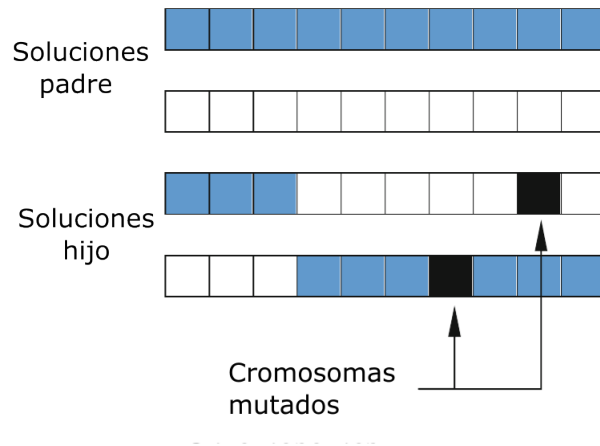


Figura 5.7: Diagrama que muestra el funcionamiento del operador mutación, este altera uno o múltiples genes en las soluciones hijo después de la fase de cruce [29].

La siguiente generación

El proceso descrito se itera hasta que una solución aceptable se encuentre.

Algoritmo y pseudocódigo

Se presenta el diagrama de flujo de un Algoritmo Genético simple en la Figura 5.8 . El proceso de optimización empieza con la generación de cromosomas aleatorios dentro de la población. Después de generarlos, el proceso de iteración inicia. El primer paso dentro del bucle es obtener los valores de la función objetivo para cada cromosoma. Realizado lo anterior, se procede con la creación de nuevas cromosomas. Los padres son seleccionados a través de algún proceso de selección, luego se realiza la cruce y por último la mutación. La nueva población de cromosomas ahora serán usados para reemplazar a cromosomas existentes. El proceso se repite hasta que el criterio de terminación se cumpla.

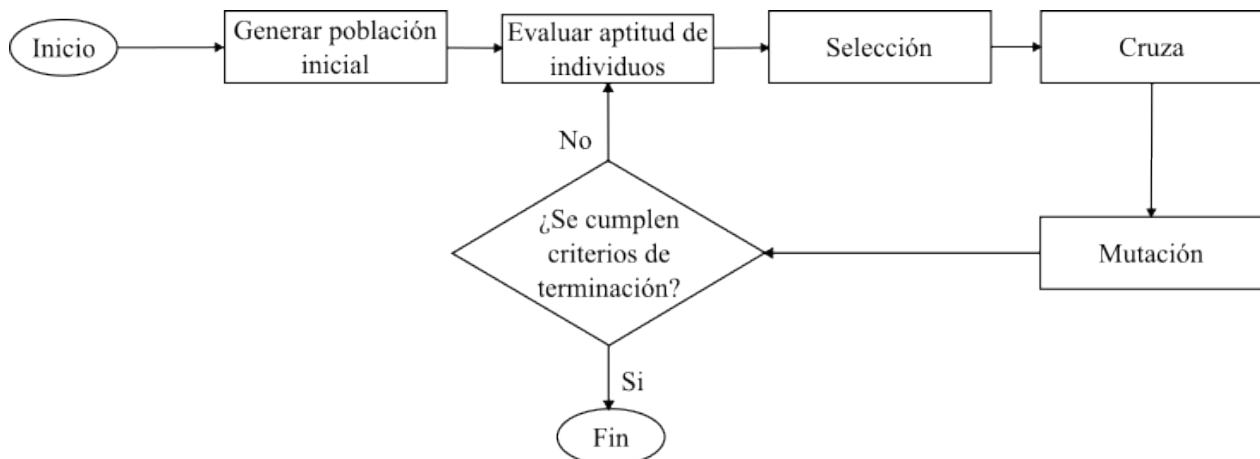


Figura 5.8: Diagrama de flujo de un algoritmo genético simple.

5.4. Estrategias evolutivas.

Las estrategias evolutivas son descritas como una especialización de los algoritmos evolutivos. Durante la inicialización, se crea la primera generación que puede consistir en uno o más individuos, y se evalúan las aptitudes de éstos. Después de la inicialización, se entra al ciclo evolutivo. La recombinación crea nuevos individuos de la población padre, denominados descendencias. Se suelen distinguir dos tipos principales de recombinación: recombinación dominante y recombinación intermedia. En la recombinación dominante, una propiedad de un individuo padre se hereda en la descendencia, es decir, esta propiedad domina sobre la propiedad correspondiente de los otros individuos. En la recombinación intermedia, se tienen en cuenta las propiedades de todos los individuos, de modo que, por ejemplo, en el caso más simple, se utiliza su valor promedio [37].

El operador de mutación proporciona la principal fuente de variación de la descendencia en una estrategia evolutiva. Basándose en el muestreo de variables aleatorias, se modifican las propiedades de los individuos. Luego, se evalúan los individuos recién creados, es decir, se calculan sus valores de aptitud. Usando estos valores de aptitud, la selección identifica un subconjunto de individuos que forman la nueva población que se utiliza en la siguiente iteración del ciclo evolutivo. El ciclo se termina en función de un criterio de terminación establecido por el usuario, como alcanzar un número máximo de evaluaciones, alcanzar un valor de aptitud objetivo o estancamiento en el proceso de búsqueda [37].

Las estrategias evolutivas son particularmente adecuados para tareas de optimización no lineal, se podría decir que son una instanciación de los algoritmos genéticos y son caracterizados por las siguientes propiedades [37]:

- La selección de los individuos para la recombinación es imparcial.
- La selección es un proceso determinista.
- Los operadores de mutación son parametrizados y por lo tanto pueden cambiar sus propiedades durante la optimización.
- Los individuos consisten en la decisión de parámetros normales tanto como de parámetros estratégicos.

5.4.1. Mutación

Se presentan tres principios rectores para el diseño de operadores de mutación [37]:

1. Cualquier punto del espacio de búsqueda debe ser alcanzable con una probabilidad estrictamente mayor que cero mediante un número finito de aplicaciones de mutación.
2. La mutación debe ser imparcial, lo cual se puede lograr utilizando una distribución de máxima entropía.
3. El operador se parametriza, de manera que se pueda controlar la amplitud de la variación.

En \mathbb{R}^n esos requerimientos son cumplidos con la distribución normal multivariable. Un vector aleatorio n -dimensional \mathbf{X} es multivariable normalmente distribuido con expectación $\bar{\mathbf{x}} \in \mathbb{R}^n$ y una

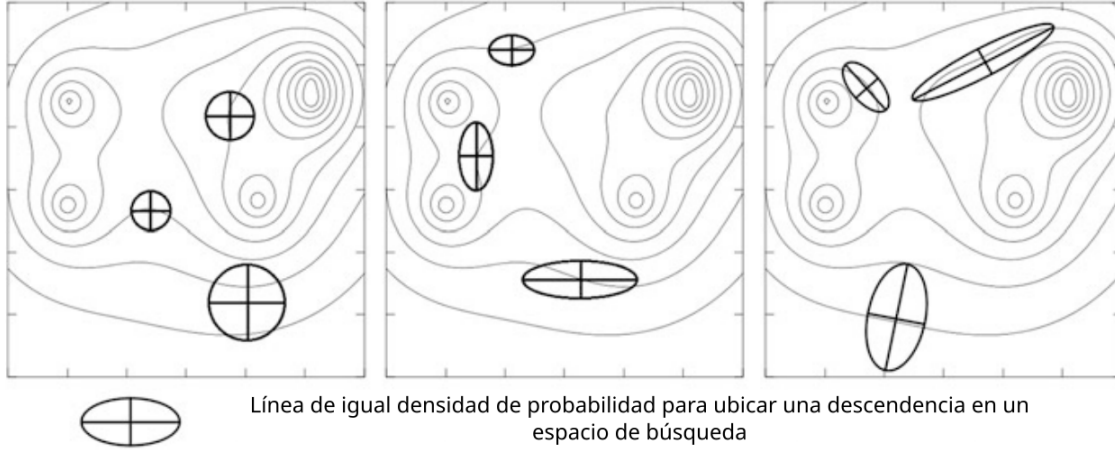


Figura 5.9: Elipsoides de mutación que representan $N(\mathbf{0}, \mathbf{I})$, $N(\mathbf{0}, \text{diag}(\delta^2))$ y $N(\mathbf{0}, \mathbf{C})$ (de izquierda a derecha) [37].

matriz de covarianza definida y positiva $\mathbf{C} \in \mathbb{R}^{n \times n}$ si su densidad de probabilidad es definida de acuerdo a [37]:

$$f_X(x) = \frac{1}{(2\pi)^{n/2}(\det \mathbf{C})^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \bar{\mathbf{x}})^T \mathbf{C}^{-1}(\mathbf{x} - \bar{\mathbf{x}})\right), \quad (5.21)$$

la cual típicamente se escribe como $\mathbf{X} \sim N(\bar{\mathbf{x}}, \mathbf{C})$, donde $N(\bar{\mathbf{x}}, \mathbf{C})$ denota la distribución normal multivariada en su forma general. En las ecuaciones matemáticas, $N(\bar{\mathbf{x}}, \mathbf{C})$ es usado típicamente con un vector, refiriéndose a vector como un muestreo de acuerdo a la distribución dada. También es posible simplificar su escritura y que se estará utilizando de la siguiente manera:

$$\mathbf{x}' = \mathbf{x} + N(\mathbf{0}, \mathbf{C}). \quad (5.22)$$

Dada la propiedad positiva de la matriz de covarianza \mathbf{C} , la siguiente eigendescomposición existe:

$$\mathbf{C} = \mathbf{B}\mathbf{D}^2\mathbf{B}^T, \quad (5.23)$$

donde \mathbf{B} denota una matriz ortogonal, las cuales sus columnas son eigenvectores de \mathbf{C} . $N(\bar{\mathbf{x}}, \mathbf{C})$ se reduce a la distribución $N(\mathbf{0}, \mathbf{I})$ por medio de la eigendescomposición de la ecuación 5.23, de acuerdo a

$$N(\bar{\mathbf{x}}, \mathbf{C}) \sim \bar{\mathbf{x}} + \mathbf{B}\mathbf{D}N(\mathbf{0}, \mathbf{I}). \quad (5.24)$$

En el campo de las estrategias evolutivas, los tres casos especiales $N(\mathbf{0}, \mathbf{I})$, $N(\mathbf{0}, \text{diag}(\delta^2))$ y $N(\mathbf{0}, \mathbf{C})$ son usados por definición en la mayoría de algoritmos. En la Figura 5.9 se muestra un bosquejo de las elipsoides de mutación correspondientes, es decir, isóneas de las funciones de densidad de probabilidad, incorporados a un espacio solución hipotético bidimensional [37].

El caso más simple para generar una mutación \mathbf{x}' de parte de \mathbf{x} está basado en el uso de las matrices $\mathbf{B} = \mathbf{I}$ y $\mathbf{D} = \sigma\mathbf{I}$ con un tamaño de paso global de $\delta \in R^+$ para las matrices \mathbf{B} y \mathbf{D} como se usan en la ecuación 5.24:

$$\mathbf{x}' = \mathbf{x} + \delta \cdot N(\mathbf{0}, \mathbf{I}), \quad (5.25)$$

esta expresión corresponde a esferas con radio individual definido por δ , como se muestra en la primera imagen de la Figura 5.9 (de izquierda a derecha). En el caso de una distribución descendiente se le denomina como isotrópica [37].

Para convertir las esferas en elipsoides anisotrópicos, con los ejes principales paralelos a los ejes coordenados, como se aprecia en medio de la Figura 5.9, la matriz \mathbf{D} en la ecuación 5.24 debe convertirse a una matriz diagonal $\boldsymbol{\delta} = (\delta_1, \dots, \delta_n)^T \in \mathbb{R}^n$ con diferentes entradas en la diagonal principal. Como en el caso anterior, \mathbf{B} es una matriz diagonal de tal manera que [37]:

$$\begin{aligned}\mathbf{x}' &= \mathbf{x} + \mathbf{I} \text{diag}(\boldsymbol{\delta})N(\mathbf{0}, \mathbf{I}) \\ &= \mathbf{x} + N(\mathbf{0}, \text{diag}(\boldsymbol{\delta}^2))\end{aligned}\tag{5.26}$$

Las relaciones de longitud de los ejes principales de los elipsoides de mutación dependen de las relaciones entre los componentes correspondientes del vector $\boldsymbol{\delta}$. Una rotación de los hiperelipsoides de mutación con respecto a los ejes de coordenadas, como se muestra en la parte derecha de la Figura 5.9, se logra utilizando una matriz de covarianza \mathbf{C} con entradas fuera de la diagonal diferentes de cero. Este caso se denota con el término mutación correlacionada. A diferencia de los dos casos anteriores, la matriz \mathbf{B} no es solo una matriz identidad [37]:

$$\begin{aligned}\mathbf{x}' &= \mathbf{x} + \mathbf{B} \text{diag}(\boldsymbol{\delta})N(\mathbf{0}, \mathbf{I}) \\ \mathbf{x}' &= \mathbf{x} + \mathbf{B}N(\mathbf{0}, \text{diag}(\boldsymbol{\delta}^2)) \\ \mathbf{x}' &= \mathbf{x} + N(\mathbf{0}, \mathbf{C})\end{aligned}\tag{5.27}$$

La elección de uno de los tres casos explicados anteriormente tiene un impacto directo en la complejidad de los parámetros endógenos que controlan la distribución normal multivariada. En general, si n denota la dimensionalidad del espacio de búsqueda, el número de parámetros de estrategia endógenos en el caso de la ecuación 5.25 es $O(1)$ es decir, constante. En el caso de 5.26 se requiere un vector de tamaño $O(n)$ de parámetros endógenos y la adaptación de una matriz de covarianza arbitraria, es decir una matriz simétrica de $n \times n$ de acuerdo a la ecuación 5.27, requiere $O(n^2)$ parámetros endógenos [37].

5.4.2. Técnicas avanzadas de adaptación

Todos los mecanismos de control o adaptación suponen tácitamente que el alcance operativo de una estrategia evolutiva se encuentra en las proximidades de su posición actual $\bar{x} \in \mathbb{R}^n$ donde la función de aptitud es aproximada de forma apropiada por una expansión de Taylor de segundo orden si el problema es diferenciable, lo que significa:

$$f(x) = f(\bar{x}) + (x - \bar{x})' \nabla f(\bar{x}) + \frac{1}{2} (x - \bar{x})' \nabla^2 f(\bar{x}) (x - \bar{x}),\tag{5.28}$$

donde $\nabla f(\bar{x})$ es el gradiente y $\nabla^2 f(\bar{x})$ es la matriz Hessiana, denotada por \mathbf{H} . Para una función cuadrática f , la cual es simétrica y positiva definida. Si la función de aptitud es cuadrática, entonces la expansión de Taylor es exacta y la matriz \mathbf{H} contiene información sobre el escalado y la orientación de los iso-hiperelipsoides que especifican posiciones en el espacio de búsqueda con el mismo valor de aptitud. Si la expansión de Taylor no es exacta, entonces la matriz Hessiana proporciona al menos una aproximación razonable. Hansen describe la relación entre la matriz Hessiana

\mathbf{H} y la matriz de covarianza \mathbf{C} de una distribución $N(\mathbf{0}, \mathbf{C})$ de manera informal. Argumenta que usando $\mathbf{C} = \mathbf{H}^{-1}$ para optimizar una función cuadrática es equivalente a usar $\mathbf{C} = \mathbf{I}$ para la optimización de una función isotrópica, tal como la función esfera $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}\mathbf{x}^T$. En otras palabras, adaptando una matriz de covarianza simplifica la optimización al transformar la función objetivo a una función isotrópica [30, 37].

5.4.3. Adaptación de tamaños de paso basados en Frecuencia de éxitos.

Los primeros experimentos (Schwefel 1964) han llevado a la observación de que el tamaño del paso no debe fijarse durante la búsqueda para lograr un buen rendimiento. Intuitivamente, es fácil aceptar el argumento de que los tamaños de los pasos deberían reducirse cuanto más se acerca la estrategia al óptimo; de lo contrario, la mayoría de las mutaciones no conducirían a mejoras. En realidad, existe un tamaño de paso óptimo dependiendo de la posición actual y de la distancia hasta el óptimo. Dado que no se conoce la posición del óptimo, se requiere un mecanismo inteligente para obtener una pista si el tamaño del paso actual es demasiado grande o demasiado pequeño [30].

Rechenberg (1973) analizó teóricamente dos funciones objetivo extremadamente diferentes (“modelo de esfera” y “modelo del corredor”) con el resultado de que el mayor progreso hacia el óptimo se logra para un tamaño de paso que conduce a una frecuencia relativa de mutaciones exitosas (es decir, mutaciones que conducen a un mejor valor de aptitud) de aproximadamente 0.270 para el modelo de esfera y aproximadamente 0.184 para el modelo de corredor. Este hallazgo llevó a la conjetura de que para una función de aptitud arbitraria, el tamaño del paso debería ajustarse a una longitud que conduce a una frecuencia de éxito entre los dos valores extremos. Eligiendo $0.2 = 1/5$ como cercanías entre los extremos de cada caso, el análisis de Rechenberg llegó al primer mecanismo de control del tamaño de paso en las estrategias evolutivas, llamada la regla de $1/5$ de éxitos. Fue implementada de la siguiente manera: Contar el número de mutaciones exitosas s dentro de cierto periodo de tiempo p . Si s/p es menor que $1/5$, entonces el tamaño de paso es demasiado grande y debe ser decrementado por un factor $1/\gamma < 1$. De otro modo, el tamaño de paso debe ser muy pequeño y debe ser incrementado. Claramente, p debe ser múltiplo de 5 y $\gamma = (\frac{20}{17})^{1/n}$ [30].

5.4.4. Estrategia Evolutiva (1+1)

Inspirados en las clases acerca de la evolución biológica, los estudiantes Hans-Paul Schwefel, Ingo Rechenberg, y Peter Bienert se propusieron a desarrollar un método de solución inspirado en los principios de variación y selección. En su primera versión, utilizaron un ciclo evolutivo muy simple sin ningún parámetro endógeno. Este algoritmo genera una sola descendencia

$$\mathbf{x}' = \mathbf{x} + (N_1(0, \sigma), \dots, N_n(0, \sigma))^T = \mathbf{x} + \sigma \cdot N(\mathbf{0}, \mathbf{I}) \quad (5.29)$$

de un único individuo padre $\mathbf{x} \in \mathbb{R}^n$. El valor de σ fue elegido de manera que la primera aproximación de esta estrategia fuera exitosa. Se implementó la conjetura anterior de que la velocidad máxima de convergencia (es decir, la velocidad del progreso de la optimización) se logra cuando $1/5$ de todas las mutaciones son exitosas, es decir, mejoras sobre su padre. Si cerca de $1/5$ de todas las mutaciones son exitosas, el tamaño de paso es óptimo y no hay adaptación requerida. Si la tasa de éxitos cae por valores menores de $1/5$, el tamaño de paso requiere ser reducido. En cambio, si crece más de $1/5$ el tamaño de paso necesita incrementarse [37].

Para obtener el tamaño de paso nuevo $\sigma' = \sigma \cdot c^{\{-1,1\}}$, el valor previo de σ , incrementa o disminuye, al multiplicar o dividirlo por un valor c entre 0.817 y 1. El valor recomendado de 0.817 fue deducido por Schwefel de acuerdo a argumentos teóricos acerca de la velocidad del tamaño de paso. La adaptación del tamaño de paso de acuerdo a lo anterior, es aplicarlo cada n iteraciones del algoritmo, y teniendo una tasa de éxitos se mide en un rango de por lo menos $10 \cdot n$ mutaciones. El algoritmo 1 detalla mejor el proceso de mutación del individuo [37].

Datos: Individuo inicial \mathbf{x} , número de evaluaciones n .

Resultado: Población aleatoria.

$P_0 \leftarrow \{x\};$

$\phi \leftarrow f(x);$

$p_s \leftarrow 0;$

Crear arreglo A que guarde las mutaciones exitosas;

$t \leftarrow 0;$

mientras $t \leq n$ **hacer**

$t \leftarrow t + 1;$

$\mathbf{x}' \leftarrow \mathbf{x} + \sigma \cdot \mathbf{N}(\mathbf{0}, \mathbf{I});$

$\phi' \leftarrow f(\mathbf{x}');$

si $\phi' < \phi$: **entonces**

$\mathbf{x} \leftarrow \mathbf{x}';$

$\phi \leftarrow \phi';$

 Guardar éxito en A ;

en otro caso

 Guardar fracaso en A ;

fin

$P_t \leftarrow \{\mathbf{x}\};$

si $t \bmod n = 0$: **entonces**

 obtener número de éxitos y de fracasos de al menos $10n$ entradas en A ;

$p_s = \frac{\text{éxitos}}{\text{éxitos} + \text{fracasos}};$

$\sigma' \leftarrow \begin{cases} \sigma \cdot c & \text{si } p_s < 1/5 \\ \sigma/c & \text{si } p_s > 1/5 \\ \sigma & \text{si } p_s = 1/5 \end{cases}$

fin

$\sigma \leftarrow \sigma';$

fin

Algoritmo 1: Pseudocódigo de la estrategia evolutiva (1+1) como método de minimización [37].

5.4.5. Control de covarianzas

La primera estrategia que utilizó y adaptó una matriz de covarianza completa fue la estrategia evolutiva con mutaciones correlacionadas (denominado CORR-ES) propuesto por Schwefel en 1981. Conjeturó que sería óptimo alinear un hiperelipsoide de iguales probabilidades de la distribu-

ción de mutación con un hiperelipsoide de iguales valores de aptitud si el problema se describe adecuadamente mediante una expansión de Taylor de segundo orden. Si esto es cierto, entonces sería óptimo establecer la matriz de covarianza \mathbf{C} proporcional a la matriz Hessiana inversa de la expansión de Taylor. Esto se ve en la función de densidad de probabilidad de la distribución normal multivariada expresada en la ecuación 5.21 donde las isolíneas de probabilidades iguales están determinadas básicamente por la expresión exponencial. Por lo tanto, si \mathbf{H} es una matriz Hessiana de la expansión de Taylor, entonces $\mathbf{C}^{-1} = \mathbf{H}$ es necesaria para obtener hiperelipsoides igualmente alineados, o equivalentemente, $\mathbf{C} = \mathbf{H}^{-1}$ [30].

5.4.6. Estrategia Evolutiva (1+1)-Cholesky-CMA

La estrategia evolutiva (1+1)-Cholesky-CMA introduce un método de adaptación de la matriz de covarianza \mathbf{C} de forma implícita, utilizando la eigendescomposición de \mathbf{C} . En consecuencia, la aproximación al óptimo reduce la complejidad computacional dentro de cada aproximación [37].

El algoritmo se basa en el método llamado *Descomposición de Cholesky* de la matriz de covarianza $\mathbf{C} = \mathbf{A}\mathbf{A}^T$. La actualización de los factores de Cholesky \mathbf{A} es posible sin el conocimiento explícito de la matriz \mathbf{C} . El correspondiente lema menciona que, para cualquier vector $\mathbf{v} \in \mathbb{R}^n$ y que $\zeta = \frac{1}{\|\mathbf{v}\|}(\sqrt{1 + \|\mathbf{v}\|^2} - 1)$, la siguiente ecuación establece que [37]:

$$\mathbf{I} + \mathbf{v}\mathbf{v}^T = (\mathbf{I} + \zeta\mathbf{v}\mathbf{v}^T)(\mathbf{I} + \zeta\mathbf{v}\mathbf{v}^T). \quad (5.30)$$

Este lema se requiere para la demostración del siguiente teorema: Sea $\mathbf{C} \in \mathbb{R}^n$ una matriz simétrica, positiva y definida por medio de la descomposición de Cholesky $\mathbf{C} = \mathbf{A}\mathbf{A}^T$. Sea $\mathbf{C}' = \alpha\mathbf{C} + \beta\mathbf{v}\mathbf{v}^T$ una actualización de \mathbf{C} con $\mathbf{v}, \mathbf{z} \in \mathbb{R}^n$, $\mathbf{v} = \mathbf{A}\mathbf{z}$ y $\alpha, \beta \in \mathbb{R}^+$. El factor de Cholesky actualizado \mathbf{A}' de \mathbf{C}' está dado por:

$$\mathbf{A}' = \sqrt{\alpha}\mathbf{A} + \frac{\sqrt{\alpha}}{\|\mathbf{z}\|} \left(\sqrt{1 + \frac{\beta}{\alpha}\|\mathbf{z}\|^2} - 1 \right) (\mathbf{A}\mathbf{z})\mathbf{z}^T. \quad (5.31)$$

Basado en un individuo padre \mathbf{x} , una descendencia o hijo \mathbf{x}' se crea de acuerdo a

$$\mathbf{x}' = \sigma\mathbf{A}\mathbf{z}, \quad \text{con } \mathbf{z} = N(\mathbf{0}, \mathbf{I}). \quad (5.32)$$

Usando el teorema anterior, el factor de Cholesky \mathbf{A} se adapta de la siguiente manera:

$$\mathbf{A}' = c_a\mathbf{A} + \frac{c_a}{\|\mathbf{z}\|^2} \left(\sqrt{1 + \frac{(1 - c_a^2)\|\mathbf{z}\|^2}{c_a^2}} - 1 \right) \mathbf{A}\mathbf{z}\mathbf{z}^T, \quad (5.33)$$

con un parámetro constante de estrategia exógeno c_a . La adaptación anterior se aplica si el valor \bar{p}_s es menor que un valor de umbral p_t . La adaptación del tamaño de paso global δ es similar al valor de 1/5, que es la tasa de éxitos de la estrategia evolutiva (1+1). Si el hijo es mejor que el padre, $\lambda_s = 1$, de lo contrario, $\lambda_s = 0$, entonces se realiza la adaptación:

$$\bar{p}_s = (1 - c_p)\bar{p}_s + c_p\lambda_s. \quad (5.34)$$

Estos indicadores son acumulados a lo largo de las generaciones utilizando una tasa de aprendizaje c_p , resultando en una tasa de éxitos acumulados \bar{p}_s . Utilizando esta medición y su valor objetivo

p_s^t , para la tasa de éxitos, el tamaño de paso global σ , se actualiza de la siguiente manera:

$$\sigma' = \sigma \cdot \exp\left(\frac{1}{d} \left(\bar{p}_s - \frac{p_s^t}{1 - p_s^t} (1 - \bar{p}_s)\right)\right). \quad (5.35)$$

El pseudocódigo se muestra en el algoritmo 2 y la configuración predeterminada de los parámetros exógenos de la estrategia son [37]:

$$\begin{aligned} p_s^t &= \frac{2}{11}, \\ p_t &= \frac{11}{25}, \\ c_a &= \sqrt{1 - \frac{2}{n^2 + 6}}, \\ c_p &= \frac{1}{12}, \\ d &= 1 + \frac{1}{n}. \end{aligned}$$

Datos: Individuo inicial \mathbf{x} , tamaño de paso σ .

Resultado: Individuo óptimo.

$\mathbf{A} \leftarrow \mathbf{I}$;

$\bar{p}_s \leftarrow p_s^t$;

condición \leftarrow Falso

mientras condición = Falso **hacer**

$t \leftarrow t + 1$;

$\mathbf{z} \leftarrow N(\mathbf{0}, \mathbf{I})$;

$\mathbf{x}' \leftarrow \mathbf{x} + \sigma \mathbf{A} \mathbf{z}$;

si $f(\mathbf{x}') \leq f(\mathbf{x})$: **entonces**

$\lambda_s \leftarrow 1$;

en otro caso

$\lambda_s \leftarrow 0$;

fin

$\bar{p}_s \leftarrow (1 - c_p) + c_p \lambda_s$;

$\sigma' \leftarrow \sigma \cdot \exp\left(\frac{1}{d} \left(\bar{p}_s - \frac{p_s^t}{1-p_s^t} (1 - \bar{p}_s)\right)\right)$;

si $f(\mathbf{x}') \leq f(\mathbf{x})$: **entonces**

$\mathbf{x} \leftarrow \mathbf{x}'$;

si $\bar{p}_s \leq p_t$ **entonces**

$\mathbf{A} \leftarrow c_a \mathbf{A} + \frac{c_a}{\|\mathbf{z}\|^2} \left(\sqrt{1 + \frac{(1-c_a^2)\|\mathbf{z}\|^2}{c_a^2}} - 1 \right) \mathbf{A} \mathbf{z} \mathbf{z}^T$

fin

fin

 Verificar condición (Falso o Verdadero)

fin

Algoritmo 2: Pseudocódigo de la estrategia evolutiva (1+1)-Cholesky-CMA como método de minimización [37]. .

Capítulo 6

Desarrollo del software basado en la prueba de Ronchi para la evaluación de espejos.

6.1. Generalidades

6.1.1. Introducción

En el desarrollo de software de optimización, la eficacia y la flexibilidad son imperativas para enfrentar los desafíos que plantean los problemas de optimización complejos. La arquitectura del software ha sido cuidadosamente concebida para ofrecer una solución integral, proporcionando no solo una implementación robusta de algoritmos de optimización avanzados, sino la alta capacidad de personalización de parámetros que el usuario pueda usar de manera intuitiva.

En esta sección, se explora la anatomía que sustenta el software de evaluación de espejos. Desde la entrada de datos hasta la presentación de resultados, cada componente ha sido diseñado con la intención de que el software sea lo más óptimo posible, contribuyendo de manera sinérgica a su rendimiento general. Este diseño modular, además de mejorar la eficiencia, también brinda una plataforma escalable que puede adaptarse a las demandas del usuario y otras propuestas en el área de optimización e implementación de nuevos métodos tanto de evaluación de espejos como de algoritmos evolutivos.

6.1.2. Arquitectura general del software

La arquitectura del software de optimización se ha diseñado con el objetivo de proporcionar una plataforma robusta y eficiente para abordar problemas complejos de optimización. La estructura del software se organiza en módulos clave que trabajan de manera concertada para ofrecer funcionalidades avanzadas y escalabilidad. Esta estructura se aprecia en la Figura 6.1. A continuación, se describen los componentes principales del software:

1. La prueba de Ronchi, que es la simulación de birronchigramas dados los parámetros del espejo.
2. El método de normalización, que consiste en que las intensidades de la imagen del birronchigrama experimental son escaladas matemáticamente en un rango de cero a uno.

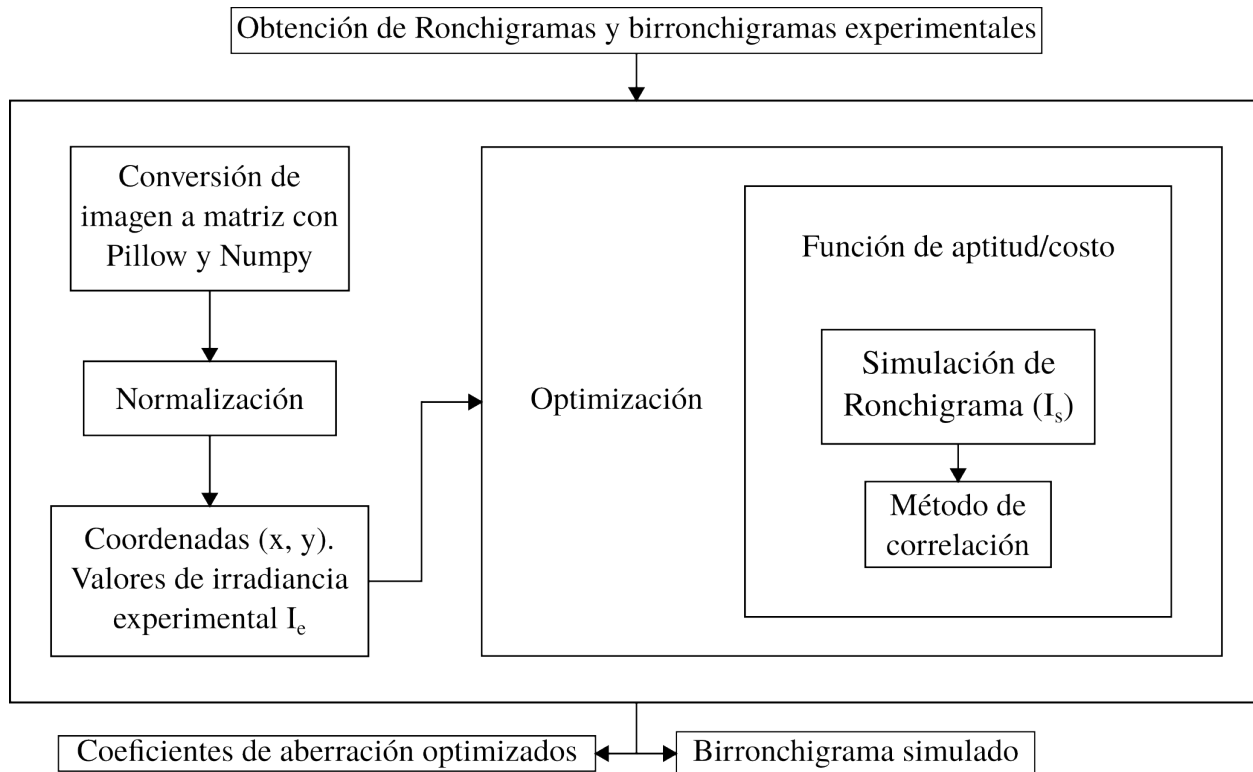


Figura 6.1: Arquitectura general del software para evaluación de espejos cóncavos.

3. El método de optimización basado en programación de un algoritmo genético híbrido.
4. El método de correlación de imágenes, el cual forma una parte de la función de costo del método de optimización.

La arquitectura del software se ha diseñado con un enfoque en la modularidad y la extensibilidad, permitiendo que el sistema evolucione con los avances en técnicas de algoritmos genéticos, estrategias evolutivas y los requerimientos que el usuario desee modificar.

6.1.3. Lenguajes y herramientas utilizadas

Para este software de evaluación se empleó el lenguaje de programación de Python, el cual es un lenguaje interpretado, interactivo y orientado a objetos. Incorpora módulos, excepciones, tipado dinámico, tipos de datos de muy alto nivel y clases. Soporta múltiples paradigmas de programación mas allá del orientado a objetos, por ejemplo, el procedural y la programación funcional. Python combina un poder notable con sintaxis muy clara. Tiene interfaces para muchas bibliotecas y llamadas al sistema, así como para varios sistemas de ventanas, y es extensible en C o C++. También se puede utilizar como lenguaje de extensión para aplicaciones que necesitan una interfaz programable. Finalmente, Python es portátil: se ejecuta en muchas variantes de Unix, incluidos Linux y macOS, y en Windows [38].

El uso de las siguientes bibliotecas fueron clave para la elaboración del programa:

Pillow

La biblioteca de imágenes de Python [39] agrega capacidades de procesamiento de imágenes al intérprete de Python. Proporciona una amplia compatibilidad con formatos de archivo, una representación interna eficiente y capacidades de procesamiento de imágenes bastante potentes.

La biblioteca de imágenes principal (PIL por sus siglas en inglés) está diseñada para un acceso rápido a los datos almacenados en unos pocos formatos de píxeles básicos. Es ideal para aplicaciones de procesamiento por lotes y archivo de imágenes. Se puede utilizar la biblioteca para crear miniaturas, convertir entre formatos de archivo, imprimir imágenes, etc.

La biblioteca contiene funcionalidades básicas de procesamiento de imágenes, incluyendo operaciones de punto, filtrado con un conjunto de núcleos de convolución integrados y conversiones de espacios de color.

NumPy

NumPy [40] es la biblioteca de código abierto de programación de matrices principal para el lenguaje Python. Tiene un papel esencial en los procesos de análisis de investigación en campos diversos como física, química, astronomía, geociencias, biología, psicología, ciencia de materiales, ingeniería, finanzas y economía.

NumPy proporciona un objeto de matriz Python multidimensional junto con funciones compatibles con matrices que operan en él. Debido a su simplicidad inherente, la matriz NumPy es el formato de intercambio de facto para datos de matriz en Python. NumPy opera en matrices en memoria utilizando la unidad central de procesamiento (CPU). Para utilizar hardware y almacenamiento modernos y especializados, recientemente ha habido una proliferación de paquetes de matrices basados en esta biblioteca.

La matriz NumPy es una estructura de datos que almacena y accede de manera eficiente a matrices multidimensionales (también conocidas como tensores) y permite una amplia variedad de cálculos científicos. Consiste en un apuntador a la memoria, junto con metadatos utilizados para interpretar los datos almacenados allí, en particular “tipo de datos”, “forma” y “pasos”.

Matplotlib

Matplotlib es una biblioteca de trazado basada en Python con soporte completo para gráficos en 2D y soporte limitado para gráficos en 3D, ampliamente utilizada en la comunidad de cómputo científico de Python. La biblioteca abarca una amplia gama de casos de uso. Puede incrustar gráficos en el conjunto de herramientas de interfaz de usuario que elija y actualmente admite gráficos interactivos en todos los principales sistemas operativos de escritorio utilizando las herramientas GTK+, Qt, Tk, FLTK, wxWidgets y Cocoa. Se puede llamar de forma interactiva desde la consola de Python para producir gráficos con comandos simples y procedurales, muy parecidos a Mathematica, IDL o MATLAB. Matplotlib también se puede incrustar en un servidor web sin interfaz gráfica para proporcionar copias en formato rasterizado como Portable Network Graphics (PNG) y en formatos vectoriales como PostScript, Portable Document Format (PDF) y Scalable Vector Graphics (SVG) que se ven excelentes en papel [41].

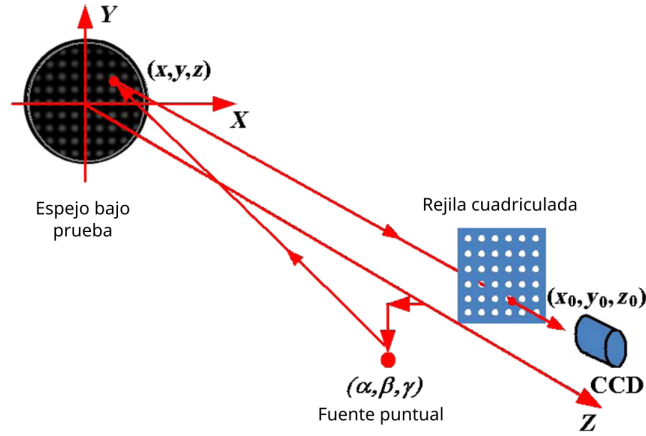


Figura 6.2: Geometría de la prueba de Ronchi usando una rejilla cuadriculada [2].

SciPy

SciPy es una biblioteca de rutinas numéricas para el lenguaje de programación Python que proporciona componentes fundamentales para modelar y resolver problemas científicos. SciPy incluye algoritmos de optimización, integración, interpolación, problemas de valores propios, ecuaciones algebraicas, ecuaciones diferenciales y muchas otras clases de problemas; también proporciona estructuras de datos especializadas, como matrices dispersas y árboles de k dimensiones. SciPy está construido sobre NumPy, que proporciona estructuras de datos de matriz y rutinas numéricas rápidas relacionadas. SciPy es en sí mismo la base sobre la cual se construyen bibliotecas científicas de nivel superior, incluyendo scikit learn y scikit-image [42].

6.1.4. Consideraciones de eficiencia y rendimiento.

La eficiencia de este software de evaluación recae directamente en el método de optimización, en donde se van a contemplar diferentes opciones de programación. Se consideró utilizar un algoritmo genético híbrido utilizando un algoritmo genético de valores reales y alguna de las estrategias evolutivas presentes en este trabajo.

6.2. Simulación de Birronchigramas.

A. Cordero Dávila y J. González García desarrollaron un algoritmo para correlacionar birronchigramas simulados y experimentales [2]. En este artículo se evalúan errores de superficie sin simetría aparente a través de analizar la correlación existente entre birronchigramas simulados y experimentales. La Figura 6.2 muestra la geometría de la prueba de Ronchi con una rejilla cuadriculada. Un rayo incidente proviene de una fuente puntual de luz (α, β, γ) hacia un punto (x, y, z) en la superficie. El rayo reflejado atraviesa el plano de la rejilla cuadriculada ubicada en $z = z_0$ hacia el punto coordenado (x_0, y_0) dado por:

$$x_0 = x + (z_0 - z) \times \left\{ \frac{(x - \alpha)(1 - z_x^2 + z_y^2) - 2z_x[z_y(y - \beta) + (\gamma - z)]}{(\gamma - z)(1 - z_x^2 - z_y^2) + 2[z_x(x - \alpha) + z_y(y - \beta)]} \right\}, \quad (6.1)$$

$$y_0 = y + (z_0 - z) \times \left\{ \frac{(y - \beta)(1 - z_x^2 + z_y^2) - 2z_y[z_x(x - \alpha) + (\gamma - z)]}{(\gamma - z)(1 - z_x^2 - z_y^2) + 2[z_x(x - \alpha) + z_y(y - \beta)]} \right\}, \quad (6.2)$$

z_x y z_y son las derivadas parciales de la sagita con respecto a x y a y . z representa la ecuación de la sagita superficial. Esta ecuación está dada por la suma de la sagita ideal mas el error de superficie, es decir:

$$z(x, y) = z_{id}(x, y) + w(x, y). \quad (6.3)$$

El rayo incidente con irradiancia unitaria atraviesa la rejilla, Esta irradiancia I se define como:

$$I = \left[\frac{1 + \cos 2\pi x_0/d}{2} \right] \left[\frac{1 + \cos 2\pi y_0/d}{2} \right], \quad (6.4)$$

donde d es el periodo de la rejilla cuadrada en la dirección del eje x o y .

El simulador de birronchigramas es necesario para obtener el valor ideal de intensidad dados los parámetros del espejo, para luego comparar estos valores con los de intensidad del birronchigrama normalizado.

Inicialmente, se construyó una función denominada como `zernike_params`, que hace la delimitación dentro de la pupila unitaria cuyas coordenadas internas serán evaluadas con los polinomios de Zernike, además del cálculo de sus derivadas parciales. Los argumentos de entrada requeridos son las coordenadas de la imagen y el valor de irradiancia para cada par de puntos (x, y) . Como salida de esta función son las coordenadas en el eje x y y , el valor de la irradiancia dentro de la pupila unitaria, la evaluación de los polinomios de Zernike con las coordenadas anteriores, sus derivadas y al final el parámetro $r_0^2 = x^2 + y^2$ que es un cálculo que se usará para las futuras partes del programa general.

El programa de simulación de birronchigramas está compuesto por los siguientes procesos:

1. **Definir parámetros iniciales del espejo.** Es decir, los parámetros de curvatura y radio del espejo, además de la distancia del espejo a la rejilla, y las coordenadas de la irradiancia de entrada, obtenidos de la función `zernike_params`.
2. **Escalar valores de los coeficientes.** El algoritmo escala los valores de los coeficientes propuestos. Cada valor de los coeficientes tiene su propia escala con un factor $\lambda = 50 \times 10^{-6}/r$; en este caso r corresponde al radio del espejo.
3. **Calcular el error w .** Se realiza la suma del producto entre el valor de los coeficientes escalados previamente, y de los valores de los polinomios de Zernike obtenidos de la función `zernike_params`.
4. **Calcular el valor de la sagita.** Que corresponde al valor de la ecuación de la sagita 6.3, siendo z la ecuación de la sagita ideal mostrada en la ecuación 2.8, pero sin los términos de asfericidad, y el error w calculado anteriormente.
5. **Calcular las derivadas de la sagita.** Se guarda un vector con los valores calculados de las derivadas parciales, tanto de x como de y de la sagita. Por regla de la cadena, las derivadas parciales respecto a las variables, se definen como:

$$\begin{aligned} \frac{\partial z}{\partial x} &= \frac{\partial z}{\partial \rho} \frac{\partial \rho}{\partial x}, \\ \frac{\partial z}{\partial y} &= \frac{\partial z}{\partial \rho} \frac{\partial \rho}{\partial y}, \end{aligned} \quad (6.5)$$

mientras que las respectivas derivadas de la sagita respecto a ρ y a los ejes x y y se definen como:

$$\begin{aligned}\frac{\partial z}{\partial \rho} &= \frac{c\rho}{\sqrt{1 - (K+1)(c^2\rho^2)}}, \\ \frac{\partial \rho}{\partial x} &= \frac{x}{\rho}, \\ \frac{\partial \rho}{\partial y} &= \frac{y}{\rho}.\end{aligned}\tag{6.6}$$

Éstos cálculos se realizan con los argumentos de salida de la función `zernike_params`.

6. **Generar las coordenadas x_0 y y_0 que pasan a través de la rejilla.** Se requiere encontrar los valores de las coordenadas en las que el rayo de luz pasa a través de la rejilla. Para ello se utilizan las ecuaciones 6.1 y 6.2, donde se ingresan los resultados calculados a partir de los pasos anteriores.
7. **Calcular el valor de la irradiancia del rayo.** Asignar en un nuevo vector el valor de la irradiancia del rayo incidente por medio de la ecuación 6.4, para cada valor de las coordenadas.

El diagrama de flujo que detalla el proceso de simulación de un birronchigrama se encuentra en la Figura 6.3. Dentro del software la simulación se encuentra dentro de la función `evaluator`. Requiere como argumentos de entrada: los coeficientes propuestos, las coordenadas y los polinomios de Zernike con sus respectivas derivadas parciales. Los parámetros geométricos como la constante de conicidad, el centro de curvatura, también el número de líneas por pulgada de la rejilla de Ronchi utilizada, y la distancia a la cual esta es colocada con respecto al vértice del espejo son llamadas a partir del diccionario principal. Como retorno, la función otorga la irradiancia calculada por la ecuación 6.10. En la Figura 6.4a se muestra la simulación de un birronchigrama con constante de conicidad $K = 0$, y 50 líneas por pulgada. En la Figura 6.4b se muestra un birronchigrama simulado con constante de conicidad $K = 2$ y 50 líneas por pulgada. En la Figura 6.4c, el birronchigrama fue simulado con la constante de conicidad $K = 0.5$ y 50 líneas por pulgada, mientras que en la Figura 6.4d se utilizó la constante de conicidad $K = -0.75$ y 100 líneas por pulgada. En la Figura 6.4e, se utilizó la constante de conicidad $K = -1$ con 50 líneas por pulgada, y por último la Figura 6.4f muestra un birronchigrama simulado con constante de conicidad $K = -0.5$ y 50 líneas por pulgada.

6.3. Programación del método de normalización

En algunas técnicas de medición dentro del área de las pruebas ópticas, la normalización de la intensidad de la imagen o patrón de Ronchi es esencial para garantizar una interpretación precisa de la información que recibimos de una imagen. La variabilidad en las condiciones de iluminación puede distorsionar significativamente las imágenes, afectando la calidad y confiabilidad del análisis. El presente software aborda este problema programando el trabajo realizado por R. Juárez Salazar et. al. que detalla en su artículo [43]. Se trata de un método no iterativo directo para la interferometría de cambio de fase generalizada. Esta técnica está diseñada para recuperar la información de fase envuelta de un conjunto de interferogramas que tienen cambios de fase desconocidos entre

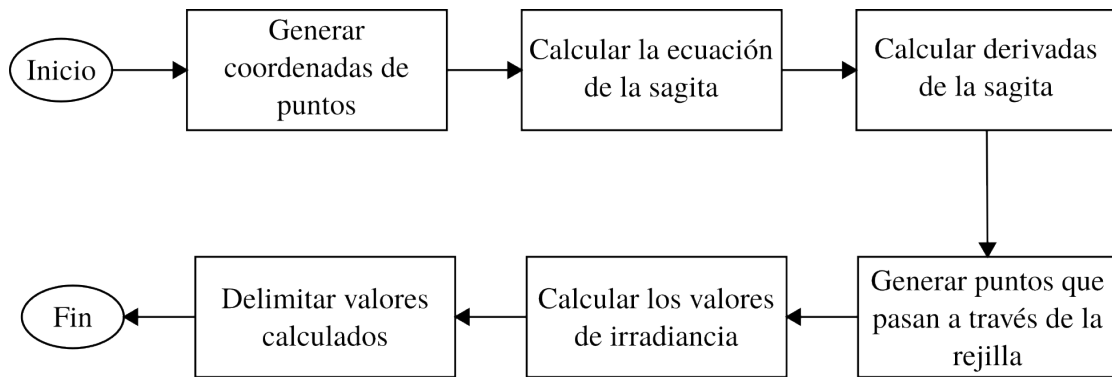


Figura 6.3: Diagrama de flujo que muestra el proceso de la simulación de un ronchigrama o un birronchigrama.

0 y π radianes. El enfoque propuesto utiliza el método de mínimos cuadrados para calcular cuatro parámetros esenciales: la luz de fondo y modulación, cambios de fase y la distribución de fase envuelta. El algoritmo introduce un procedimiento de normalización para los interferogramas, lo que le permite manejar casos con visibilidad espacio-temporal variable, superando las limitaciones asociados con los métodos convencionales para manejar la visibilidad espacial variable.

El método del presente trabajo combina algunas propiedades de procesamiento de imágenes para que la información obtenida de la imagen pueda ser analizada aplicando el procedimiento que la del artículo con respecto a la normalización de la intensidad en los patrones de Ronchi. Los componentes correspondientes son los siguientes:

1. **Procesamiento de la imagen.** Se requiere la biblioteca PIL, que cargará la imagen del birronchigrama capturado de la prueba de Ronchi experimental a un objeto *Image* y luego se realiza la conversión de la imagen a escala de grises. Una vez terminada la lectura, los datos de intensidad se guardan en una matriz con NumPy. Muy importante: la imagen birronchigrama o ronchigrama que deseamos ingresar deberá ser cuadrada, es decir debe tener la misma cantidad de píxeles tanto de largo como de ancho.
2. **Delimitar las coordenadas que se utilizarán.** Los polinomios de Zernike operan bajo una pupila circular unitaria [17], por lo que es necesario delimitar todos los puntos en un radio menor o igual a 1 al normalizar las coordenadas que se encuentran dentro del radio y centrar los datos útiles de las coordenadas seleccionadas.
3. **Representar los valores de irradiancia con los polinomios de Zernike.** En su artículo, el autor utiliza la expansión de Taylor para calcular la irradiancia como un polinomio de segundo grado y otro polinomio de cuarto grado. En este caso se utilizaron los polinomios de Zernike de orden 14 y de orden 28 respectivamente. Por lo que es necesario escribir cada uno de los valores de estos polinomios en coordenadas cartesianas. La tabla 6.1 muestra los 28 primeros polinomios de Zernike que se utilizan para la normalización. Se definen entonces dos matrices de valores: z_i que representan los primeros 14 elementos de los polinomios de Zernike, y q_i , un arreglo representando los 28 elementos.
4. **Resolver los dos sistemas planteados.** Los arreglos considerados en el punto anterior de-

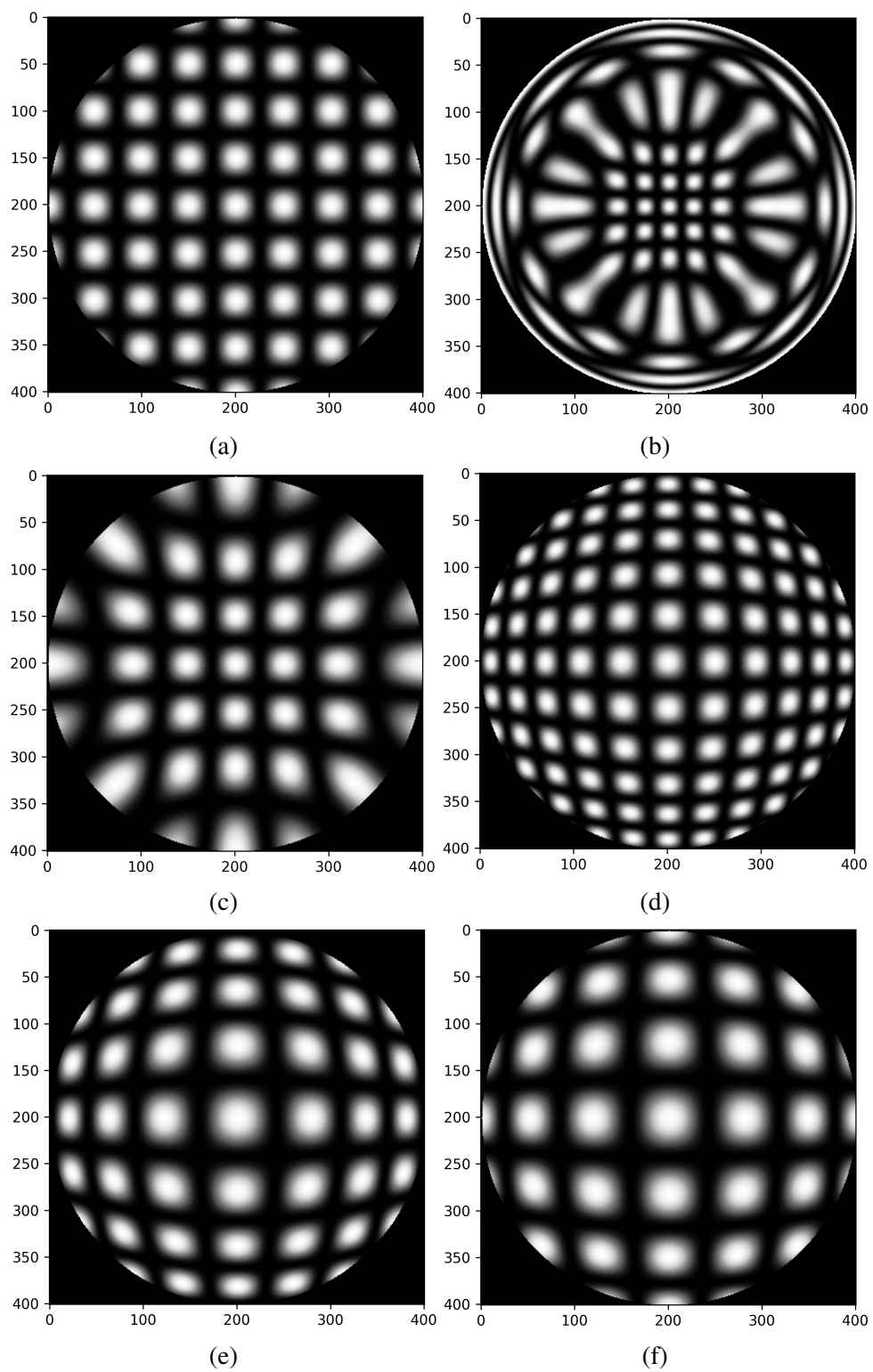


Figura 6.4: Esquema de imágenes que muestran la simulación de birnchigramas con diferentes parámetros.

Tabla 6.1: Tabla que muestra los 28 primeros polinomios de zernike, utilizados en el proceso de normalización.

Índice	Polinomio de Zernike
1	1
2	2x
3	2y
4	$\sqrt{3}(2\rho^2 - 1)$
5	$2\sqrt{6}xy$
6	$\sqrt{6}(x^2 - y^2)$
7	$\sqrt{8}y(3\rho^2 - 2)$
8	$\sqrt{8}x(3\rho^2 - 2)$
9	$\sqrt{8}y(3x^2 - y^2)$
10	$\sqrt{8}x(x^2 - 3y^2)$
11	$\sqrt{5}(6\rho^4 - 6\rho^2 + 1)$
12	$\sqrt{10}(x^2 - y^2)(4\rho^2 - 3)$
13	$2\sqrt{10}xy(4\rho^2 - 3)$
14	$\sqrt{10}(\rho^4 - 8x^2y^2)$
15	$4\sqrt{10}xy(x^2 - y^2)$
16	$\sqrt{12}x(10\rho^4 - 12\rho^2 + 3)$
17	$\sqrt{12}y(10\rho^4 - 12\rho^2 + 3)$
18	$\sqrt{12}x(x^2 - 3y^2)(5\rho^2 - 4)$
19	$\sqrt{12}y(3x^2 - y^2)(5\rho^2 - 4)$
20	$\sqrt{12}x(16x^4 - 20x^2\rho^2 + 5\rho^4)$
21	$\sqrt{12}y(16y^4 - 20y^2\rho^2 + 5\rho^4)$
22	$\sqrt{7}(20\rho^6 - 30\rho^4 + 12\rho^2 - 1)$
23	$2\sqrt{14}xy(15\rho^4 - 20\rho^2 + 6)$
24	$\sqrt{14}(x^2 - y^2)(15\rho^4 - 20\rho^2 + 6)$
25	$4\sqrt{14}xy(x^2 - y^2)(6\rho^2 - 5)$
26	$\sqrt{14}(8x^4 - 8x^2\rho^2 + \rho^4)(6\rho^2 - 5)$
27	$\sqrt{14}xy(32x^4 - 32x^2\rho^2 + 6\rho^4)$
28	$\sqrt{14}(32x^6 - 48x^4\rho^2 + 18x^2\rho^4 - \rho^6)$

scriben los sistemas definidos como:

$$a \approx \sum_{i=0} C_{uv} z_i \quad (6.7)$$

$$\frac{1}{2} b^2 \approx \sum_i D_{uv} q_i \quad (6.8)$$

donde a y b son elementos que se utilizarán para la normalización. C_{uv} y D_{uv} representan los coeficientes por hallar.

- a) **Resolver para a.** Se utiliza el método de mínimos cuadrados incluido dentro de la biblioteca SciPy para encontrar los coeficientes C_{uv} de a en la ecuación 6.7, donde el

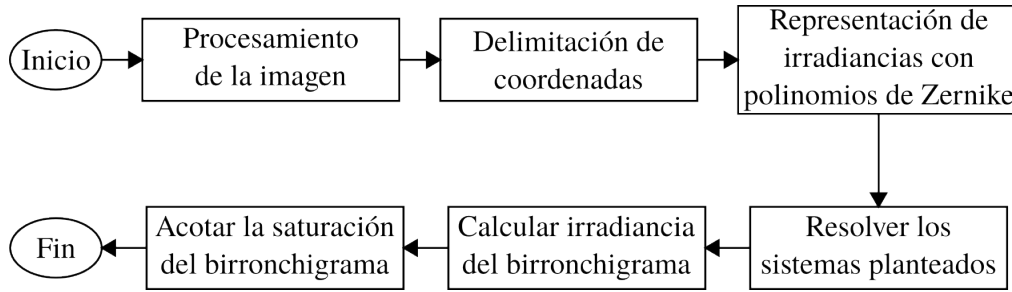


Figura 6.5: Diagrama de flujo que muestra el proceso de normalización de la intensidad de la imagen del patrón del ronchigrama o birronchigrama.

arreglo z_i y los valores de irradiancia dentro de la pupila unitaria del birronchigrama fueron usados con `lstsq` como argumentos de entrada.

b) **Resolver para b.** Se requirió representar el polinomio q_i de la siguiente manera:

$$q_i = [I - a]^2 \quad (6.9)$$

donde I es el vector de valores de irradiancia obtenidos del birronchigrama experimental, y a es el valor obtenido anteriormente. Después se aplicó el método de mínimos cuadrados de la biblioteca `lstsq` de SciPy, junto con q_i para encontrar los coeficientes D_{uv} y así encontrar b de acuerdo con la ecuación 6.8

5. **Calcular la irradiancia normalizada del birronchigrama.** Se utiliza el valor calculado de las ecuaciones 6.7 y 6.8, además de la intensidad I del birronchigrama para resolver la siguiente ecuación:

$$\bar{I} = [I - a] / b \quad (6.10)$$

donde \bar{I} es la intensidad del birronchigrama normalizado.

6. **Acotar la saturación del birronchigrama.** La ecuación 6.10 no garantiza que los valores de la irradiancia estén dentro de los parámetros debido al ruido presente dentro de los birronchigramas, por eso se define la función de saturación:

$$sat(\bar{I}) = \begin{cases} 1 & \text{si } \bar{I} > 1, \\ \bar{I} & \text{si } |\bar{I}| \leq 1, \\ -1 & \text{si } \bar{I} < -1. \end{cases} \quad (6.11)$$

El diagrama de flujo del proceso completo de normalización se muestra en la Figura 6.5.

La función de normalización está incluido en el programa con el nombre `normalizador`, que requiere como argumento de entrada la matriz de datos de la imagen y retorna un arreglo unidimensional que son los valores de irradiancia en cada coordenada de la imagen.

El usuario únicamente se encarga de describir la localización de la imagen a normalizar dentro del diccionario principal. Los parámetros necesarios para su funcionamiento son obtenidos directamente de las características de la imagen, como número de píxeles y dimensiones.

Se ha aplicado el software en entornos con variaciones de iluminación, tanto de ronchigramas como de birronchigramas. Como producto de pruebas de ejecución y funcionamiento, se presenta el resultado de normalizar algunos de ellos en el conjunto de figuras 6.6. En el caso de los birronchigramas experimentales de las imágenes 6.6a, 6.6c, 6.6e, especialmente el último, el programa de normalización logra mejorar de manera significativa la calidad del birronchigrama, como se aprecian en las figuras 6.6b, 6.6d y 6.6f, respectivamente. El cambio de intensidades se puede notar aún cuando la intensidad es tan baja como en la Figura 6.6e.

El presente software ha demostrado ser relativamente rápido en un procesador Intel Pentium G4400 a 3.3 GHz con 12 GB de RAM, consumiendo un tiempo de ejecución promedio de 2.21 segundos utilizando imágenes de hasta 550×550 píxeles, y comportándose de manera similar tanto en formato de entrada PNG o BMP.

6.4. Programación del método de correlación de imágenes (Birronchigramas)

El uso del método de correlación es la parte esencial del programa, pues su principal función es hacer la comparación entre el birronchigrama simulado y el experimental. Ésto es lo que se conoce como la función de aptitud. El programa está basado en la continuación del método descrito por A. Cordero Dávila y J. González García [2] utilizado en la programación del método de simulación de birronchigramas. El proceso implica recibir una propuesta de los valores de los coeficientes de Zernike, y usar estos datos para simular un birronchigrama nuevo, para luego correlacionarlo con los datos experimentales, una vez normalizado el valor de irradiancia. La metodología utilizada se describe a continuación:

1. **Evaluar los datos simulados.** Se requiere como argumento de entrada los valores para simular el birronchigrama a correlacionar, utilizando la función y los argumentos de entrada de la función `evaluator`.
2. **Calcular el coeficiente de correlación entre la irradiancia obtenida y la irradiancia del birronchigrama experimental.** La correlación de la simulación empieza al comparar el grupo de triadas (x_{Ei}, y_{Ei}, I_{Si}) y (x_{Ei}, y_{Ei}, I_{Ei}) . La función de correlación mide qué tan bien coinciden estas triadas. Se define como:

$$C = \frac{\sum_{i=1}^N [(I_{Ei} - \bar{I}_E)(I_{Si} - \bar{I}_S)]}{[\sum_{i=1}^N (I_{Ei} - \bar{I}_E)^2 \sum_{i=1}^N (I_{Si} - \bar{I}_S)^2]^{1/2}} \quad (6.12)$$

donde \bar{I}_S (irradiancia simulada) y \bar{I}_E (irradiancia experimental) representan los valores promedio. \bar{I}_{Si} y I_{Ei} representa cada uno de los elementos en las triadas de valores, respectivamente.

Este método se implementa en el algoritmo de optimización, llamado `fitness_function`. El argumento de salida de esta función es el coeficiente de correlación $C \in [0, 1]$. Para realizar las pruebas de esta función, se utiliza valores de los coeficientes de Zernike de un mismo birronchigrama con coeficientes de correlación obtenidos con el algoritmo genético utilizado en el trabajo de correlación de birronchigramas [2]. La tabla 6.2 muestran algunas pruebas evaluando tres grupos con los primeros 14 polinomios de Zernike, los cuales otorgan un coeficiente de correlación al final de ésta.

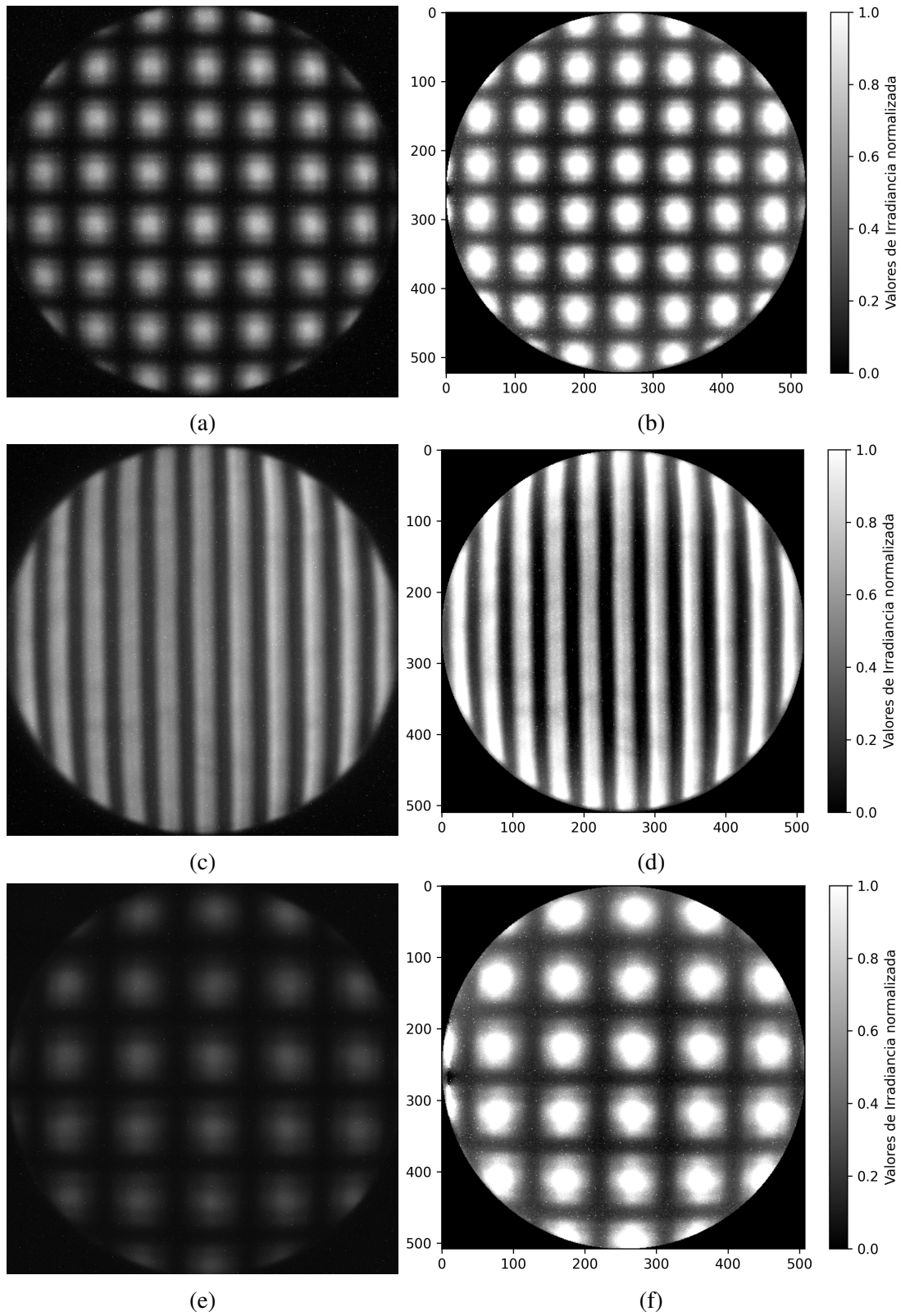


Figura 6.6: Esquema de imágenes que muestran la imagen del birronchigrama o ronchigrama experimental (lado izquierdo), y sus respectivas normalizaciones (lado derecho).

Tabla 6.2: Tabla que muestra 3 grupos de los coeficientes de Zernike que fueron evaluados con la función de aptitud, y se comparan con los coeficientes de correlación C obtenidos con el algoritmo genético utilizado en [2].

Parámetro	Coeficientes		
Tilt y	17.8115	19.4366	18.0679
Tilt x	-11.5951	0.1511	-9.0820
Defoco	1655.8838	1660.4919	1660.2478
Astigmatismo primario 45°	36.6867	34.9396	33.1940
Astigmatismo primario 0°	-1.8753	0.0885	-1.4664
Coma primaria y	1.9974	-42.8635	-31.1356
Coma primaria x	0.0443	-1.99	-1.8890
Az 8	14.4394	7.7789	23.6480
Az 9	5.2307	1.3000	-6.3538
Esférica	0.01	-3.1738	-3.4790
Az11	-0.2090	-3.5614	-0.5691
Az12	-2.2485	-0.799	1.5588
Az13	0.1935	0.0079	-2.9791
Az14	1.2414	5.8313	1.3263
C (Algoritmo genético)	0.928889394	0.929760754	0.929760754
C (Función de aptitud)	0.928292320	0.924560675	0.930492757

6.5. Programación del método de optimización

6.5.1. Algoritmo Genético

Se utilizó un algoritmo genético para valores reales, el cual se basa en la Figura 5.8, en donde se probaron 3 métodos de selección: por torneo, por ruleta, y selección natural. Para el resto del programa, se utilizó el desarrollo definido por R. L. y S. E. Haupt [28]:

1. **Definir parámetros iniciales para la ejecución del algoritmo.** El algoritmo genético requiere de tres parámetros para su funcionamiento: número de generaciones, probabilidad de mutación y el valor objetivo del algoritmo. Todos estos valores se encuentran dentro del diccionario `parametros`.
2. **Generar población inicial.** Usando `random.uniform` de NumPy se genera aleatoriamente una población con tamaño definido y acotado por los límites de cada variable.
3. **Iniciar el ciclo.** La condición de terminación es una bandera llamada `keep`, que tiene el booleano `True` hasta que se cumpla la condición definida:
 - a) **Evaluación de los individuos.** Se evalúan los individuos por medio de la función de aptitud o de costo, en el algoritmo la función se llama `cost_function`
 - b) **Seleccionar la población.** El método utilizado es una implementación del método por torneo que requiere una función adicional de búsqueda, llamado `indexor`, la cual

entrega la población encontrada de acuerdo a la posición en el array procesado por la función `tournament_selection`.

- c) **Cruza de la población.** Se cuenta con una función de cruce llamada `crossover`, que utiliza a su vez dos funciones: `blxa` que es un el método de cruce de cromosomas definido en el capítulo anterior y también la función `concatenate` de NumPy.
- d) **Agrupar a nuevos individuos con padres.** Para ello se usa `concatenate` de NumPy para unir los padres que fueron cruzados y los hijos obtenidos.
- e) **Mutar la nueva población generada.** Se utiliza la función `mutator`, que además de requerir la población inicial utiliza un valor numérico de 0 a 1 representando la probabilidad de mutación.

El algoritmo termina cuando se cumple el criterio de terminación, y la bandera `keep` cambia de `True` a `False`. El mismo diagrama de flujo detallado en la Figura 5.8 puede aplicarse para representar los pasos anteriores.

6.5.2. Estrategia Evolutiva (1+1)

La estrategia evolutiva (1+1) es relativamente sencilla de programar. Al requerir únicamente cálculos computacionales sencillos y además del uso de pocas funciones, puede procesar un mayor número iteraciones en un menor tiempo. La función `oneplusone` contiene esta estrategia.

1. **Definir parámetros iniciales.** Se definen inicialmente el parámetro endógeno c de la estrategia (que en este caso se usó el definido por el autor [37]) y el tamaño de paso inicial `sigma`. También se inicializan los contadores de variables y el número de iteraciones.
2. **Evaluar al individuo padre.** Se evalúa la aptitud del individuo con la función de aptitud.
3. **Iniciar ciclo:** Se define una variable llamada `keep` que servirá como bandera para detener el ciclo. Se mantendrá inicialmente con el booleano `True`.
 - a) **Crear descendencia.** Se le suma al individuo padre una cantidad dada por el tamaño de paso actual multiplicado por un valor aleatorio generado por la librería `random.normal` de NumPy, tal como lo indica la ecuación 5.29.
 - b) **Evaluar descendencia.** Se evalúa al individuo descendiente con la función de aptitud.
 - c) **Comparar la evaluación del padre con la descendencia.** Se elegirá al individuo como el nuevo padre que cumpla la condición de comparación (dependiendo si es un proceso de maximización o minimización), además de guardar en un contador, el número de éxitos o fracasos que se hayan dado al realizar esta comparación.
 - d) **Cambiar el tamaño de paso.** El cambio de paso se realiza como se describe en la sección 5.4.
 - e) **Verificar condición de terminación.** Si la condición de terminación es igual al fijado en los parámetros iniciales, el valor de la bandera pasará de `True` a `False`.

El diagrama de flujo que describe la estrategia se detalla en la Figura 6.7. La función que engloba a la estrategia se llama `oneplusone` y requiere como argumentos de entrada un individuo inicial y el número máximo de evaluaciones de la estrategia.

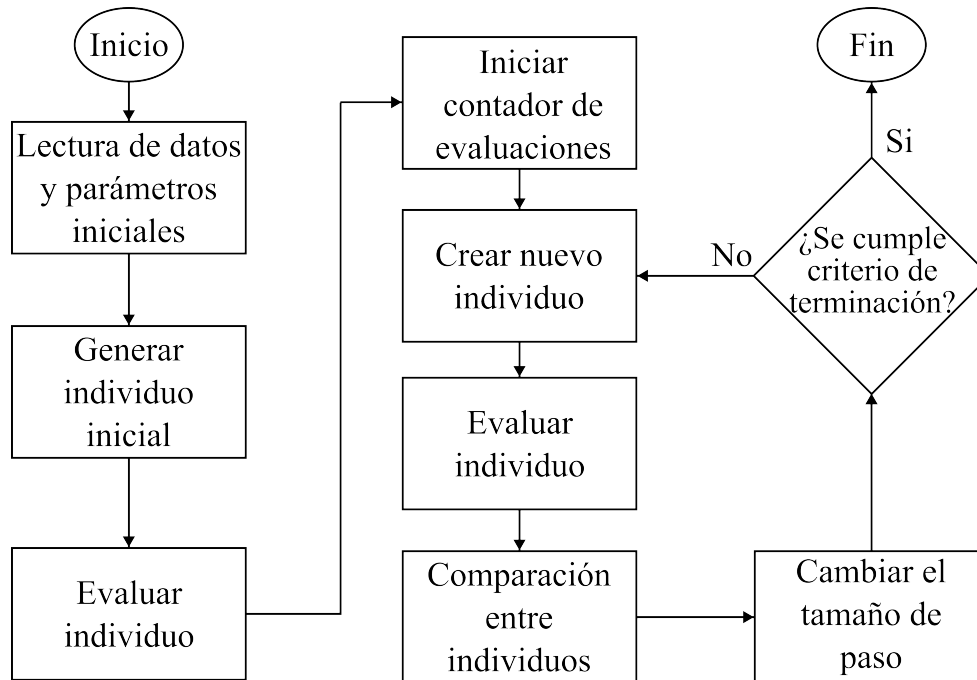


Figura 6.7: Diagrama de flujo que describe el algoritmo de la estrategia evolutiva (1+1).

6.5.3. Estrategia (1+1)-Cholesky-CMA

Se programó la estrategia evolutiva (1+1)-Cholesky-CMA basado en el algoritmo 2, y utilizando los parámetros iniciales definidos por el autor [37]:

1. **Lectura de datos y parámetros iniciales.** Se utilizaron los parámetros endógenos definidos originalmente por Thomas Bäck et. al. [37].
2. **Evaluar individuo inicial.** Se evalúa el individuo padre con la función de aptitud previamente establecida.
3. **Repetir:**
 - a) **Crear descendencia.** Se crea la descendencia por medio de la mutación del padre como se muestra la sección 5.4.6.
 - b) **Comparación entre aptitudes.** Se comparan las evaluaciones del padre y del hijo. Dependiendo del caso, se selecciona el mejor individuo. Si la descendencia mejora, entonces $\lambda_s = 1$, de lo contrario, $\lambda_s = 0$.
 - c) **Modificar tasa de éxitos acumulado y el tamaño de paso.** Este paso se realiza con ayuda de las ecuaciones 5.34 y 5.35.
 - d) **Comparación entre aptitudes.** Si la aptitud evaluada del hijo es mejor que la del padre, entonces la descendencia toma el lugar del padre.
 - 1) **Comparación entre valor umbral y la tasa de éxitos.** Si en el caso de que haya nueva descendencia, se compara el valor de la tasa de éxitos \bar{p}_s que sea menor o

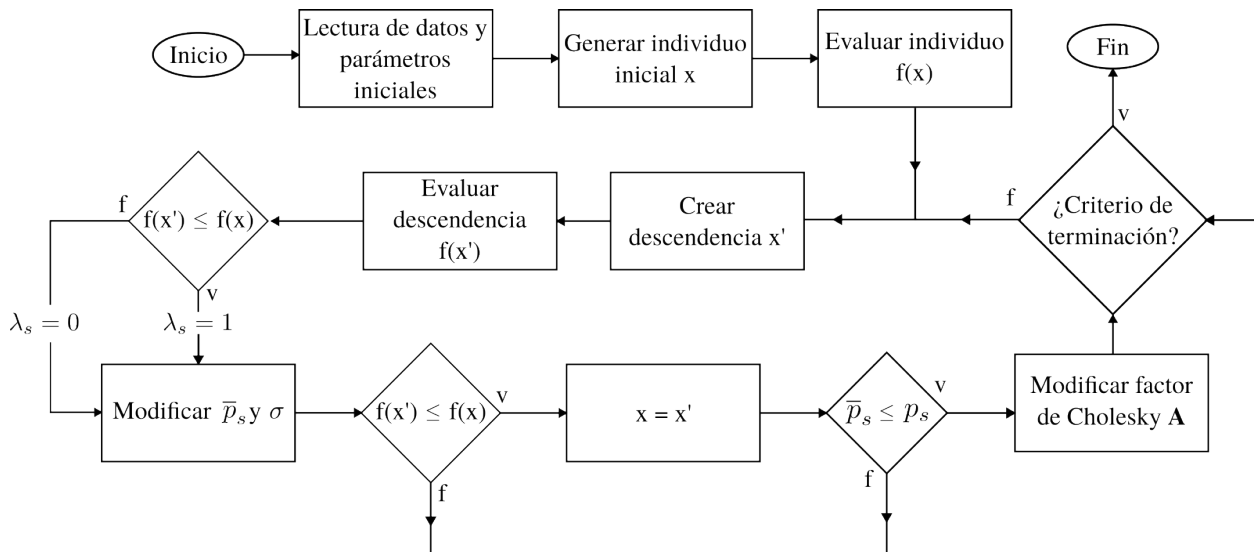


Figura 6.8: Diagrama de flujo que representa la estrategia evolutiva (1+1)-Cholesky-CMA (caso de minimización).

igual al valor de umbral p_s . Si esto es cierto se modifica la matriz factor \mathbf{A} como lo describe la ecuación 5.33.

4. **Terminar** hasta que se cumpla el criterio de terminación.

El diagrama de flujo que representa la programación de la estrategia evolutiva (1+1)-Cholesky-CMA se presenta en la Figura 6.8. Este diagrama de flujo se simula un proceso de minimización. La función `cholesky_ES` requiere un individuo aleatorio como argumento de entrada y que funciona como el individuo inicial. Esta función retorna el individuo óptimo encontrado y su valor de aptitud.

6.5.4. Algoritmo de optimización híbrido

El método de optimización empleado es un algoritmo híbrido que está formado por la estrategia evolutiva (1+1)-Cholesky-CMA y el algoritmo genético de valores reales. Una descripción de la programación de este algoritmo se presenta a continuación:

1. **Definir parámetros de ejecución del algoritmo híbrido.** Los valores necesarios son el número de población, el número de generaciones, rangos del espacio de búsqueda, probabilidad de mutación, número de evaluaciones de la estrategia, rango de búsqueda y el valor objetivo del algoritmo. Estos valores se extraen dentro del diccionario `parametros`.
2. **Generar individuo inicial** Se genera un individuo aleatoriamente dentro de los rangos definidos y con la librería de NumPy `random.uniform`.
3. **Realizar la búsqueda del individuo óptimo por medio de la estrategia evolutiva.** Se utiliza la estrategia evolutiva (1+1)-Cholesky-CMA para realizar la búsqueda del individuo óptimo.

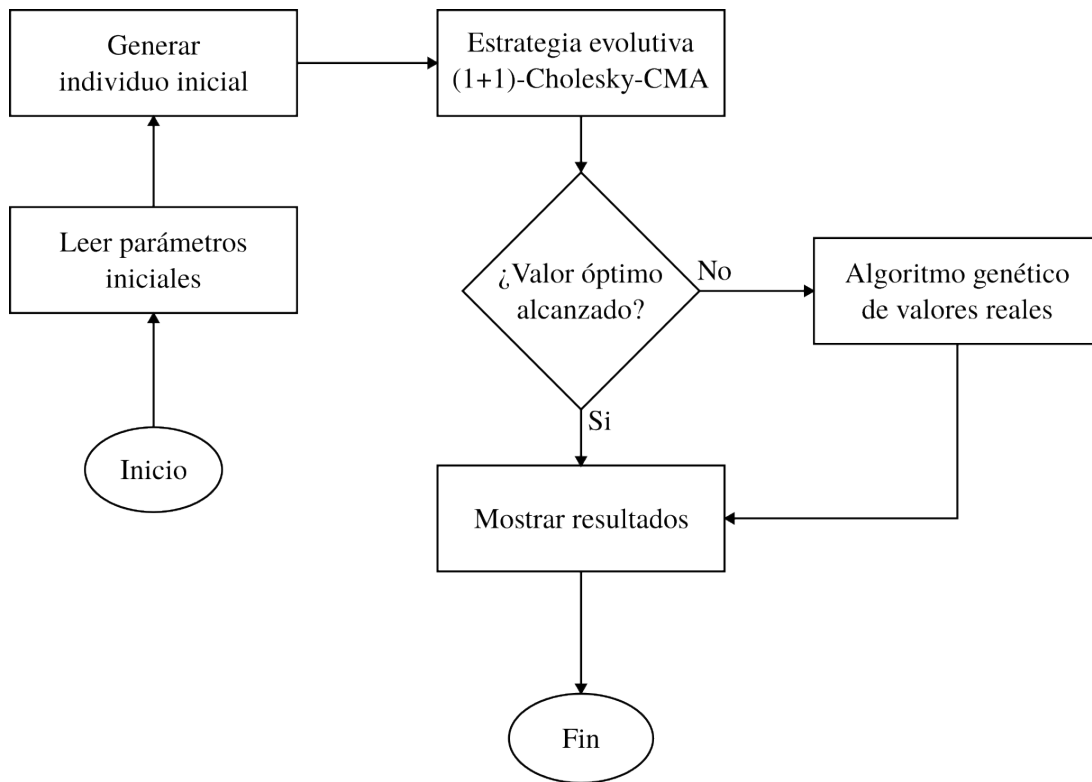


Figura 6.9: Diagrama de flujo de la programación del algoritmo híbrido.

4. **Verificar resultados.** Se muestra en pantalla el resultado obtenido. Si la estrategia evolutiva logra encontrar el valor óptimo deseado, termina el programa, de lo contrario, aparecerá un mensaje en donde indica que la búsqueda se realizará con el mejor individuo encontrado por la estrategia, generando una población inicial dentro del porcentaje del rango de establecido. Esta población ingresará al algoritmo genético de valores reales.
5. **Mostrar resultados.** Se muestran los resultados obtenidos, ya sea que por medio de la estrategia evolutiva se encontró el valor óptimo o bien por el algoritmo genético si éste encontró una mejor aproximación.

El esquema del diagrama de flujo del algoritmo híbrido se encuentra en la Figura 6.9. `Hybrid_2` es la función dentro del software que ejecuta el algoritmo híbrido. Los argumentos de entrada para esta función son el diccionario `parametros` y también de los límites iniciales de búsqueda. La rutina entrega tanto los coeficientes hallados como su valor de aptitud.

6.6. Programa integral de evaluación de espejos

El programa general une todas las funciones en un solo programa. Después de que se haya logrado obtener una correlación óptima, ésta se mostrará en pantalla con los valores de cada uno de los coeficientes de Zernike, además de una imagen que resume el proceso del software: el birronchigrama original, el birronchigrama normalizado y por último el birronchigrama simulado mediante

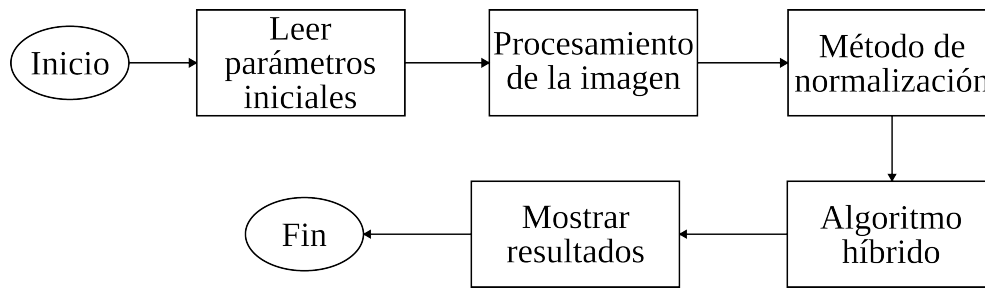


Figura 6.10: Diagrama de flujo del software de evaluación de espejos.

los coeficientes óptimos calculados. La descripción del software integral de evaluación de espejos se describe a continuación:

1. **Leer parámetros iniciales.** Esto implica tanto las propiedades del espejo como los parámetros del algoritmo híbrido. Todos estos datos se colocan dentro del diccionario `parametros`.
2. **Procesamiento de la imagen.** Se hace el procesamiento de la imagen a escala de grises y la conversión de la imagen con la biblioteca de Pillow a una matriz de datos con NumPy.
3. **Método de normalización.** Se procede a realizar el método de normalización en los datos del birronchigrama procesado anteriormente para normalizar la irradiancia de 0 a 1 dentro de una pupila de radio unitario.
4. **Algoritmo híbrido.** Se implementa la estrategia evolutiva (1+1)-Cholesky-CMA con el algoritmo genético de valores reales para la búsqueda de un valor de correlación óptimo.
5. **Mostrar resultados.** Se muestran en pantalla el valor de correlación obtenido y los coeficientes de aberración. También se muestra una ventana con los tres birronchigramas, el original, el normalizado y el simulado.

El diagrama de flujo del software general se aprecia en la Figura 6.10. El archivo del programa se nombrará `software_general_espejos.py` y se requerirá abrir en un editor de texto para modificar la localización del birronchigrama, así como los parámetros del espejo y de funcionamiento del método de optimización.

Capítulo 7

Pruebas de eficiencia

7.1. Introducción

Como se ha dado a conocer en el capítulo anterior, la programación del software general incluye una rutina con un algoritmo híbrido para encontrar coeficientes de polinomios de un número relativamente alto. Este programa es el que más recursos y tiempo ocupa dentro del software, por lo que se tendrá que escoger la combinación de algunos de los modelos disponibles para que el tiempo de ejecución y de recursos sea óptimo. Para ello se consideran dos pruebas para demostrar la eficiencia del programa:

- **Caso 1.** Prueba de evaluación de un molde para lentes flexibles, utilizando la técnica de pantallas nulas. La técnica consiste en proyectar un haz de luz sobre la superficie y medir la ubicación del haz reflejado mediante la cámara CCD. Las desviaciones en la posición del haz reflejado respecto al ideal implican errores en el perfil de la superficie [44]. El polinomio para representar la superficie está descrito como:

$$A + Bx^3 + Cx^2 + Dx + Ex^2y + Fxy + Gxy^2 + Hy + Iy^2 + Jy^3, \quad (7.1)$$

donde $A, B, C, D, E, F, G, H, I$ y J son los coeficientes de un polinomio de grado 3. En el mismo artículo, muestra los resultados de los coeficientes que los autores estimaron por medio del método de mínimos cuadrados. En este caso, para estimar el error en la superficie por medio del algoritmo híbrido, se utiliza la suma de errores al cuadrado como función de aptitud, que se define como:

$$S = \sum_i^{np} [\omega_s(i) - \omega_e(i)]^2, \quad (7.2)$$

y las variables a optimizar serán los coeficientes del polinomio planteado. De la ecuación 7.2 se define $\omega_s(i)$ como la evaluación de la ec. 7.1 de cada punto (x_i, y_i) en la superficie, mientras que $\omega_e(i)$ son los puntos en la superficie obtenidos de manera experimental por el método de pantallas nulas. El valor np es el número de puntos totales medidos de la superficie del molde.

Evaluando el polinomio completo con los coeficientes mostrados en la Figura 7.1, el error estimado otorga la cantidad de 2.9954, lo que representará un referente al minimizar el error con la función 7.2.

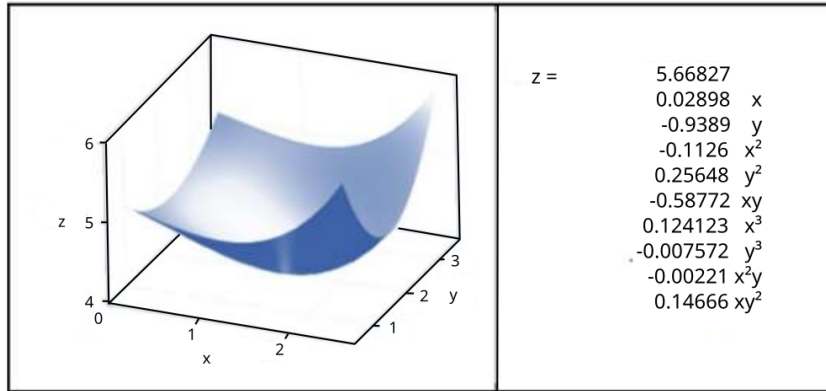


Figura 7.1: Superficie generada junto con los coeficientes y las variables del lado derecho [44].

- Caso 2.** La segunda prueba es el propósito general del software, que consiste en la maximización de la correlación entre un birronchigrama obtenido de un banco de 50 espejos [9] con los coeficientes generados de forma óptima por el algoritmo híbrido. El espacio de búsqueda está definido por los límites mostrados en la tabla 7.1.

Tabla 7.1: Valores límite para el espacio de búsqueda en la prueba de maximización.

Límite	Valores												
Límite inferior	-150	-150	500	-50	-50	-50	-50	-200	-100	-50	-20	-20	-20
Límite superior	150	150	2500	50	50	50	50	200	100	50	20	20	20

Se requiere saber cuál es la mejor combinación entre variables, tanto por el número de individuos, número de generaciones, la tasa de mutación y el número de evaluaciones para la estrategia evolutiva. Todas las pruebas fueron realizadas en una computadora con procesador Intel Pentium G4400, dual-core a 3.30 GHz con 12 GB de RAM, utilizando Fedora 38 como sistema operativo.

7.2. Proceso de eficiencia del algoritmo híbrido para el caso 1.

7.2.1. Método de selección

Para verificar la mejor configuración en el algoritmo genético de valores reales, se tuvo que elegir la mejor opción entre diferentes métodos de selección de individuos: el método de la ruleta, por rango y una implementación que combina el método de selección por torneo, con selección natural. Para ello se utiliza el algoritmo genético aplicado al problema de minimización, con 300 individuos y 50 generaciones, en un rango de búsqueda general para todos los cromosomas de $[-10, 10]$. Este algoritmo se ejecutó 10 veces para cada una de las versiones de selección a una tasa de mutación fija del 7%.

Se observa que tanto la selección por ruleta en la Figura 7.2 y por rango mostrado en la Figura 7.3, presentan una mayor diversidad de individuos, sin alguna convergencia aparente. Los valores mínimos obtenidos a partir de alguna de las ejecuciones de esta prueba, fueron 4.58×10^5 y 4.33×10^5 respectivamente, un valor alto en comparación a los obtenidos cuando se ejecutó el algoritmo

Prueba de convergencia en un AG utilizando el método de selección por ruleta.

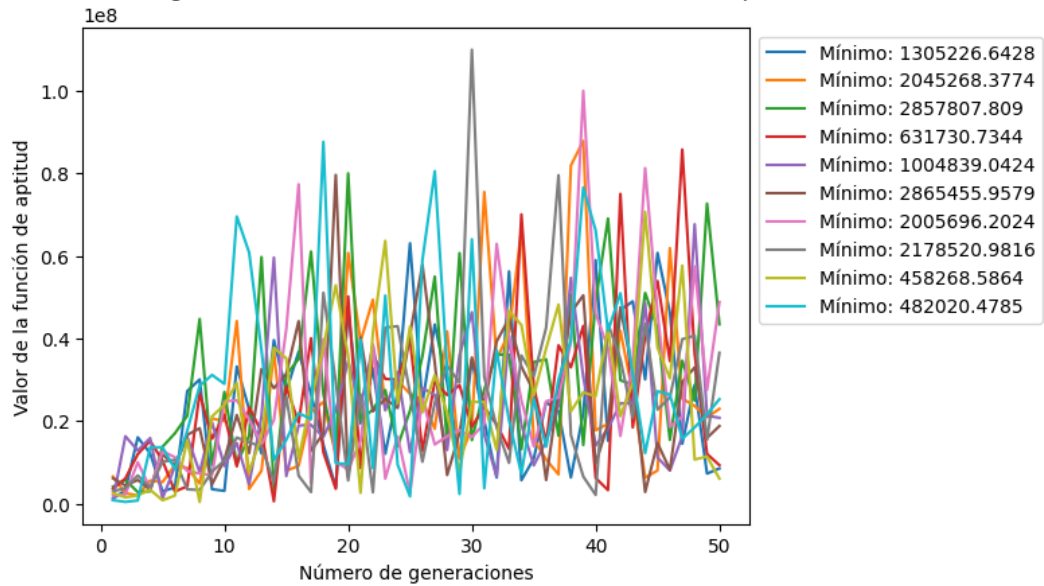


Figura 7.2: Gráfico que muestra el comportamiento en la ejecución del algoritmo genético de valores reales para 10 poblaciones con 300 individuos y 50 generaciones y tasa de mutación fija del 7%, por medio del método de selección por ruleta. También se presentan los valores mínimos obtenidos por cada población.

usando el método por torneo. Las pruebas con el método de selección por torneo se muestran en la Figura 7.4. Entre las ejecuciones realizadas se obtuvo un mínimo de $4,96 \times 10^3$. También se aprecia en la misma figura, que el método de selección implementado en las pruebas tienden a una convergencia uniforme.

7.2.2. Tasa de mutación variable

También se utilizó una tasa de mutación variable para comprobar si este factor influía en una mejor aproximación al mínimo deseado, y utilizando las mismas características utilizadas en las pruebas anteriores, con tasa de mutación inicial del 7% y un factor de aumento de 1.001 por generación, se presentan los gráficos 7.5, 7.6 y 7.7.

En estas pruebas se puede apreciar que la diversidad de la población aumenta, sin embargo, la convergencia hacia la solución del problema se ve afectada, ya que se observa que al ejecutar 10 veces el algoritmo con el método de selección por ruleta en la Figura 7.5, se observa que el mínimo valor del mejor individuo en una de las poblaciones llega a 2.02×10^5 , al igual que la selección por rango en la Figura 7.6, en donde el mejor individuo de una las poblaciones logra un valor de 8.19×10^5 . En el caso de la selección por torneo de la Figura 7.7, se observa una convergencia similar a la prueba con el mismo método anterior, pero no logra ser mejor que éste, ya que se aprecia un aumento del valor mínimo obtenido. Es decir, que el mejor valor para esta prueba fue de 2.08×10^4 , lo que al comparar entre pruebas hace concluir que el método por torneo sin mutación variable es la mejor opción al ejecutar el algoritmo genético de valores reales.

Prueba de convergencia en un AG utilizando el método de selección por rango.

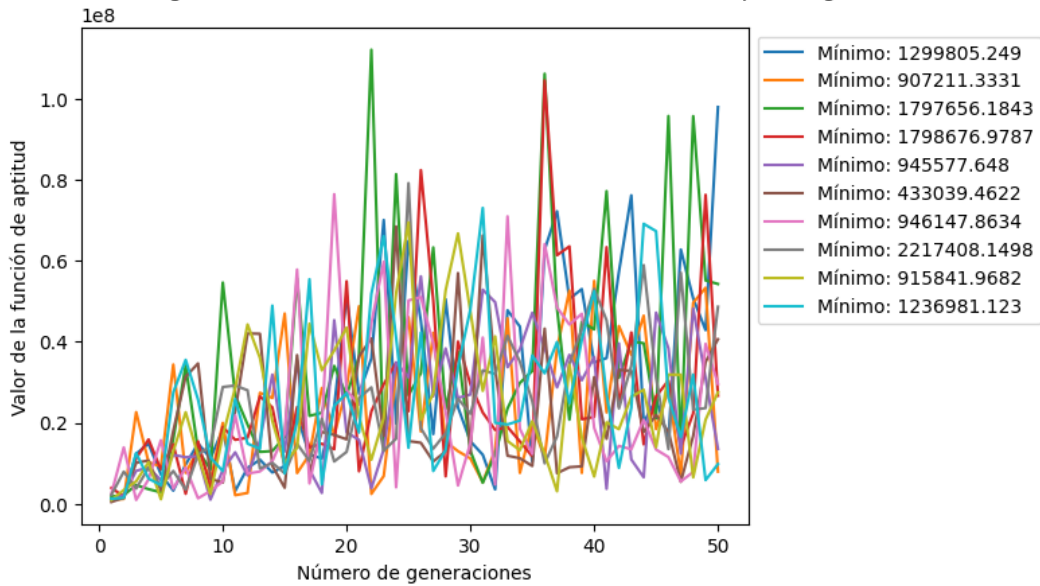


Figura 7.3: Gráfico que muestra el comportamiento en la ejecución del algoritmo genético de valores reales para 10 poblaciones con 300 individuos y 50 generaciones y tasa de mutación fija del 7%, por medio del método de selección por rango. También se presentan los valores mínimos obtenidos por cada población.

Prueba de convergencia en un AG utilizando el método de selección por torneo.

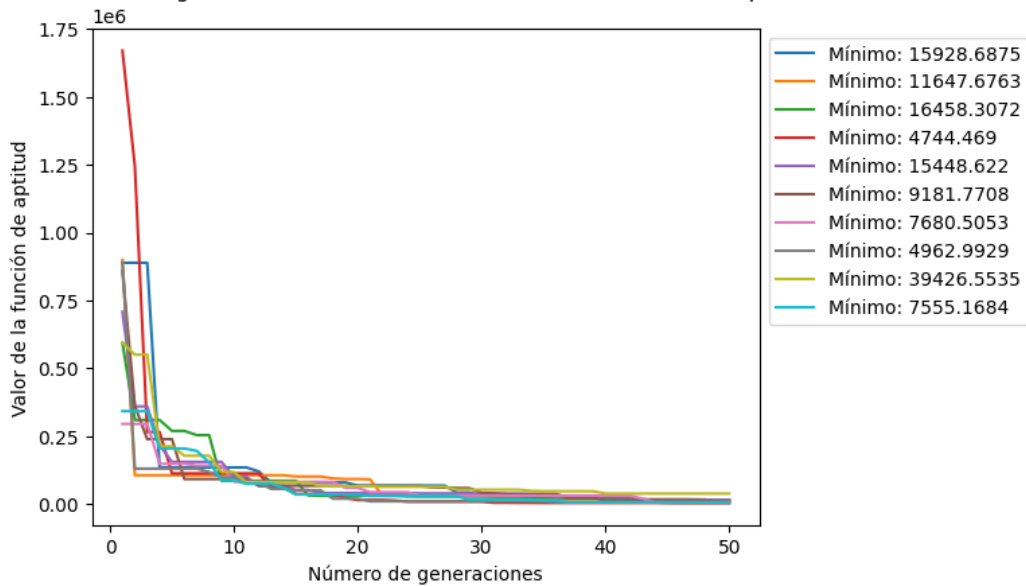


Figura 7.4: Gráfico que muestra el comportamiento en la ejecución del algoritmo genético de valores reales para 10 poblaciones con 300 individuos y 50 generaciones y tasa de mutación fija del 7%, por medio del método de selección por torneo. También se presentan los valores mínimos obtenidos por cada población.

Prueba de convergencia en un AG utilizando el método de selección por ruleta.

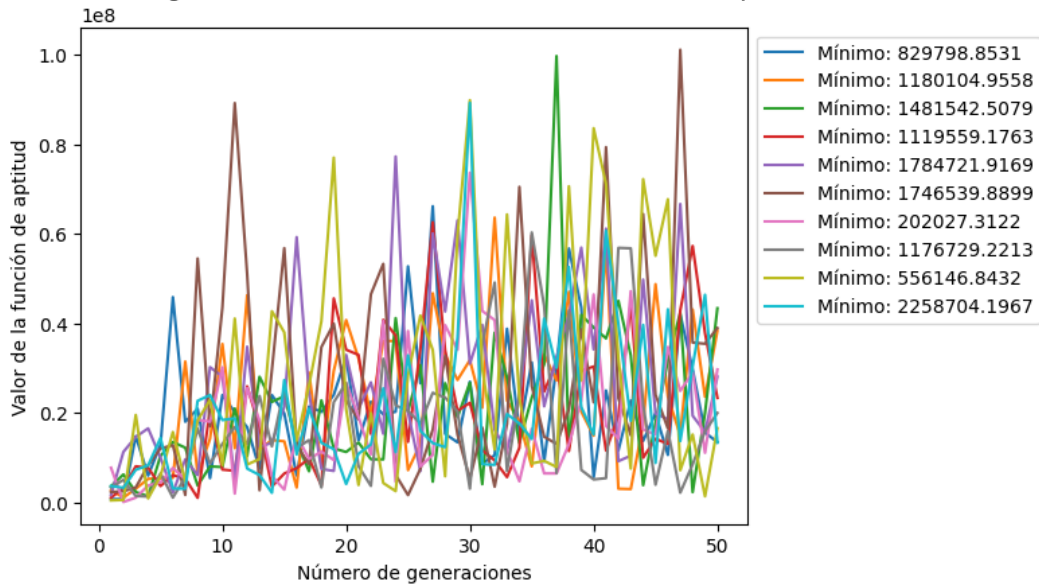


Figura 7.5: Gráfico que muestra el comportamiento de los mejores individuos de 10 poblaciones con 300 individuos, 50 generaciones y con tasa de mutación variable por el método de selección por ruleta con la tasa de mutación variable inicial del 7%.

Prueba de convergencia en un AG utilizando el método de selección por rango.

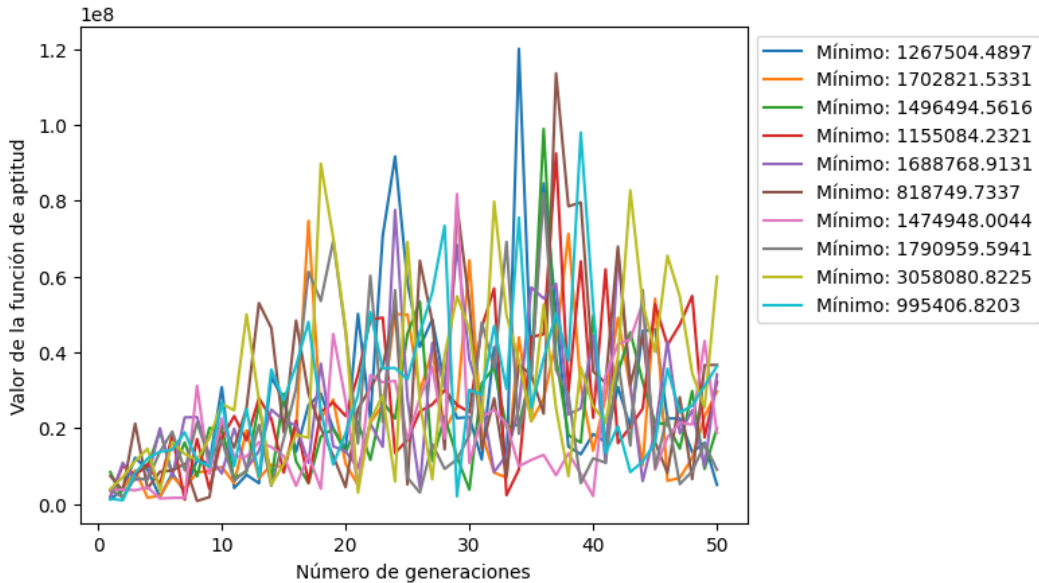


Figura 7.6: Gráfico que muestra el comportamiento de los mejores individuos de 10 poblaciones con con 300 individuos, 50 generaciones y una tasa de mutación variable por el método de selección por rango con la tasa de mutación variable inicial del 7%.

Prueba de convergencia en un AG utilizando el método de selección por torneo.

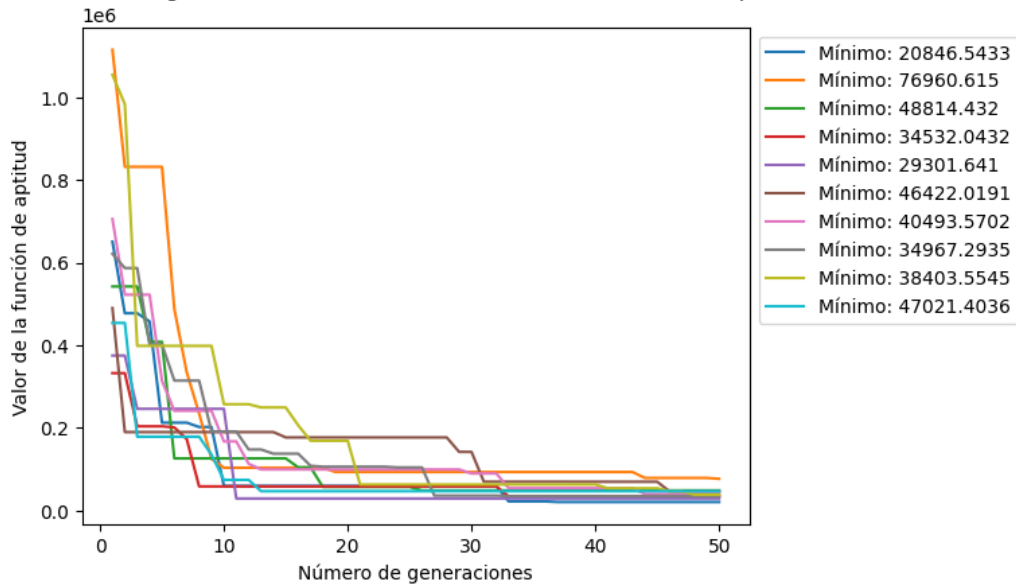


Figura 7.7: Gráfico que muestra el comportamiento de los mejores individuos de 10 poblaciones con 300 individuos, 50 generaciones y una tasa de mutación variable por el método de selección por torneo con la tasa de mutación variable inicial del 7%.

7.2.3. Número de individuos y número de generaciones óptimos

Para dar una visión completa del funcionamiento del algoritmo genético de valores reales, se realizaron 6 pruebas cambiando parámetros como los son el número de individuos, número de generaciones, y probabilidad de mutación. R. L. Haupt y S. E. Haupt [28] sugieren que la probabilidad de mutación para este algoritmo sea del 20%. Aún así se pusieron a prueba diferentes probabilidades para observar si ocurre un cambio en la evolución de los individuos por generación. En el caso de las pruebas mostradas tanto en la Figura 7.8a, en donde se configuró para 300 individuos con 100 generaciones, como en la Figura 7.8b, configurándose con 750 individuos y 500 generaciones, se utilizó el 5% de probabilidad de mutación. En el caso de las pruebas mostradas en la Figura 7.8c con 500 individuos y 200 generaciones, y en la Figura 7.8d con 1000 individuos y 750 generaciones, fueron realizadas con el 10% de probabilidad de mutación. Por otra parte las pruebas de la Figura 7.8e con 500 individuos y 300 generaciones, y las pruebas de la Figura 7.8f con 1000 individuos y 750 generaciones, se ejecutaron con el 20% de mutación.

Se aprecia que el comportamiento en las diferentes ejecuciones del algoritmo genético que la convergencia hacia el óptimo se intensifica en las primeras 100 evaluaciones. También entre más cercana sea la aproximación al objetivo, la población requiere más generaciones para llegar a él. Se aprecia en las figuras 7.8e y 7.8f que una mayor probabilidad de mutación conlleva a una mayor diversidad de individuos debido a que estos gráficos presentan mayores variaciones durante la convergencia, además que los mejores individuos no son tan óptimos como los de las figuras 7.8a y 7.8b.

Por otra parte, al ejecutar las pruebas de rendimiento, se midieron los tiempos de ejecución de cada rutina. Se estimó que en las pruebas de la Figura 7.8a, el tiempo promedio de ejecución fue de 16 segundos, que en las pruebas de la Figura 7.8b fue de 3.5 minutos, en las pruebas de la

Figura 7.8c el tiempo promedio fue de 56.71 segundos, en las pruebas de la Figura 7.8d el tiempo promedio fue de 8.6 minutos, en las pruebas de la Figura 7.8e el tiempo promedio fue de 1.41 minutos, y en las pruebas de la Figura 7.8f el tiempo promedio fue de 8.16 minutos.

En todas las pruebas realizadas con el algoritmo genético de valores reales, se observó que, aunque el proceso de búsqueda es eficiente, no logra alcanzar de manera satisfactoria el óptimo establecido en el trabajo de investigación [44]. Por lo tanto, se decidió explorar métodos más efectivos que pudieran cumplir con el objetivo, considerando las estrategias evolutivas como una alternativa viable.

7.2.4. Optimización de la estrategia evolutiva (1+1).

Para encontrar la configuración óptima de la estrategia evolutiva se efectuaron las pruebas con los mismos límites inferior y superior del algoritmo genético de valores reales. Las variables que se pusieron a prueba fueron el tamaño de paso y el parámetro c , usando la condición que $0.817 \leq c \leq 1$. También el programa se configuró para detenerse cuando llegue a 3, siendo un valor óptimo cercano al publicado por A. Santiago Alvarado et. al. [44].

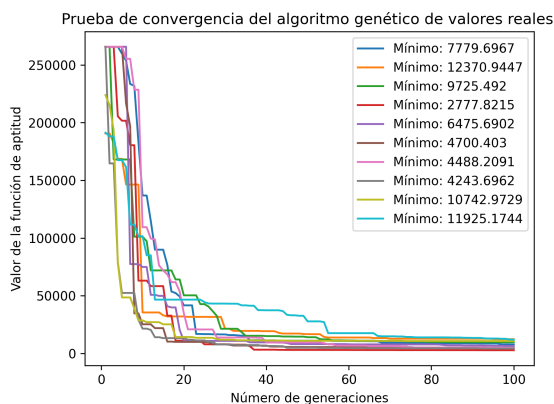
Se realizaron 6 pruebas mostradas en la Figura 7.9, que consisten en la ejecución de la estrategia, cambiando el tamaño de paso σ y el parámetro c . En las figuras 7.9a y 7.9b, la configuración utilizada fue de $\sigma = 0.5$, con $c = 0.817$ y $c = 0.99$, respectivamente. Mientras que en las figuras 7.9c y 7.9d, se utilizó el valor de la desviación estándar del individuo inicial con ayuda de la función `numpy.std` y los valores de $c = 0.817$ y 0.99 , respectivamente. Por último se utilizó un valor del tamaño de paso de $\sigma = 0.005$ en las figuras 7.9e y 7.9f, con parámetros $c = 0.817$ y de $c = 0.99$ respectivamente.

En las ejecuciones realizadas se observa una convergencia casi inmediata en las primeras evaluaciones, desde valores de amplitud de escala mayores a 1×10^7 , hasta valores que rondan en la escala de 1×10^3 . El problema observado entre todas las ejecuciones realizadas, fue que después de ciertas evaluaciones la convergencia al óptimo ocurre más lenta, usando un número de evaluaciones elevadas para llegar al objetivo y extendiendo el tiempo de ejecución del algoritmo.

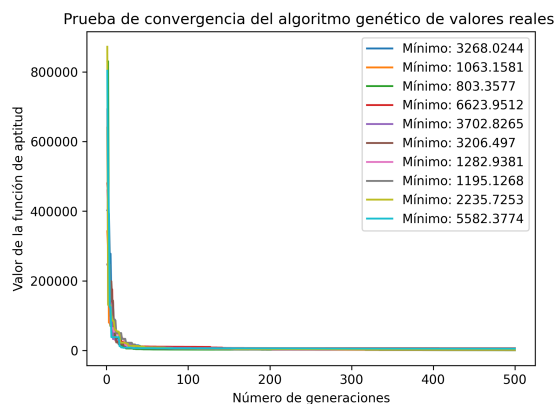
Para la prueba de la Figura 7.9a, el tiempo promedio de ejecución fue de 1153.26 segundos, en la Figura 7.9b fue de 1382.19 segundos, en la Figura 7.9c el tiempo promedio fue de 1546.15 segundos, en las pruebas de la Figura 7.9d fue de 4113.11 segundos, en las pruebas de 7.9e el tiempo fue de 1109.39 segundos y en la Figura 7.9f fue de 1660.45 segundos. Si bien la estrategia logra llegar al objetivo, el tiempo de evaluaciones es alto en todos los grupos de ejecuciones, rondando los 20 minutos en promedio.

7.2.5. Optimización de la estrategia evolutiva (1+1)-Cholesky-CMA

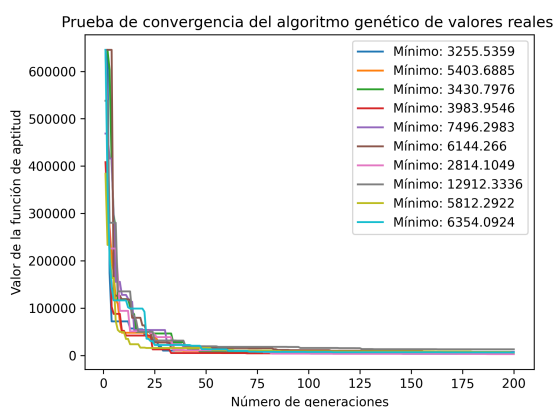
En contexto del algoritmo de la estrategia (1+1)-Cholesky-CMA, se realizó la prueba con los valores predefinidos por T. Bäck et. al. [37], ya que éstos valores cambiarán de acuerdo a la evolución del individuo. Únicamente se cambiará el tamaño de paso σ inicial, para verificar si este afectará la búsqueda del óptimo. En el grupo de figuras 7.10, se observan 4 grupos de pruebas realizadas con la estrategia (1+1)-Cholesky-CMA. En la imagen 7.10a, se utilizó el valor de tamaño de paso $\sigma = 0.005$, mientras que en 7.10b, $\sigma = 0.5$. En la Figura 7.10c, se utilizó el valor de la desviación estándar del individuo con `numpy.std`. Este valor varía en un promedio de 5.1. Por último en la Figura 7.10d, se utilizó un valor de $\sigma = 50$. Se puede apreciar en todas las figuras, sin



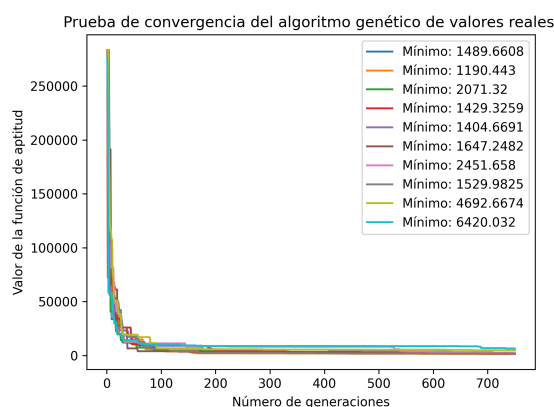
(a)



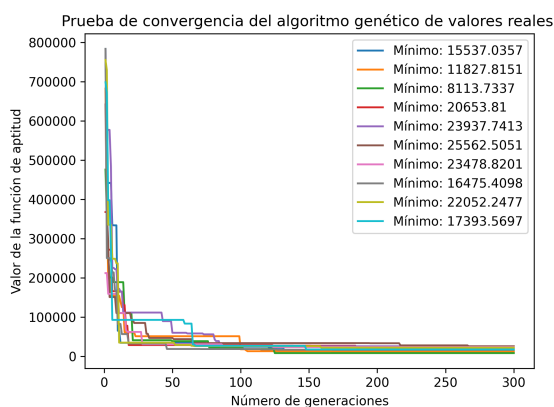
(b)



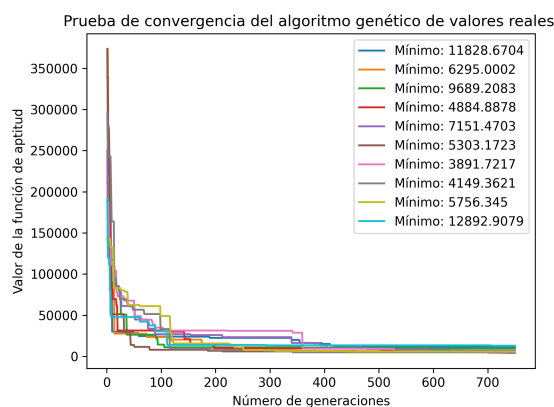
(c)



(d)

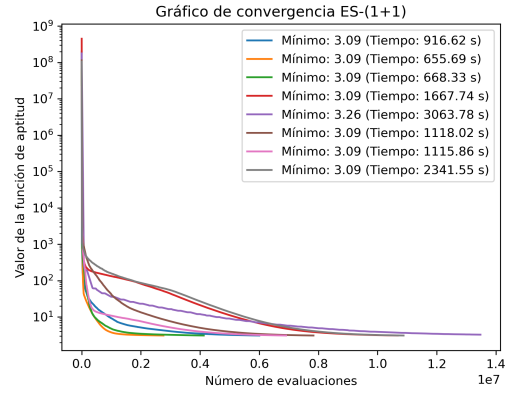
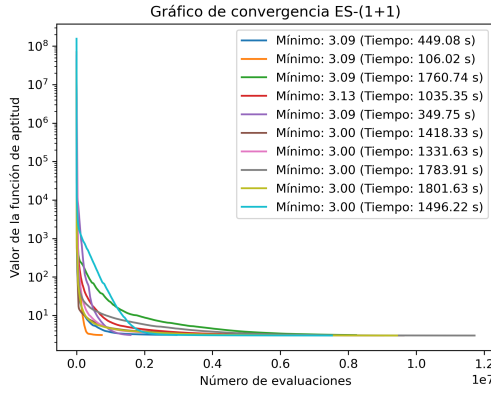


(e)

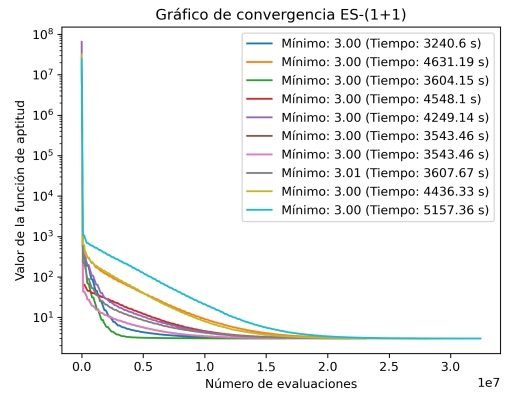
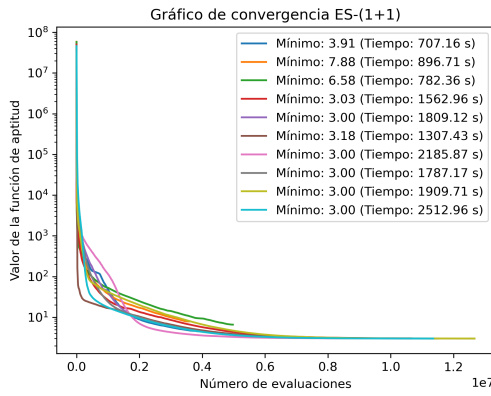


(f)

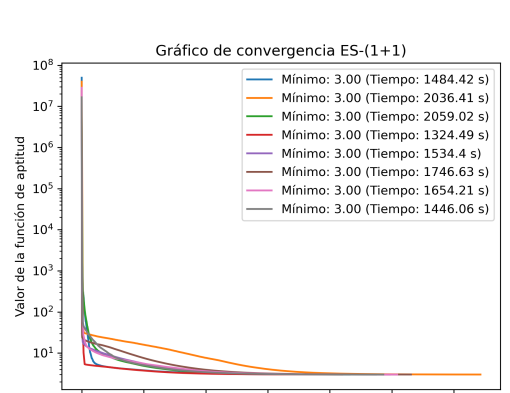
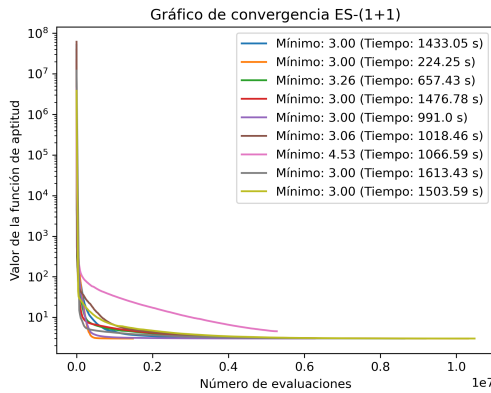
Figura 7.8: Gráficos que muestran el comportamiento del algoritmo genético de valores reales con 10 poblaciones de (a): 300 individuos cada uno, con 100 generaciones, (b) 750 individuos y 500 generaciones, ambos con el 5% de probabilidad de mutación. (c) 500 individuos con 200 generaciones, (d) 1000 individuos con 750 generaciones, ambos con el 10% de probabilidad de mutación. (e) 500 individuos con 300 generaciones y (f) 1000 individuos con 750 generaciones, ambos con el 20% de probabilidad de mutación.



(a) $\sigma = 0.5, c = 0.817, 6.62 \times 10^7$ evaluaciones. (b) $\sigma = 0.5, c = 0.99, 6.92 \times 10^6$ evaluaciones.



(c) $\sigma = \text{std}(), c = 0.817, 9.19 \times 10^6$ evaluaciones. (d) $\sigma = \text{std}(), c = 0.99, 2.26 \times 10^6$ evaluaciones.

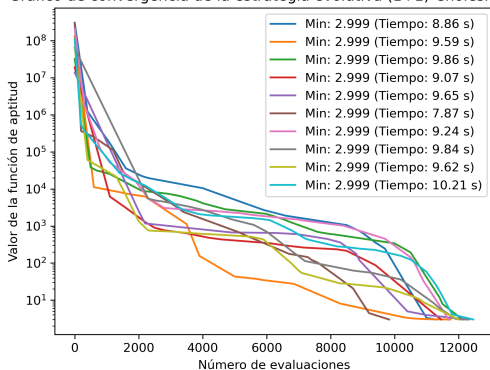


(e) $\sigma = 0.005, c = 0.817, 7.11 \times 10^6$ evaluaciones.

(f) $\sigma = 0.005, c = 0.99, 9.05 \times 10^6$ evaluaciones.

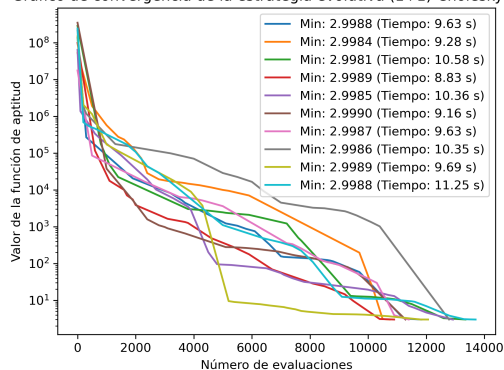
Figura 7.9: Gráficos que muestran el comportamiento de la estrategia evolutiva (1+1), cambiando el tamaño de paso, y el parámetro c .

Gráfico de convergencia de la estrategia evolutiva (1+1)-Cholesky-CMA



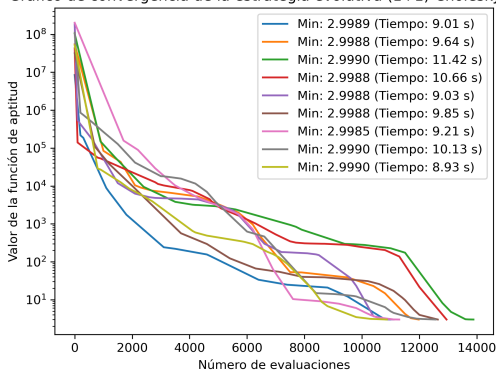
(a) $\sigma = 0.005$

Gráfico de convergencia de la estrategia evolutiva (1+1)-Cholesky-CMA



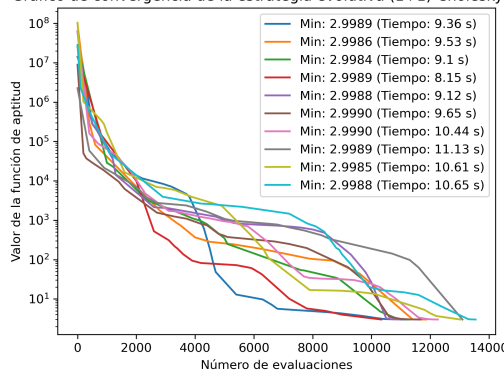
(b) $\sigma = 0.5$

Gráfico de convergencia de la estrategia evolutiva (1+1)-Cholesky-CMA



(c) $\sigma = std()$

Gráfico de convergencia de la estrategia evolutiva (1+1)-Cholesky-CMA



(d) $\sigma = 50$

Figura 7.10: Gráficos que muestran la convergencia de la estrategia evolutiva (1+1)-Cholesky-CMA con diferentes tamaños de paso inicial.

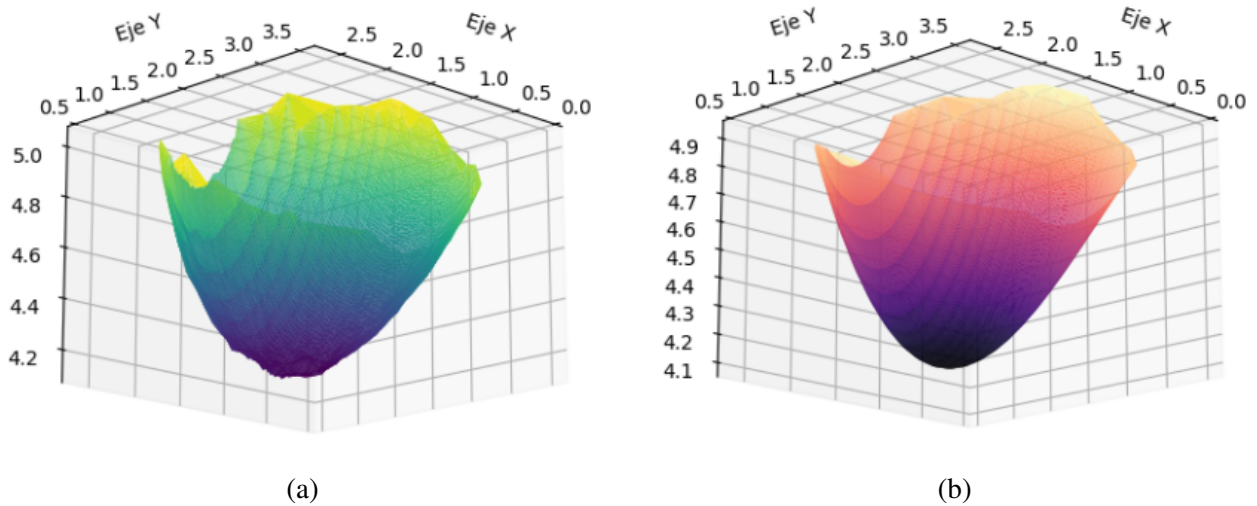


Figura 7.11: (a): Gráfico obtenido al simular los datos experimentales obtenidos por A. Santiago Alvarado et. al. [44]. (b) Gráfico producido al simular los datos evaluados con los coeficientes óptimos obtenidos por el algoritmo híbrido.

importar el valor inicial del tamaño de paso, el comportamiento de la estrategia para este problema resulta ser similar. El tiempo promedio entre todas las ejecuciones fue de 9.87 segundos, mientras que el mejor tiempo medido fue con el del valor del tamaño de paso de $\sigma = 0.005$, con un valor promedio de 9.38 segundos. En esta prueba también se puede apreciar que todas las ejecuciones lograron el óptimo, en un tiempo mínimo, siendo el candidato ideal para la prueba 2 (prueba de maximización de correlación entre birronchigramas).

7.2.6. Pruebas del algoritmo híbrido

De acuerdo con los resultados obtenidos en las pruebas de eficiencia, el algoritmo híbrido fue programado con la estrategia evolutiva (1+1)-Cholesky-CMA y el algoritmo genético de valores reales, tal como se muestra en el diagrama de flujo 6.9. En las pruebas realizadas, se logró encontrar el valor óptimo, sin necesidad de pasar por el algoritmo genético de valores reales.

A continuación se muestran los resultados obtenidos al ejecutar el algoritmo híbrido en búsqueda de los coeficientes óptimos. Anteriormente se mencionó que con la función de aptitud el error obtenido a partir de los coeficientes publicados es de 2.9954, mientras que utilizando los coeficientes encontrados con el algoritmo híbrido se encontró un error mínimo de 2.9988. La Figura 7.11a muestra la superficie generada por los datos experimentales, mientras que en la Figura 7.11b, se muestra la superficie generada de la superficie obtenida al graficar los datos usando los coeficientes hallados con el algoritmo híbrido. En la tabla 7.2 se comparan los valores del mejor individuo obtenido por el algoritmo híbrido, siendo 9.19 segundos el tiempo necesario para llegar al valor reportado.

Tabla 7.2: Tabla que muestra los coeficientes de los valores publicados el artículo de A. Santiago et. al. [44] y el de los calculados con el algoritmo híbrido.

Término	Artículo (mínimos cuadrados)	Algoritmo híbrido
constante	5.6683	5.6863
x	0.0290	0.0214
y	-0.9389	-0.9655
x^2	-0.1130	-0.1194
xy	-0.5877	-0.5808
y^2	0.2565	0.2687
x^3	0.1241	0.1245
x^2y	-0.0022	-0.0033
xy^2	0.1467	0.1456
y^3	-0.0076	-0.0094
Error	2.9954	2.9988

7.3. Proceso de eficiencia del algoritmo híbrido para el caso 2.

A pesar de que el funcionamiento en la búsqueda del óptimo es similar tanto para la maximización y minimización, se requieren modificar algunas funciones, para que en vez de que el algoritmo busque el mínimo, se enfoque en buscar el máximo. El tipo de problema que se abarcará tiene diferencias significativas como lo es su función de aptitud y el número de variables a tratar, por lo que es necesario hacer una nueva búsqueda de parámetros que permitan ejecuciones ágiles, encontrando el valor óptimo en el menor tiempo posible.

7.3.1. Parámetros óptimos en algoritmo genético de valores reales

Se realizaron 6 pruebas con diferentes parámetros para encontrar la mejor configuración del algoritmo genético de valores reales. En cada prueba se modificó el valor del número de individuos y también la probabilidad de mutación. También se configuró al algoritmo detenerse al encontrar el valor óptimo de 0.935, o bien que se detenga después de las 500 evaluaciones. Los resultados obtenidos se pueden observar en el grupo de figuras 7.12.

Sea cualquiera la configuración elegida, el comportamiento de la convergencia fue similar. La única diferencia fue entre los tiempos medidos que se detallarán a continuación. En la Figura 7.12a que tiene 100 individuos y 20% de probabilidad de mutación, se observó que el tiempo de ejecución ronda en promedio de 378.81 segundos. En la Figura 7.12b las pruebas con 30 individuos y 20% de probabilidad de mutación se observó un tiempo promedio de 278.96 segundos. En la Figura 7.12c, las pruebas se realizaron con 50 individuos con el 10% de probabilidad de mutación, el tiempo promedio fue de 98.76 segundos. En la Figura 7.12d, las pruebas se configuraron para 30 individuos y 10% de probabilidad de mutación y se obtuvo un promedio de 164.26 segundos. Por otra parte en la Figura 7.12e, las pruebas se configuraron con 50 individuos y el 7% de probabilidad de mutación, el promedio de tiempo fue de 98.76 segundos. Para finalizar, el conjunto de pruebas de la Figura 7.12f, utilizando 30 individuos y el 7% de probabilidad de mutación, se obtuvo un tiempo promedio de 129.16 segundos.

Se observa que en algunas pruebas, se requiere un número menor de evaluaciones que otras.

Esto puede deberse a que existe algún o algunos individuos que tienen mejores condiciones endógenas, es decir, que algunos de los cromosomas están ya cercanos al óptimo, y que esta población con individuos mejor evolucionados están generando mejores descendencias. Cabe recalcar que las trayectorias mostradas en el conjunto de figuras, se muestra el proceso donde el mejor individuo es seleccionado a través de las generaciones, lo que manifiesta que en los caminos cortos, se ocupa un número menor de evaluaciones y por lo tanto un menor tiempo de ejecución. También se puede apreciar que hay algunas trayectorias de evolución en donde, si bien el mejor individuo genera un valor óptimo de correlación, el programa se detendrá hasta alcanzar un valor mayor o igual al objetivo.

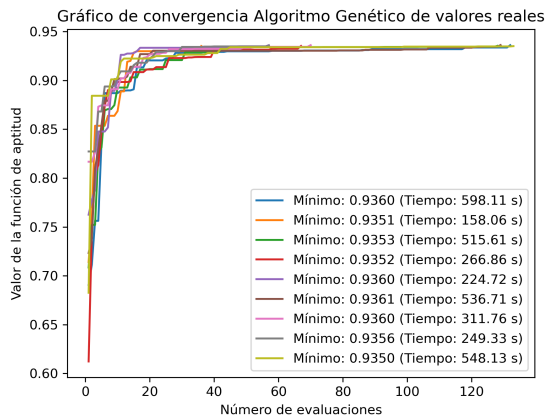
7.3.2. Optimización de la estrategia evolutiva (1+1)

Se efectuaron 6 pruebas con diferente conjunto de parámetros iniciales, estas pruebas se muestran en la Figura 7.13. Como se describió anteriormente, la función de evaluación requiere más recursos que para el caso de minimización, y por lo tanto, cada evaluación tarda en realizarse. En la Figura 7.13a se realizaron las evaluaciones por individuo con un tamaño de paso de $\sigma = 0.005$ y parámetro $c = 0.817$, con un tiempo promedio de 284.55 segundos. En la Figura 7.13b se muestran las ejecuciones de la estrategia con el tamaño de paso inicial $\sigma = 0.005$ y $c = 0.99$ con un tiempo promedio de 790.12 segundos. En la Figura 7.13c se muestran las ejecuciones con el tamaño de paso inicial $\sigma = 0.5$ y $c = 0.817$, con un tiempo promedio 786.17 segundos. En la Figura 7.13d el tamaño de paso inicial fue de $\sigma = 0.5$, con $c = 0.99$, en donde el tiempo promedio entre ejecuciones fue de 1003.86 segundos. En las dos últimos grupos de ejecuciones, se utilizó el tamaño de paso inicial generado por la función `numpy.std`. En la imagen 7.13e el parámetro c fue de 0.817, con un tiempo promedio de 391.76 segundos y en la Figura 7.13f se realizaron las pruebas con $c = 0.99$ con un tiempo promedio de 377.37 segundos.

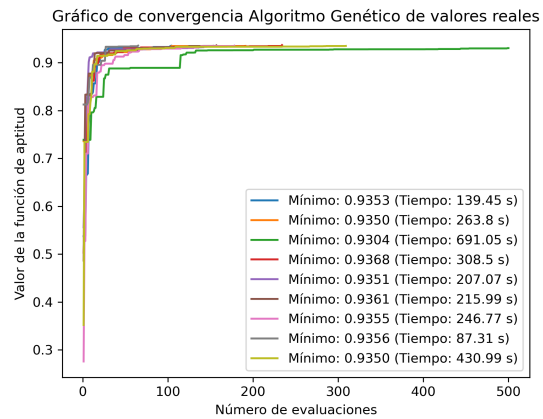
En las ejecuciones de esta prueba de forma general se observa que la evolución de varios individuos no pudieron encontrar valores óptimos. Esto pudiera deberse a las características del individuo inicial que no le permiten evolucionar de forma adecuada, es decir, que la calidad y cercanías a la solución no le permitieron encontrar un camino de evolución adecuado para la búsqueda de una mejor solución. Por el contrario, los mejores individuos logran llegar al valor óptimo en un tiempo relativamente mucho menor a los otros dependiendo de la cercanía a la solución. Esto se puede observar en la Figura 7.13a que solo 2 de las ejecuciones no convergieron a una solución óptima, mientras que por el contrario en la Figura 7.13b, se observan 2 soluciones óptimas, mientras que el resto no convergieron. En las figuras 7.13c, 7.13d y 7.13e se aprecia que tuvieron un comportamiento similar con poco más de la mitad de éxitos en su búsqueda. Y por último la Figura 7.13f tuvo un comportamiento aceptable pero les llevó más tiempo a algunos individuos encontrar el objetivo propuesto.

7.3.3. Pruebas con la estrategia evolutiva (1+1)-Cholesky-CMA

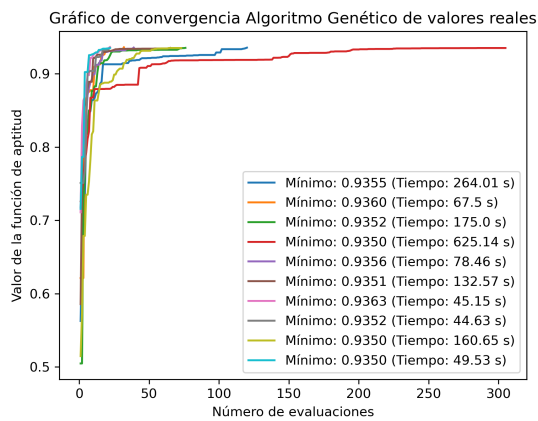
Para hacer muestra del rendimiento de la estrategia evolutiva (1+1)-Cholesky-CMA, se probaron distintas configuraciones cambiando únicamente el tamaño de paso inicial, ya que el autor no hace sugerencia o alusión si otros parámetros pueden ser modificados [37]. Como se observó que el tamaño de paso es variable a lo largo de las evaluaciones, se asume que un tamaño de paso inicial podría causar una diferencia al momento de encontrar un buen camino a la solución.



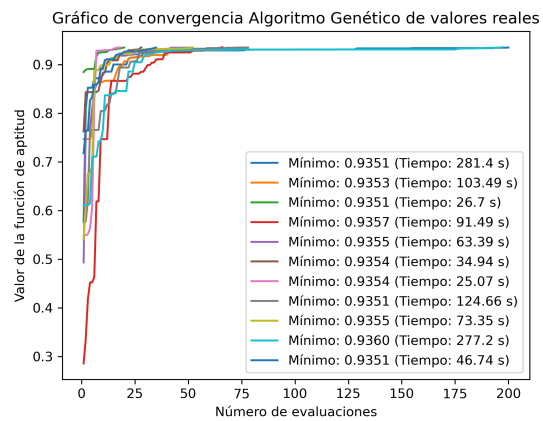
(a) 100 individuos, prob. de mutación: 20%. Tiempo promedio: 378.81 s.



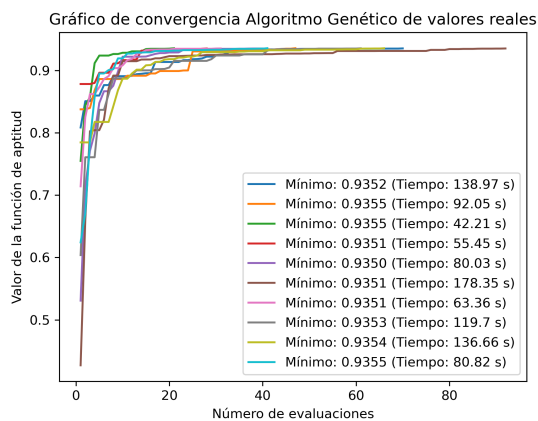
(b) 30 individuos, prob. de mutación: 20%. Tiempo promedio: 287.96 s.



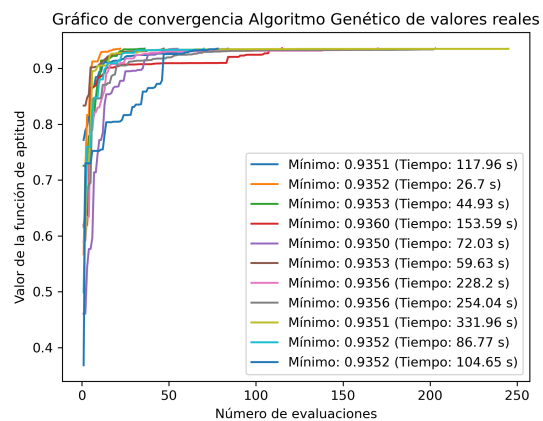
(c) 50 individuos, prob. de mutación: 10%. Tiempo promedio: 164.26 s.



(d) 30 individuos, prob. de mutación 10%. Tiempo promedio 104.4 s.

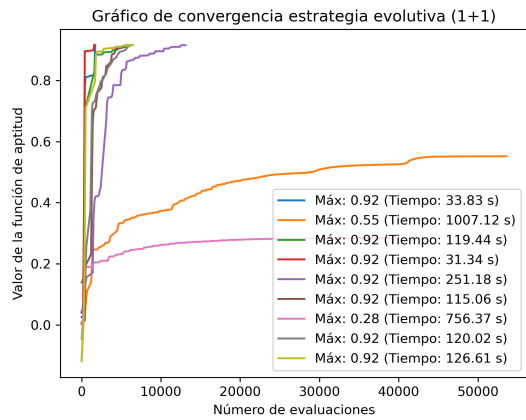


(e) 50 individuos, prob. de mutación: 7%. Tiempo promedio: 98.76 s.

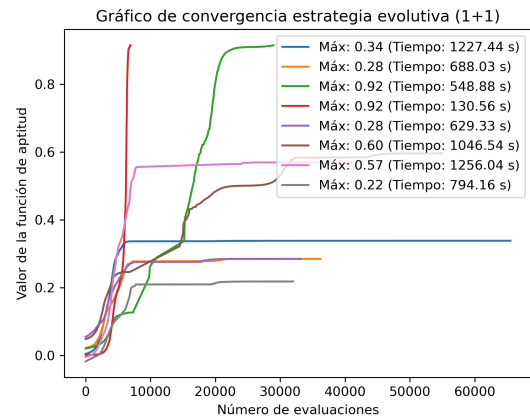


(f) 30 individuos, prob. de mutación 7%. Tiempo promedio 129.16 s.

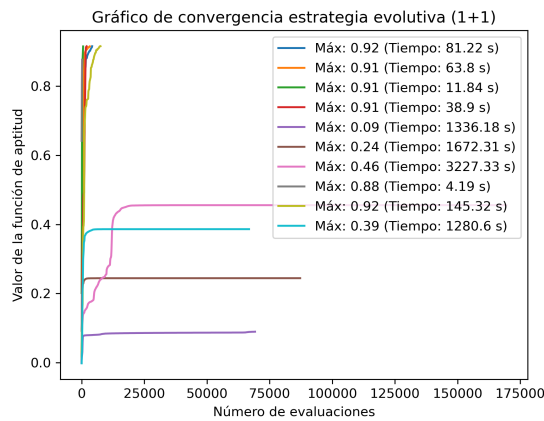
Figura 7.12: Pruebas de convergencia y rendimiento del algoritmo genético de valores reales, usando como variaciones el número de individuos y la probabilidad de mutación por cada prueba.



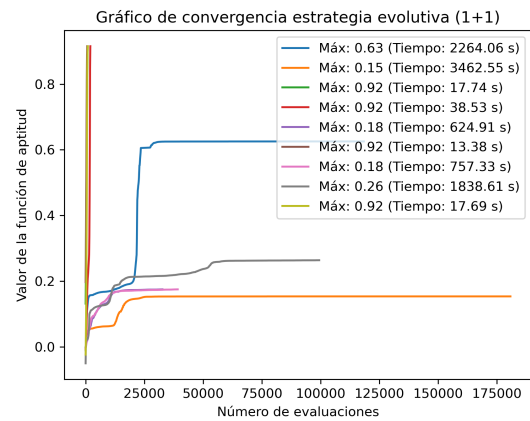
(a) $\sigma = 0.005, c = 0.817$



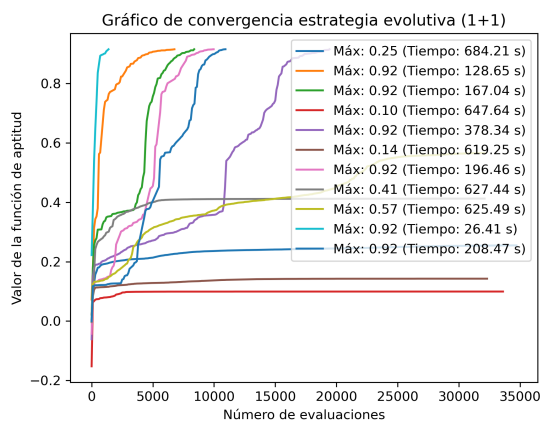
(b) $\sigma = 0.005, c = 0.99$



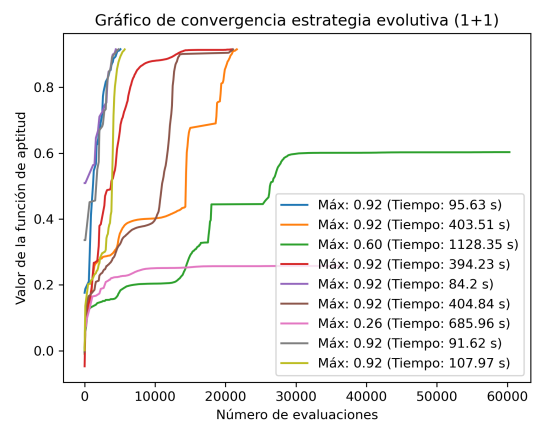
(c) $\sigma = 0.5, c = 0.817$



(d) $\sigma = 0.5, c = 0.99$



(e) $\sigma = \text{std}(), c = 0.817$



(f) $\sigma = \text{std}(), c = 0.99$

Figura 7.13: Pruebas realizadas para la maximización con la estrategia evolutiva (1 + 1).

En el grupo de figuras 7.14, se observan varios gráficos mostrando grupos de ejecuciones realizados con diferentes tamaños de paso inicial. En la Figura 7.14a se inició con un valor de tamaño de paso de 0.005, logrando un tiempo promedio de 111.76 segundos. En la Figura 7.14b se utilizó un tamaño inicial de 0.05, logrando llegar al óptimo en un promedio de 117.65 segundos. Mientras que en la Figura 7.14c, el tamaño de paso fue de 0.1, con un tiempo promedio de 127.27 segundos. En la Figura 7.14d se usó el valor de 0.5, logrando un tiempo de ejecución promedio de 77.2 segundos. En la Figura 7.14e el valor del tamaño de paso fue de 1.5, con un tiempo promedio de 146.23. Por último, la Figura 7.14f, se usó el valor de 5 y se encontraron valores óptimos a un promedio de 109.28 segundos.

De todas las pruebas realizadas se observa un comportamiento de convergencia en la mayoría de las ejecuciones. Sin embargo, hay algunas que no logran llegar al objetivo, ya que se observa en varias de las ejecuciones que ocurre una pérdida de progreso en valores específicos a lo largo de las evaluaciones. Esto se puede deber, al igual que en la estrategia evolutiva (1+1) un individuo que no logra converger de manera eficiente, y que la mutación realizada no es capaz de conducirlo a una solución óptima.

De las opciones utilizadas para verificar el comportamiento de esta estrategia, se sugiere utilizar un tamaño de paso de 0.5, ya que es la que presenta menor número de individuos que no logran converger a una solución. También es el que logra tener un promedio de 77.20 segundos, siendo el mejor tiempo 26.64 segundos. Por otra parte, las ejecuciones realizadas con el tamaño de paso inicial de 0.005 (figura 7.14a), tuvieron un rendimiento aceptable, en los que el tiempo de ejecución fue de 111.76 segundos en promedio, con el mejor tiempo de 13.17 segundos, que también podría tenerse a consideración.

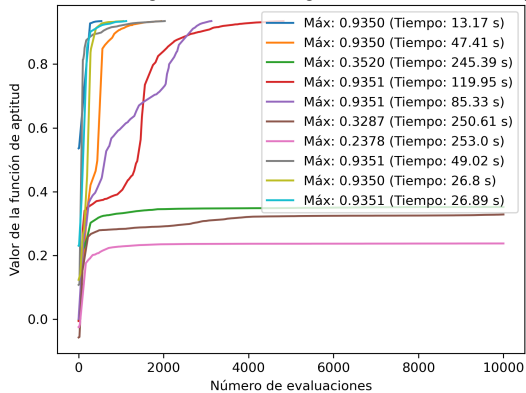
En conclusión, usar el valor de tamaño de paso inicial 0.5, sería una opción viable para incluir la estrategia dentro del algoritmo híbrido.

7.3.4. Pruebas del algoritmo híbrido

Para producir las pruebas del algoritmo híbrido, se enfocó en observar la consistencia de los resultados obtenidos y cuánto tiempo se tardó en obtenerlos. La tabla 7.3 muestra 5 ejecuciones realizados con el algoritmo híbrido, las aberraciones medidas de forma manual de los espejos realizados por C. Ortiz Lima et. al [9] y el tiempo de ejecución del algoritmo. Esta tabla muestra, detalles importantes a tomar en cuenta:

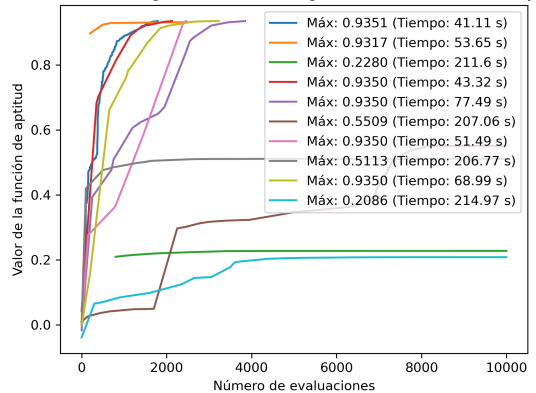
- La uniformidad en algunos de los coeficientes se encuentra presente a través de las 5 evaluaciones, es decir, los valores entre estos coeficientes no varía en gran cantidad, por lo que se puede estimar un promedio entre estos valores. Estas aberraciones son Tilt x, Defoco, Astigmatismo primario a 0°, Coma primaria en x y Esférica. Por el contrario, hay algunos otros valores entre coeficientes que no presentan alguna uniformidad, pero si se encuentran dentro de los rangos establecidos.
- Se ha logrado encontrar un valor óptimo de correlación, sin embargo, esta correlación puede aumentar al realizar una búsqueda hasta una convergencia aceptable.
- El tiempo de búsqueda no rebasa los 30 segundos. El tiempo de evaluación más pequeña entre las ejecuciones fue de 12.03 segundos, y el valor más grande fue de 27.5 segundos.

Gráfico de convergencia de la estrategia evolutiva (1+1)-Cholesky-CMA



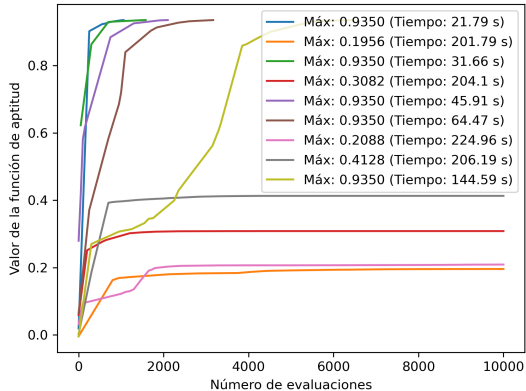
(a) Tamaño de paso inicial: 0.005.

Gráfico de convergencia de la estrategia evolutiva (1+1)-Cholesky-CMA



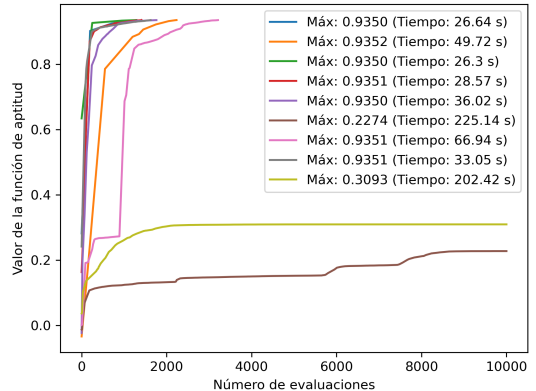
(b) Tamaño de paso inicial: 0.05.

Gráfico de convergencia de la estrategia evolutiva (1+1)-Cholesky-CMA



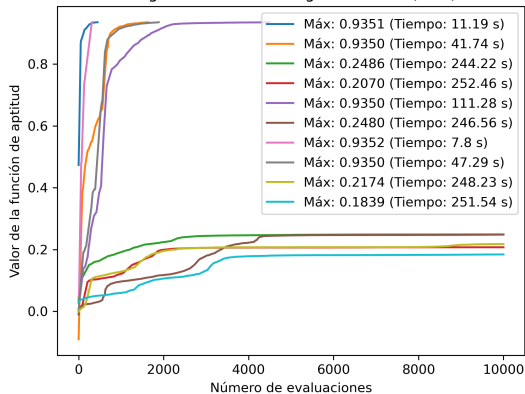
(c) Tamaño de paso inicial: 0.1.

Gráfico de convergencia de la estrategia evolutiva (1+1)-Cholesky-CMA



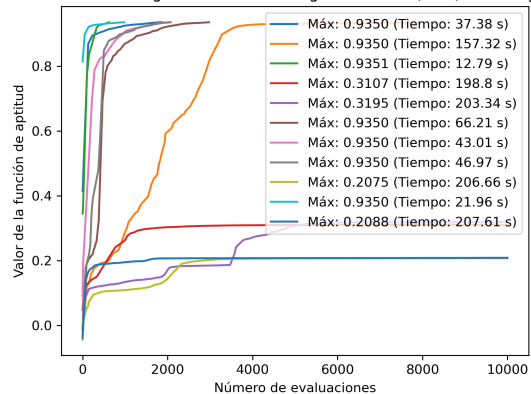
(d) Tamaño de paso inicial: 0.5

Gráfico de convergencia de la estrategia evolutiva (1+1)-Cholesky-CMA



(e) Tamaño de paso inicial: 1.5.

Gráfico de convergencia de la estrategia evolutiva (1+1)-Cholesky-CMA



(f) Tamaño de paso inicial: 5.

Figura 7.14: Pruebas realizadas para la maximización con la estrategia evolutiva (1+1)-Cholesky-CMA.

Tabla 7.3: Tabla comparativa de los resultados obtenidos en el artículo [9] con los obtenidos en el algoritmo híbrido.

Aberración	AG original	Coefficientes (Algoritmo híbrido)				
Tilt y	18.068	-9.2406	8.0738	3.9076	1.4940	1.3266
Tilt x	-9.0820	30.502	-28.348	-25.6628	-30.892	-28.537
Defoco	1660.2	1654.4	1656.5	1654.0	1653.8	1655.8
Ast. prim 45°	33.1939	-16.493	-0.36629	11.539	13.373	15.118
Ast. prim 0°	-1.4664	-2.4854	-3.7502	-2.4919	-2.5882	-3.9252
Coma primaria y	-31.1356	3.3088	-32.476	7.6608	57.137	18.047
Coma primaria x	-1.8890	-2.1196	-1.2320	-2.1965	-1.7577	-1.6029
Trefoil en y	23.6481	8.1032	1.4761	-17.3145	-2.4311	-5.5477
Trefoil en x	-6.3538	-52.636	30.913	-59.656	35.511	-1.4513
Esférica	-3.4790	-2.9066	-3.1833	-2.9847	-3.0269	-3.8003
Ast. sec. 0°	-0.5692	-5.7285	18.663	9.6479	-142.5	1.5127
Ast. sec. 45°	1.5588	0.33863	-1.2695	6.1401	-6.7746	6.45
Quatrefoil vertical	-2.9791	16.34	38.097	-19.0952	20.777	-1.0838
Quatrefoil oblicuo	1.3263	-24.135	4.3452	-7.2958	-35.837	4.5369
Coef. Correlación	0.9314	0.935	0.9351	0.935	0.9351	0.935
Tiempo (s)	>30 min.	12.03	16.32	20.97	21.2	27.5

7.3.5. Pruebas en rangos de búsqueda.

La personalización del espacio de búsqueda también juega un papel fundamental en el hallazgo de los mejores coeficientes para una solución óptima. En la siguiente prueba, se utilizaron los límites definidos en el espacio de búsqueda de la tabla 7.1, los cuales se aumentaron multiplicando por 2, 3, 5 y por 10 veces el valor original para verificar la convergencia en cada caso. Se usaron los parámetros seleccionados para el algoritmo híbrido.

En la Figura 7.15a se aprecia la ejecución del algoritmo con los límites tanto inferior como superior al doble de su valor. Se observa que la mitad de ellos llegaron al valor objetivo, mientras que los demás valores en su mayoría llegaron a valores cercanos que se pueden considerar como aceptables. Solo una de las ejecuciones obtuvo un valor menor a 0.9. En la Figura 7.15b se muestran 10 ejecuciones del algoritmo híbrido, con los límites ampliados al triple del valor original. Algunas de estas evaluaciones llegan al valor objetivo, mientras que otras logran llegar a un valor óptimo cercano aceptable. Se puede observar también que para llegar al valores objetivos o cercanos a este, el algoritmo requirió de mayor tiempo para entregar los resultados mostrados. Por otra parte, en la Figura 7.15c se realizaron 10 pruebas en donde los rangos son 5 veces el valor original tanto superior como inferior. Se puede observar que el algoritmo logra llegar al valor objetivo en algunas ejecuciones, pero también hubo otras que llegaron a valores inferiores. Entre estas ejecuciones se observó que 6 logran llegar a un valor óptimo y que el resto no pudieron converger de manera satisfactoria. Por último, en la Figura 7.15d, los valores de los límites se ampliaron 10 veces el valor original. Se puede observar que ninguna ejecución logra llegar al valor objetivo, y los valores cercanos a los que pudo llegar, no se consideran como valores aceptables para nuestro problema. También se aprecia que el tiempo para llegar a los valores mostrados, son altos a diferencia de usar los límites originales.

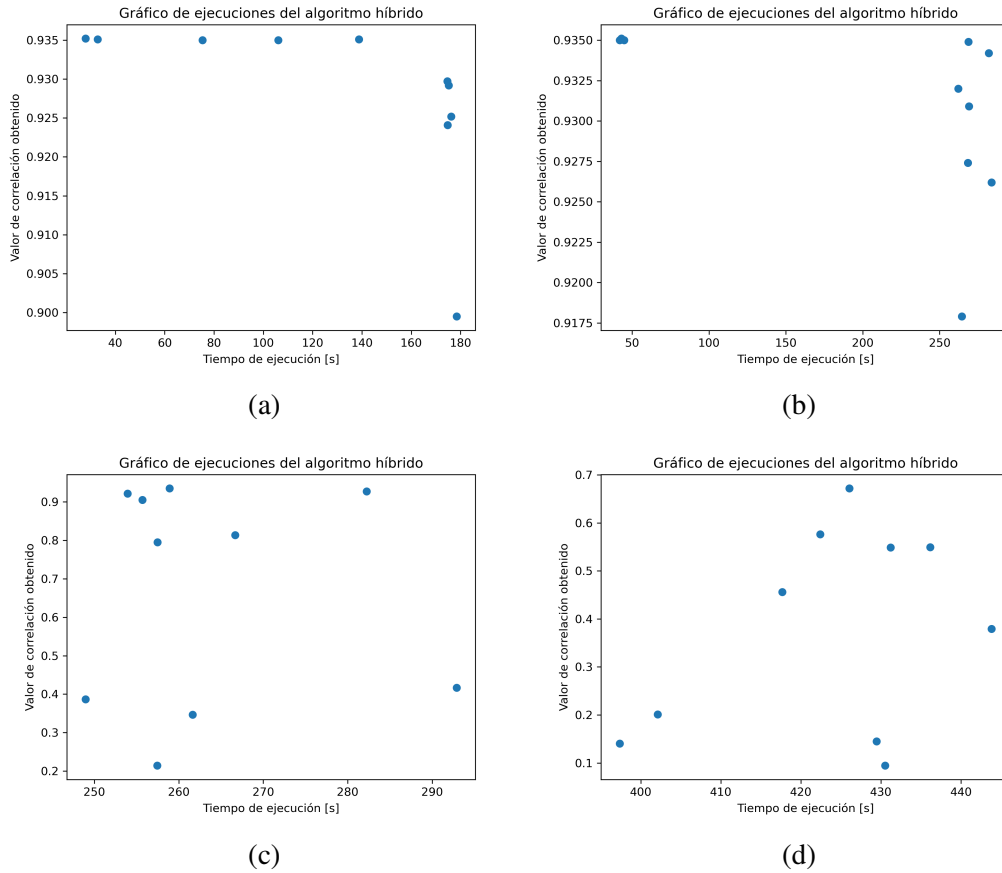


Figura 7.15: Pruebas del algoritmo híbrido cambiando el tamaño del espacio de búsqueda por factores de (a): 2, (b): 3, (c): 5 y (d): 10 veces el valor original de la tabla 7.1.

En conclusión, se observó que el algoritmo híbrido utilizando la estrategia evolutiva (1+1)-Cholesky es la que converge de manera eficiente a una solución óptima en el menor tiempo. El uso del algoritmo genético se puede considerar aplicable cuando la búsqueda por la estrategia no logra ser satisfactoria. En el siguiente capítulo, se observarán los resultados obtenidos a partir de los parámetros elegidos, además de pruebas de funcionamiento que permitirán observar la evolución de un birronchigrama al pasar por el algoritmo híbrido.

Capítulo 8

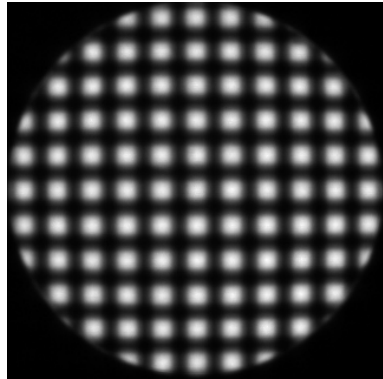
Resultados

Se llevaron a cabo pruebas donde se utilizaron los parámetros del algoritmo híbrido mostrados en la tabla 8.1. También se usaron los límites de cada coeficiente mostrados en la tabla 7.1 para encontrar una correlación óptima. Adicionalmente, se realizaron tres pruebas obtenidas del banco de 50 birronchigramas realizados por C. M. Ortiz Lima, A. Cordero Dávila y J. González García [9] que mostrarán el desempeño del software de evaluación de espejos.

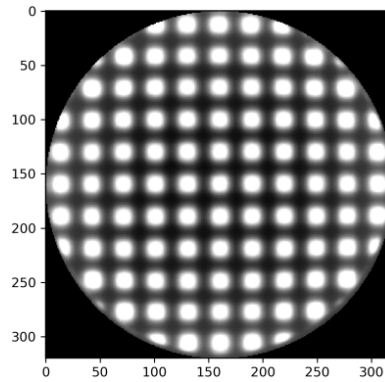
Tabla 8.1: Configuración de parámetros utilizados en el algoritmo híbrido para las pruebas con birronchigramas.

Parámetro	Valor
Número de individuos	30
Número de generaciones	10
Nuevo rango de búsqueda	15 %
Probabilidad de mutación	20 %
Evaluaciones de la estrategia	1500

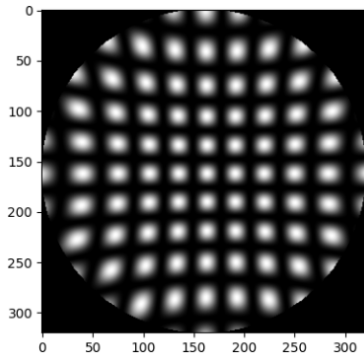
A continuación, el conjunto de figuras 8.1 muestra el proceso general del software, donde primero se presenta el birronchigrama experimental en la Figura 8.1a, y una vez siendo normalizada la irradiancia de éste en la Figura 8.1b, se procede a la búsqueda de la máxima correlación entre los datos del birronchigrama normalizado y del simulado. En la Figura 8.1c se muestra una imagen deformada que no presenta gran similitud con el birronchigrama experimental, ya que cuenta con un coeficiente de correlación de 0.4816. Mientras que en la Figura 8.1d, el coeficiente aumenta a 0.827, y a pesar que todavía la Figura presente defectos, ya se consigue una mayor similitud. Por otra parte en la Figura 8.1e se presenta una mayor correlación, obteniendo un coeficiente de 0.935, representando una similitud mayor y sin defectos perceptibles en comparación con las pruebas anteriores. En la tabla 8.2 se aprecia cada uno de los coeficientes de Zernike al ejecutar el software general. Cabe recalcar que estos coeficientes se obtuvieron al ejecutar el programa de forma independiente entre cada resultado y que cada simulación puede variar dependiendo del camino evolutivo que haya tomado.



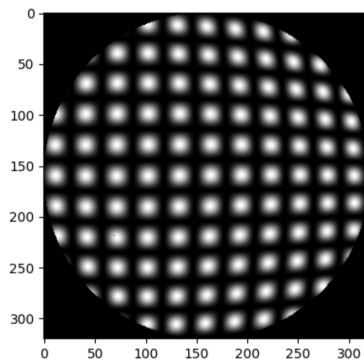
(a) Birrondigrama experimental



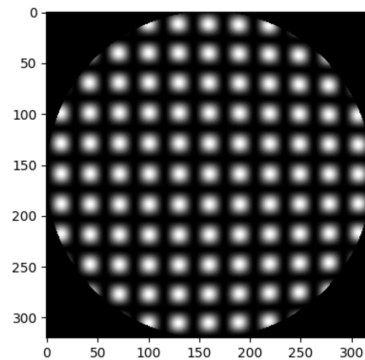
(b) Birrondigrama normalizado



(c) Coef. de correlación: 0.4816.



(d) Coef. de correlación: 0.827.



(e) Coef. de correlación: 0.935.

Figura 8.1: (a). Birrondigramas obtenidos de un banco de espejos parabólicos de 14 cm de diámetro (Parab49R), se usó una superficie esférica de referencia ($K = 0$), con una rejilla de 50.8 manchas por pulgada. (b): Normalización del birrondigrama experimental. (c): Simulación del birrondigrama con coeficiente de correlación de 0.4816, (d): Simulación del birrondigrama con coeficiente de correlación de 0.827 y (e): Simulación del birrondigrama con coeficiente de correlación de 0.935.

Tabla 8.2: Valores de Coeficientes de Correlación y Coeficientes de Zernike al realizar una prueba con el software general que muestra la evolución del algoritmo híbrido sobre el birronchigrama Parab49R.

Aberración	Coeficientes		
Tilt y	81.5383	58.96	3.8881
Tilt x	80.1016	-9.1528	-25.9384
Defoco	1682.0219	1654.7	1654.8208
Astigmatismo primario 45°	-12.1088	-67.84	18.8078
Astigmatismo primario 0°	-2.7535	9.6542	-3.1347
Coma primaria y	-5.7218	0.4167	-88.516
Coma primaria x	-11.2148	-23.047	-2.0063
Trefoil en y	-15.0002	33.234	7.5545
Trefoil en x	60.3026	175.14	75.2803
Esférica	30.4203	-9.9664	-2.1406
Astigmatismo secundario 0°	42.3368	-38.327	-10.1402
Astigmatismo secundario 45°	-12.1979	43.29	3.825
Quatrefoil vertical	-7.2762	72.13	-35.8383
Quatrefoil Oblicuo	-4.0835	26.918	5.9901
Coef. de correlación	0.4816	0.827	0.935

Tabla 8.3: Tabla de información de parámetros utilizados para evaluar los espejos bajo prueba.

Archivo	Diámetro [cm]	Periodo [l/mm]	Radio de curvatura [cm]	Distancia rejilla [cm]	Distancia rejilla (Corregido) [cm]
Parab_49R	14	2	192.61	184.72	188.95
Parab_13R	14	2	192.09	187.95	188.3
Parab_27R	14	2	192.46	184.65	188.88

8.1. Caso ideal

Se simularon los birronchigramas ideales en base a la información de la tabla 8.3, obtenida del trabajo del pulido de espejos parabólicos [9]. Al simular el birronchigrama con estos valores, se observó una diferencia notable con el número de manchas en los birronchigramas experimentales, como se pueden observar en las figuras 8.2b, 8.2e y 8.2h. Se descartaron los valores de diámetro del espejo y radio de curvatura como posibles causantes, ya que estos valores están preestablecidos desde el inicio del pulido de la superficie. También se descartó el periodo de la rejilla debido a que se usó la misma para todos los espejos. Es por eso que se modificó la distancia de la rejilla al espejo de forma manual, tratando de aproximar el valor necesario de este parámetro para que la simulación tenga la mayor similitud posible con el birronchigrama experimental. Una vez corrigiendo estas mediciones, las simulaciones de los birronchigramas 8.2a, 8.2d y 8.2g, se muestran en las figuras 8.2c, 8.2f y 8.2i, respectivamente.

Una vez obteniendo la aproximación adecuada para simular los birronchigramas experimentales de manera ideal, se procedió a realizar la obtención de los coeficientes de aberración, usando los parámetros simulados como medio para realizar la búsqueda de la correlación con estos datos. Los

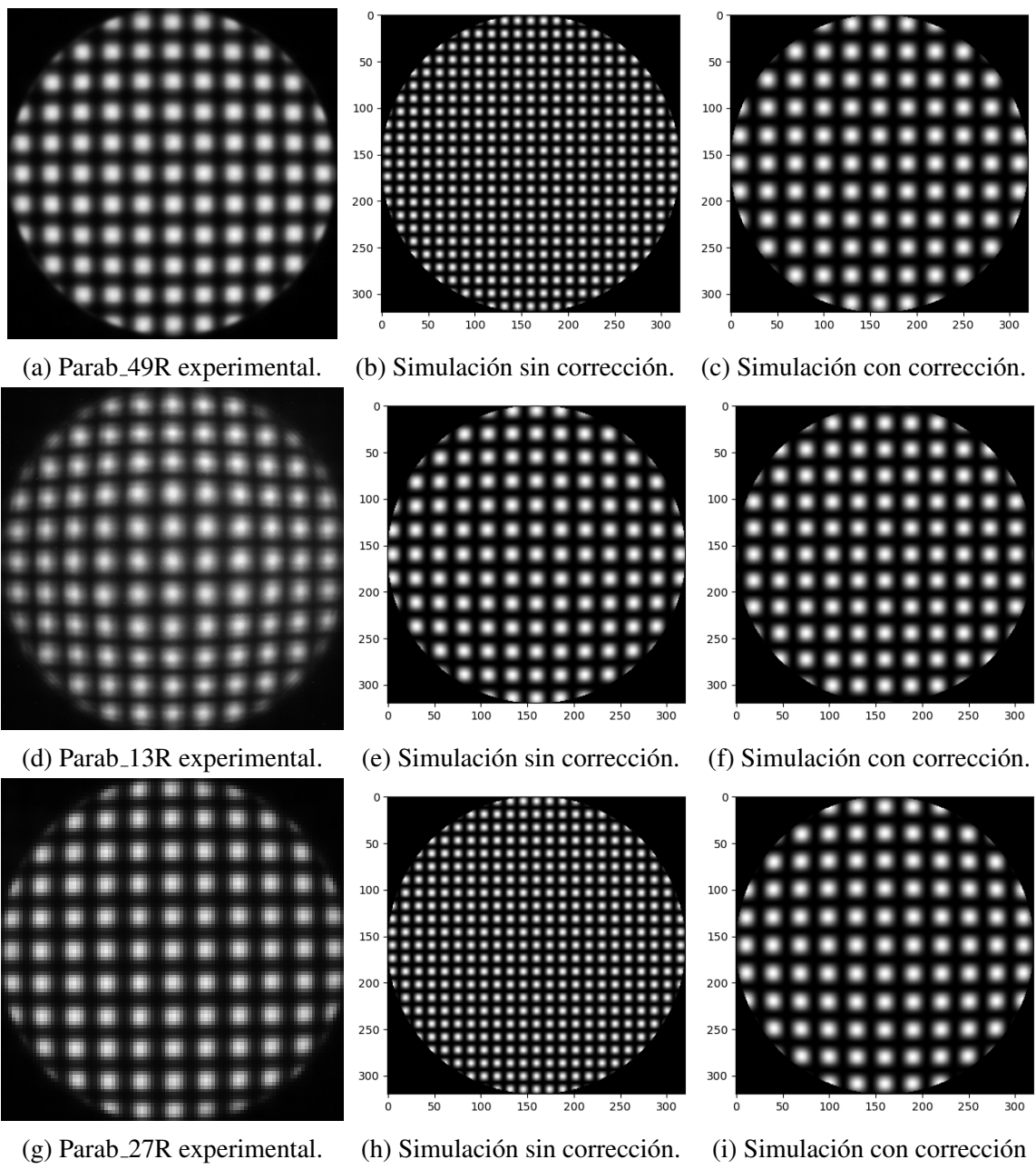


Figura 8.2: Comparación entre los birrunchigramas experimentales utilizando los parámetros utilizados en [9] y sus correcciones.

Tabla 8.4: Coeficientes obtenidos al ejecutar el software de evaluación de espejos, con el birronchigrama Parab_49R de forma ideal. 4 ejecuciones fueron realizadas con el valor de $K = 0$ y en las otras 4 se utilizó el valor de $K = -1$. También se muestran los coeficientes de correlación obtenidos y el tiempo de ejecución. Abreviaciones: AP = Astigmatismo primario, CP = Coma primaria, AS = Astigmatismo secundario, QFV = Quatrefoil vertical y QFO = Quatrefoil oblicuo.

Aberración	Coeficientes con $K = 0$				Coeficientes con $K = -1$			
Tilt x	-0.5082	-0.1563	-0.0978	0.0911	0.1545	-0.0064	-0.0737	-0.7894
Tilt y	3.3368	0.5124	-1.7582	-1.5562	1.5468	-2.4229	-0.2645	-0.5110
Defoco	2900.0	2846.9	2846.8	2846.9	2895.4	2895.4	2895.4	2895.3
AP 45°	12.779	-13.336	-5.140	17.650	-18.586	5.484	14.882	23.596
AP 0°	0.1061	-0.0868	0.1004	-0.1046	-0.3848	0.5265	0.0549	0.3934
CP y	-201.82	-33.484	100.79	90.429	-121.23	221.39	-10.971	62.266
CP x	-0.0152	0.1228	-0.0663	0.0561	-0.0131	0.0019	-0.0379	0.1576
Trefoil y	0.7231	1.8044	0.4355	-4.8588	-3.3677	-4.4364	-9.4381	-5.9682
Trefoil x	0.1229	3.7315	-0.2481	-0.7066	-15.066	1.4092	-0.6826	-2.8488
Esférica	13.266	0.7347	0.7948	0.8466	13.345	13.055	13.273	13.099
AS 45°	-114.28	13.036	-24.692	70.046	0.0750	-278.80	-0.0186	-211.48
AS 0°	-1.7002	2.5352	1.2618	-3.6661	2.1484	-1.4580	-1.3689	-4.2395
QFV	-2.5065	-1.0291	0.3556	1.4921	5.9435	-3.2788	-0.6426	-3.0926
QFO	-2.0945	-0.3354	2.0882	2.3582	-3.8228	0.5389	-0.5047	3.8589
Coef. Corr.	0.9991	0.9999	0.9998	0.9997	0.9994	0.9991	0.9998	0.9996
Tiempo (s)	155.24	76.7	152.52	148.76	150.38	149.25	152.51	154.02

resultados para el birronchigrama del archivo Parab_49R se muestran en la tabla 8.4, la cual muestra los coeficientes obtenidos al repetir la ejecución del programa 4 veces con la constante de conicidad $K = 0$ y otras 4 con el valor de la constante de conicidad de $K = -1$. Lo mismo ocurre con la tabla 8.5, donde se muestran los resultados de las simulaciones realizadas con el birronchigrama del archivo Parab_13R, y también con la tabla 8.6, la cual muestra los resultados de las simulaciones realizadas con el birronchigrama del archivo Parab_27R. Todas las pruebas de esta sección fueron realizadas con la misma configuración: El doble de los valores de los límites de la tabla 7.1, y se configuró la prueba para que el algoritmo se detenga hasta llegar al valor máximo de correlación, es decir, $C = 0.9999$.

8.2. Caso experimental

En las siguientes pruebas (experimentales), únicamente se modificó el valor de la constante de conicidad K . Esta constante cambia la superficie de referencia con la cual se calcularán las deformaciones correspondientes de cada superficie de los espejos mediante el cálculo de sus aberraciones.

Tabla 8.5: Coeficientes obtenidos al ejecutar el software de evaluación de espejos, con el birronchigrama Parab_13R de forma ideal. 4 ejecuciones fueron realizadas con el valor de $K = 0$ y en las otras 4 se utilizó el valor de $K = -1$. También se muestran los coeficientes de correlación obtenidos y el tiempo de ejecución. Abreviaciones: AP = Astigmatismo primario, CP = Coma primaria, AS = Astigmatismo secundario, QFV = Quatrefoil vertical y QFO = Quatrefoil oblicuo.

Aberración	Coeficientes con $K = 0$				Coeficientes con $K = -1$			
Tilt x	-0.1217	0.0153	-0.0730	0.3195	0.2348	0.1410	0.6820	-0.0234
Tilt y	1.0083	-0.4752	0.0531	0.3196	1.5708	-0.6619	4.4454	0.9084
Defoco	3046.1	3046.3	3046.2	3046.1	3015.2	3015.1	3015.3	3015.1
AP 45°	4.4355	-8.2789	14.284	-2.3338	60.295	-18.556	-26.194	-11.914
AP 0°	0.0453	-0.2502	-0.0763	-0.0038	-0.0472	0.0948	-0.3433	0.0338
CP y	-80.707	32.098	17.475	-31.151	-156.70	5.3648	-312.38	-59.394
CP x	-0.0141	-0.1050	0.0674	-0.1421	-0.2049	-0.0362	-0.0669	-0.0331
Trefoil y	-0.0138	-10.371	0.6448	-6.4702	1.0592	0.4317	2.9565	-9.6507
Trefoil x	0.6014	9.9239	1.4482	-5.9802	-2.6862	1.6054	-3.9779	-5.4130
Esférica	0.7904	0.7523	0.7842	0.9005	13.470	13.425	13.380	13.400
AS 45°	0.1917	-0.4301	140.68	76.923	70.832	0.0133	-42.966	0.0008
AS 0°	-0.6136	3.4679	-2.1075	0.2395	-12.259	3.5174	2.9496	2.3803
QFV	-0.5570	4.1125	-0.5381	3.3731	1.3756	0.9319	-0.3710	-3.5912
QFO	0.0628	2.6508	-0.3381	6.6092	-0.0100	-2.3099	-0.3116	5.0103
Coef. Corr.	0.9999	0.9997	0.9999	0.9998	0.9984	0.9999	0.9982	0.9997
Tiempo (s)	147.75	151.82	55.83	150.09	151.56	150.37	153.57	151.13

Tabla 8.6: Coeficientes obtenidos al ejecutar el software de evaluación de espejos, con el birronchigrama Parab_27R de forma ideal. 4 ejecuciones fueron realizadas con el valor de $K = 0$ y en las otras 4 se utilizó el valor de $K = -1$. También se muestran los coeficientes de correlación obtenidos y el tiempo de ejecución. Abreviaciones: AP = Astigmatismo primario, CP = Coma primaria, AS = Astigmatismo secundario, QFV = Quatrefoil vertical y QFO = Quatrefoil oblicuo.

Aberración	Coeficientes con $K = 0$				Coeficientes con $K = -1$			
Tilt x	0.0252	-0.1198	-0.0344	-0.6927	4.3380	0.0612	-0.2113	-0.5082
Tilt y	1.4417	3.7437	-0.9247	-0.0698	-7.4272	1.1239	-0.6229	3.3368
Defoco	2851.3	2851.5	2851.5	2851.6	2877.6	2900.0	2900.2	2900.0
AP 45°	-5.6757	-9.7104	18.037	7.9773	8.7481	6.9898	7.4616	12.779
AP 0°	0.0003	-0.0291	-0.2686	-0.1761	7.3916	0.2122	0.2510	0.1061
CP y	-90.256	-266.00	56.182	23.305	64.271	-66.826	5.2652	-201.82
CP x	-0.0168	-0.0231	0.0266	0.0092	-4.5764	-0.0116	0.0228	-0.0152
Trefoil y	-0.4159	-0.7075	2.5932	1.5045	-2.2207	-1.1093	7.5238	0.7231
Trefoil x	0.6676	0.3672	1.7693	6.0667	13.305	2.7004	-1.8604	0.1229
Esférica	0.7394	0.7330	0.6779	0.7888	26.212	13.230	13.289	13.266
AS 45°	-6.5633	-40.405	2.0771	54.322	80.369	-96.718	85.873	-114.28
AS 0°	0.5366	1.6296	-3.5704	-2.3042	6.1284	-1.7183	-1.4616	-1.7002
QFV	-0.2430	-1.3062	2.4200	-2.2307	-2.5875	-1.8329	-0.3964	-2.5065
QFO	-0.4083	1.2377	2.0960	-0.1259	-4.9863	0.8802	3.6181	-2.0945
Correlación	0.9998	0.9988	0.9998	0.9999	0.9462	0.9998	0.9999	0.9991
Tiempo (s)	152.26	158.65	160.49	53.00	153.40	154.73	52.25	155.24

Tabla 8.7: Coeficientes obtenidos al ejecutar el software de evaluación de espejos, 4 ejecuciones fueron realizadas con el valor de $K = 0$ y en las otras 4 se utilizó el valor de $K = -1$. Al final de la tabla se muestra el coeficiente de correlación al evaluar cada grupo de coeficientes. Abreviaciones: AP = Astigmatismo primario, CP = Coma primaria, AS = Astigmatismo secundario, QFV = Quatrefoil vertical y QFO = Quatrefoil oblicuo.

Aberración	Coeficientes con $K = 0$				Coeficientes con $K = -1$			
Tilt y	2.3521	2.0806	1.8497	2.1402	2.5825	2.9039	2.3815	2.0713
Tilt x	-28.2164	-26.92	-27.717	-27.142	-28.1153	-27.162	-27.227	-27.036
Defoco	1654.81	1654.8	1654.8	1654.8	1679.2	1679.3	1679.2	1679.2
AP 45°	12.4986	33.634	36.254	39.489	37.4496	1.4697	16.363	-5.7076
AP 0°	-2.1851	-2.2123	-1.8437	-2.181	-2.2372	-2.3313	-2.3527	-2.2193
CP y	15.5458	-87.551	-8.851	-67.377	26.0901	-53.411	-72.325	-71.786
CP x	-1.9408	-1.7121	-1.3281	-1.9045	-1.9541	-1.7811	-1.8765	-1.8868
Trefoil y	27.6556	27.504	16.392	21.632	19.3584	18.664	17.076	17.75
Trefoil x	6.8855	2.0826	-15.717	4.821	4.9616	4.4148	9.4872	8.9531
Esférica	-3.146	-3.1238	-3.1719	-3.1223	3.1449	3.088	3.0795	3.1915
AS 45°	-68.512	-4.3509	-11.796	54.691	-10.7355	-95.197	0.9248	-0.004
AS 0°	4.5161	4.6771	14.63	0.2552	17.4743	7.1235	4.2577	9.1064
QFV	-5.9988	-8.4424	-10.768	-5.531	-7.6499	-2.1794	-3.988	-7.941
QFO	6.5741	1.1332	-11.814	3.5044	29.636	-12.165	2.264	0.2062
coef. corr.	0.939	0.9392	0.9389	0.9391	0.9395	0.939	0.939	0.9387

8.3. Birronchigrama 1

El primer birronchigrama que se utilizó para mostrar la evolución del software general, es un espejo parabólico de 14 cm de diámetro, con radio de curvatura de 192.61 cm, distancia a la rejilla de 184.72 cm con un período en la rejilla de 2 líneas por milímetro, obtenido del banco de espejos (Parab49R). En la Figura 8.3 se muestra el resultado al procesar el birronchigrama, su normalización y el resultado de la simulación óptima con una superficie de referencia esférica, es decir, con $K = 0$. Mientras que en la Figura 8.4 se obtuvo el resultado al realizar el mismo procesamiento, con la diferencia de que la constante de conicidad se refiere a una superficie de referencia parabólica $K = -1$. En la tabla 8.7 se muestran 8 ejecuciones del programa, 4 muestran los coeficientes obtenidos al utilizar como constante de conicidad $K = 0$ y los otros 4 muestran los coeficientes obtenidos al utilizar la constante de conicidad $K = -1$.

8.4. Birronchigrama 2

El segundo birronchigrama es un espejo parabólico de 14 cm de diámetro, con radio de curvatura de 192.09 cm y con una distancia de la rejilla a la superficie del espejo de 187.95 cm (Parab.13R). Se utilizó como constante de conicidad el valor de $K = 0$ para hacer la búsqueda de coeficientes óptimos con este valor. La salida del programa para esta prueba se muestra en la Figura 8.5. También se realizó la misma prueba que la anterior, pero con constante de conicidad $K = -1$, donde los resultados se muestran en la Figura 8.6. Los coeficientes de correlación obtenidos al realizar

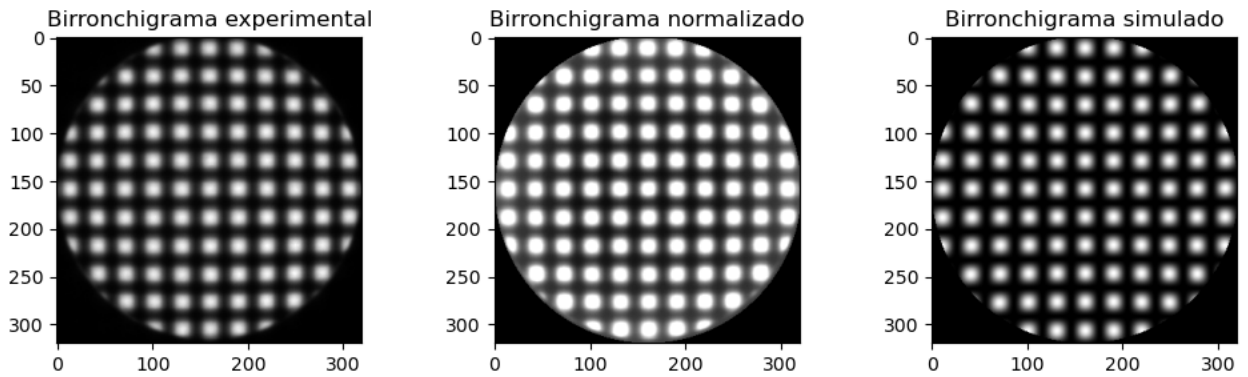


Figura 8.3: Resultados obtenidos al procesar el birronchigrama de un espejo de 14 cm de diámetro con radio de curvatura de 192.61 cm, distancia a la rejilla de 184.72 cm y constante de conicidad de $K = 0$ (Parab49R) con el software de evaluación de espejos.

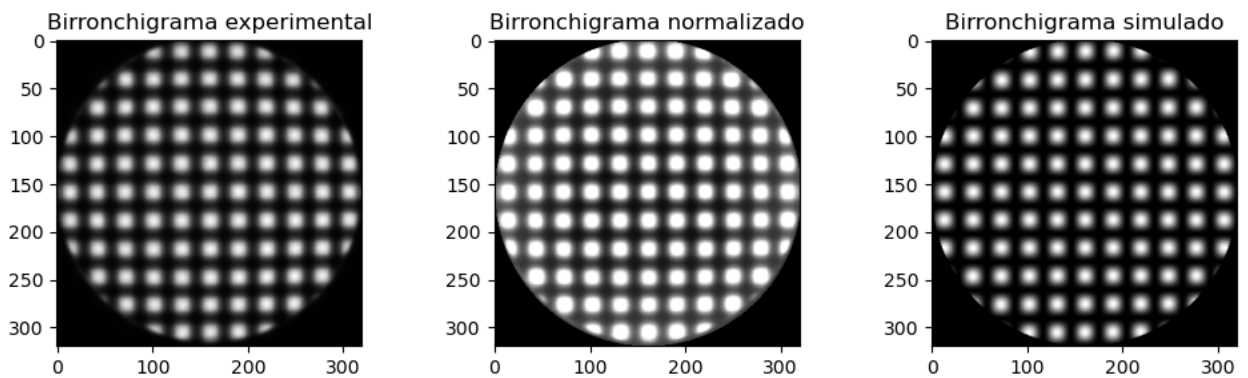


Figura 8.4: Resultados obtenidos al procesar el birronchigrama de un espejo de 14 cm de diámetro con radio de curvatura de 192.61 cm, distancia a la rejilla de 184.72 cm y constante de conicidad de $K = -1$ (Parab49R) con el software de evaluación de espejos.

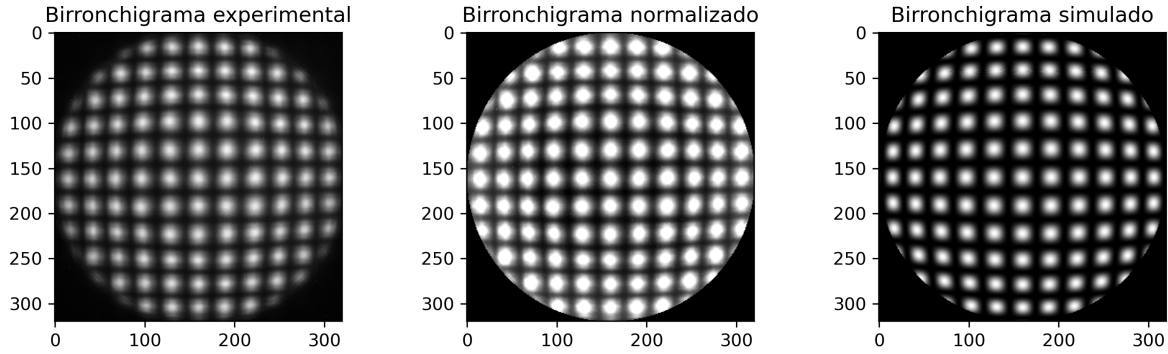


Figura 8.5: Resultados obtenidos al procesar un birronchigrama obtenido de un espejo parabólico de 14 cm de diámetro, con radio de curvatura de 192.09 cm y con una distancia de la rejilla a la superficie del espejo de 187.95 cm y constante de conicidad $K = 0$ (Parab_13R).

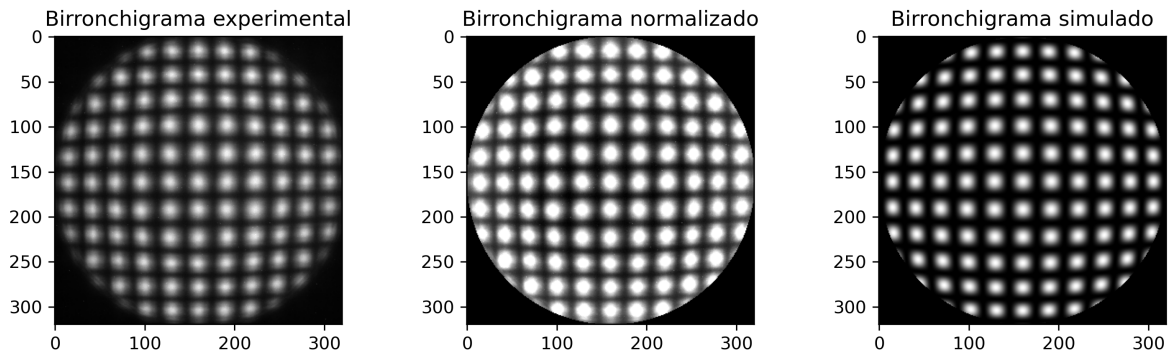


Figura 8.6: Resultados obtenidos al procesar un Birronchigrama obtenido de un espejo parabólico de 14 cm de diámetro, con radio de curvatura de 192.09 cm y con una distancia de la rejilla a la superficie del espejo de 187.95 cm y constante de conicidad $K = -1$ (Parab_13R).

8 pruebas, 4 con constante de conicidad $K = 0$ y las otras 4 con constante de conicidad $K = -1$, además de los coeficientes de correlación estimados, se encuentran en la tabla 8.8.

8.5. Birronchigrama 3

En esta prueba se ejecutó el programa con un birronchigrama obtenido del mismo banco de pruebas realizadas para la elaboración de espejos (parab_27R). Este espejo tiene un diámetro de 14 cm, un radio de curvatura de 192.46 y se realizó la prueba de Ronchi con una rejilla de Ronchi de 2 líneas por milímetro, con una distancia a la rejilla de 184.65 cm.

El resultado al ejecutar el software con los mismos parámetros del algoritmo híbrido, se obtuvieron al realizar de igual manera 8 ejecuciones, 4 utilizando la constante de conicidad $K = 0$ y 4 utilizando $K = -1$. Se puede apreciar en la Figura 8.7 la salida de la ejecución del programa con la constante de conicidad $K = 0$, mientras que en la Figura 8.8 se muestra la salida de la ejecución del software con la constante de conicidad $K = -1$. Los coeficientes de Zernike y el coeficiente de correlación obtenidos en estas pruebas se muestran en la tabla 8.9.

Tabla 8.8: Coeficientes obtenidos al ejecutar el software de evaluación de espejos con el birronchigrama 2. 4 ejecuciones fueron realizadas con el la constante $K = 0$ y en las otras 4 se utilizó el valor de $K = -1$. Al final de la tabla se muestra el coeficiente de correlación al evaluar cada grupo de coeficientes. Abreviaciones: AP = Astigmatismo primario, CP = Coma primaria, AS = Astigmatismo secundario, QFV = Quatrefoil vertical y QFO = Quatrefoil oblicuo.

Aberración	Coeficientes con $K = 0$				Coeficientes con $K = -1$			
Tilt y	-6.829	-6.5593	-6.5236	-6.9457	-7.7434	-5.5021	-8.5449	-7.5189
Tilt x	27.063	26.7957	28.2029	28.102	22.897	26.372	31.814	27.4379
Defoco	189.23	189.60	189.112	189.62	215.04	214.39	213.52	213.79
AP 45°	-56.854	30.737	26.339	-89.958	20.425	-6.7226	-178.02	-8.4915
AP 0°	8.9956	8.8259	8.3106	8.8475	8.3311	7.7052	9.3098	8.1803
CP x	0.4609	15.239	-82.858	52.022	245.85	56.7631	-323.23	32.6226
CP y	-0.3534	0.3635	-0.2002	-0.0457	0.5566	0.3468	0.2046	-0.3402
Trefoil y	28.037	36.345	69.617	41.95	53.572	96.0988	34.969	68.4319
Trefoil x	4.6162	2.6386	26.2679	17.88	17.061	118.29	19.72	4.5948
Esférica	-19.259	-19.572	-18.972	-19.532	-13.231	-13.2451	-12.62	-12.7364
AS 0°	0.064	41.923	53.456	-156.42	-0.0013	26.9759	-48.405	16.6796
AS 45°	-4.0103	-149.089	-22.796	-3.0264	-25.171	-81.1809	9.3705	-10.915
QFV	-11.933	-8.3446	5.8929	6.318	37.72	14.828	1.1469	0.2473
QFO	-27.682	-22.5911	-14.626	-5.4937	29.689	-28.8745	-29.459	-1.8988
Coef. corr.	0.925	0.9223	0.9236	0.9266	0.9185	0.9146	0.9229	0.9242

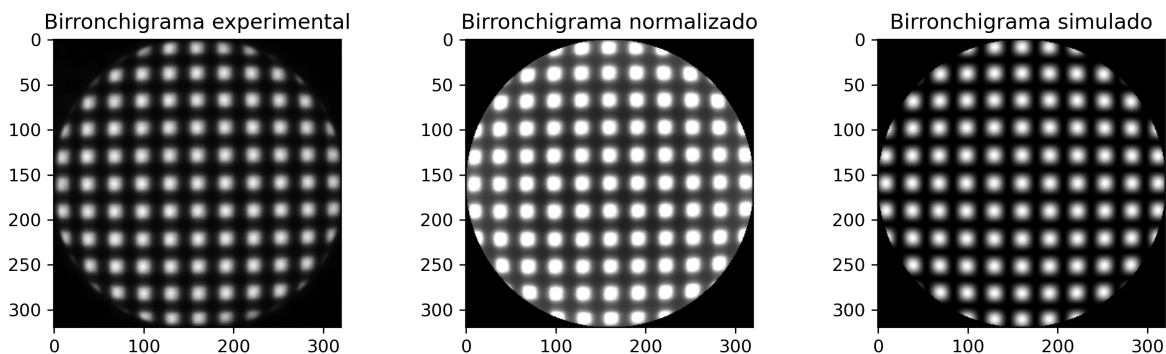


Figura 8.7: Resultados obtenidos al procesar un birronchigrama obtenido de un espejo parabólico de 14 cm de diámetro, con radio de curvatura de 192.46 cm, con una distancia de la rejilla a la superficie del espejo de 184.65 cm y constante de conicidad $K = 0$ (parab_27R).

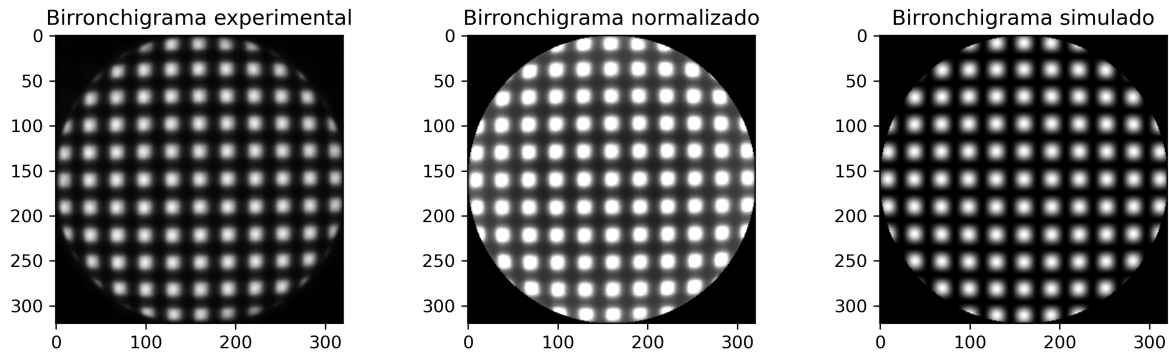


Figura 8.8: Resultados obtenidos al procesar un Birronchigrama obtenido de un espejo parabólico de 14 cm de diámetro, con radio de curvatura de 192.09 cm, con una distancia de la rejilla a la superficie del espejo de 187.95 cm y constante de conicidad $K = -1$ (parab.27R).

Tabla 8.9: Coeficientes obtenidos al ejecutar el software de evaluación de espejos con el birronchigrama 3. 4 ejecuciones fueron realizadas con el valor de $K = 0$ y en las otras 4 se utilizó $K = -1$. Al final de la tabla se muestra el coeficiente de correlación al evaluar cada grupo de coeficientes. Abreviaciones: AP = Astigmatismo primario, CP = Coma primaria, AS = Astigmatismo secundario, QFV = Quatrefoil vertical y QFO = Quatrefoil oblicuo.

Aberración	Coeficientes con $K = 0$				Coeficientes con $K = -1$			
Tilt y	5.9151	1.1426	1.4824	1.1703	1.6007	1.3884	0.9546	1.7697
Tilt x	4.2427	-11.444	-12.062	-12.77	-10.934	-12.248	-11.785	-12.115
Defoco	1682.4	1680.9	1680.7	1681.0	1705.2	1705.2	1705.2	1705.2
AP 45°	50.284	-1.9675	38.58	-6.4555	-12.003	7.4724	11.242	-1.8409
AP 0°	0.1583	1.1599	1.1713	1.2329	1.8737	1.0787	1.5091	1.0449
CP y	55.222	-22.581	-28.609	68.871	-63.935	53.496	-42.926	21.74
CP x	-3.6995	0.08453	-0.07794	0.04522	-0.0414	0.0805	0.0804	-2.75e-4
Trefoil y	-4.4411	-2.0344	0.8735	1.8828	-5.3152	-1.4959	-9.2541	-1.6191
Trefoil x	-41.889	8.9906	1.3082	3.2926	9.628	11.387	8.4546	6.071
Esférica	-8.808	-6.741	-6.7317	-6.772	-0.7822	-0.5422	-0.5651	-0.465
AS 0°	107.65	-27.757	31.023	-0.5347	2.3449	-104.63	0.4795	-63.383
AS 45°	-22.041	44.01	-5.0336	1.5671	3.5739	-1.7331	-2.4573	0.4332
QFV	-27.811	-2.691	-4.0389	-5.8212	4.2593	-2.1316	-3.1406	-1.7844
QFO	23.647	-41.153	-2.8912	-0.7236	1.7054	2.4548	2.0381	-0.236
Coef. corr.	0.9417	0.9571	0.9529	0.9533	0.9532	0.9536	0.9533	0.9536

Capítulo 9

Discusión de Resultados

Los resultados obtenidos a partir de la simulación con parámetros ideales de los birronchigramas, de un inicio mostraban un error de medición en los parámetros esenciales para la simulación, los cuales afectaron en primera instancia la visualización en comparación al birronchigrama obtenido de forma experimental que afecta los resultados.

Aún con este error, el algoritmo de optimización demostró ser capaz de encontrar coeficientes de aberración para reproducir el birronchigrama experimental.

Con respecto a los birronchigramas experimentales, el proceso de normalización utilizado ha demostrado ser eficiente, hasta con birronchigramas con poco nivel de luminosidad, además de tener una velocidad de ejecución relativamente alto, con un promedio de 2.21 segundos, lo cual no representa impacto considerable en el desempeño del software general.

En el proceso de optimización, se tenía contemplado únicamente usar algoritmos genéticos para este trabajo de tesis, pero al validarlo con el proceso de minimización, éste no ha demostrado ser la herramienta útil que encuentre una solución aceptable en un tiempo que sea considerado como en tiempo real. Fue por ello que se consideró la implementación de estrategias evolutivas, que han demostrado ser una alternativa que garantiza encontrar los valores óptimos en el menor tiempo, logrando que la estrategia evolutiva (1+1)-Cholesky-CMA sea aproximadamente 6 veces más veloz que el algoritmo genético en ejecutarse y mostrar los resultados publicados para encontrar los coeficientes de ajuste de la superficie de un molde para fabricar lentes elásticas [44]. Por lo anterior, la estrategia evolutiva también fue utilizada para el caso de maximización. Se esperaba un comportamiento similar que la primera aplicación, pero las funciones de aptitud son diferentes, en el caso de la correlación (maximización), requiere más tiempo de procesamiento debido a los costos computacionales necesarios para su simulación. Se observó que la estrategia evolutiva (1+1)-Cholesky-CMA generó resultados muy cercanos al óptimo obtenido con una programación anterior no eficiente. A pesar de ello, también se pudo observar que en ocasiones, la estrategia se estancaba en la búsqueda de la solución, por lo que nuevamente se implementó el algoritmo genético de valores reales, agregado a la estrategia, generando un software híbrido. Se notó que por sí solo, el algoritmo genético no es lo suficientemente rápido, pero garantiza encontrar un valor cercano al óptimo cuando la estrategia evolutiva no lo encuentra. Por lo tanto, el funcionamiento del algoritmo híbrido consiste en la búsqueda del valor óptimo trabajando inicialmente con la estrategia evolutiva (1+1)-Cholesky-CMA; en dado caso de que la estrategia agote el número de evaluaciones realizadas, o se estanque en algún valor no deseado, se procederá a generar una población inicial para el algoritmo genético de valores reales en base a la región de búsqueda cercana al mejor

individuo generado por la estrategia. Al final, el algoritmo híbrido mostrará la mejor solución encontrada, garantizando que ésta, es la más cercana al óptimo y mostrará en pantalla una imagen con el birronchigrama experimental, el normalizado y el simulado con los coeficientes de aberración optimizados. El algoritmo híbrido por lo tanto, garantiza 1): en una sola corrida encontrar una solución aceptable (solución muy cercana al óptimo), y 2): obtener esos resultados dentro del concepto de tiempo real.

La evolución en la búsqueda de la correlación entre el birronchigrama experimental y el generado no es una solución única, por lo que siempre habrá diferentes formas de la solución que puede simular la misma superficie del espejo.

Se observan en las tablas de las simulaciones de los birronchigramas ideales y experimentales un alto valor en la aberración de Defoco. Este alto valor podría ser generado debido a que la imagen obtenida del birronchigrama no se obtuvo correctamente durante el proceso experimental de la prueba de Ronchi, es decir, pudo haber errores en el montaje o la medición de parámetros necesarios para la simulación, como lo es por ejemplo, la distancia del espejo a la rejilla. Se presentaron otros valores altos de aberración no tan considerables como el Defoco en las tres tablas. Estos valores son los de Astigmatismo primario a 45° , Coma primaria en y y el valor de Astigmatismo secundario en 45° .

Por otro lado, en las optimizaciones realizadas con el birronchigrama experimental, se presenta un aumento en forma general en comparación a las optimización de los patrones ideales de todas las aberraciones. Sin embargo, en el caso del birronchigrama 1, se observa este cambio en Tilt x , Coma primaria en y y Trefoil en y . En las pruebas realizadas con el birronchigrama 2 se presenta un aumento en el Tilt en ambos ejes, Astigmatismo primario en 0° , Trefoil en ambos ejes, Quatrefoil vertical y oblicuo, pero disminuyendo el valor de Defoco en comparación a los datos entregados al ejecutar el algoritmo con la superficie ideal. Por último, en el birronchigrama 3 se presenta un aumento en los valores de las aberraciones de Tilt en ambos ejes, Coma y Trefoil en x , Astigmatismo a 0° , y ambos tipos de quatrefoil.

Por último, en todas las pruebas se realizó la comparación entre los valores de aberración esférica cuando se usan superficies de referencia esférica ($K=0$) y parabólica ($K=-1$), los cuales, se observó que el valor de esta aberración aumentó cuando se usó el valor $K = -1$, lo cual muestra que el algoritmo hace la comparación efectiva con las superficies de referencia correspondientes.

Capítulo 10

Conclusiones

Se desarrolló un software escrito en Python, el cual evalúa las deformaciones de la superficie de espejos por medio del concepto de aberraciones utilizando polinomios de Zernike. Este programa corresponde a la unificación de todo el trabajo realizado en una década de investigaciones respecto a la simulación, procesamiento y optimización de birronchigramas para espejos cóncavos con superficies cónicas, logrando un software eficiente que puede realizar sus procesos en lo que se conoce como en tiempo real.

El software mencionado muestra una imagen con 3 birronchigramas, el obtenido experimentalmente, el normalizado y el simulado. Además presenta los primeros 14 coeficientes de los polinomios de Zernike los cuales han sido optimizados y se emplean para la simulación, junto con los parámetros del birronchigrama experimental. Junto a estos resultados se incluyen el coeficiente de correlación y el tiempo de ejecución.

En la búsqueda del método de optimización más eficiente, se implementaron dos tipos de algoritmos evolutivos. Inicialmente se consideró el uso exclusivo de algoritmos genéticos. Sin embargo, durante las pruebas de validación en un problema de minimización del error para la aproximación de la superficie de un molde, el algoritmo genético no logró resultados satisfactorios. Esto llevó a la incorporación de la estrategia (1+1)-Cholesky-CMA, formando un algoritmo híbrido que no solo garantizó un tiempo de ejecución mínimo, sino que también mejoró la precisión en la búsqueda del óptimo, produciendo soluciones más cercanas al valor óptimo establecido.

Posteriormente, este algoritmo híbrido fue aplicado al problema de maximización de la correlación de birronchigramas. En este contexto, se observó que los distintos algoritmos empleados presentan comportamientos diferentes según el tipo de búsqueda y la función de aptitud o costo utilizada. Por lo tanto, el programa desarrollado permite la personalización de funcionalidades, ofreciendo la posibilidad de ajustar sus características en función de las necesidades específicas del usuario.

Se observó estabilidad en la mayoría de los casos de prueba, con algoritmos que convergen de manera consistente hacia soluciones óptimas. Las excepciones identificadas se analizaron para mejorar en un futuro la robustez del software. También se observaron cambios de los coeficientes de aberración respecto a las simulaciones ideales, los cuales muestran el aumento o disminución de aberraciones dentro de los birronchigramas experimentales, y así tomar nuevas consideraciones de pulido en el caso de la fabricación de los elementos reflectantes.

Las pruebas con espacios grandes de búsqueda han revelado una capacidad razonable para encontrar soluciones cercanas al óptimo. En el conjunto de gráficos 7.15 se muestra que el programa

logró encontrar un valor óptimo de correlación en rangos amplios. Sin embargo, se identificaron áreas de mejora para que el algoritmo pueda hacer búsquedas de rangos amplios de búsqueda, puesto que a partir de estos límites, el algoritmo presenta valores que no podrían considerarse como aceptables.

Trabajo a futuro

Dado que se tiene una biblioteca con código en Python. Es posible crear un API sobre un servidor para recibir peticiones en forma de archivos con datos e imágenes, y a partir de estos obtener como respuesta tablas con las pruebas realizadas o las imágenes de los birronchigramas normalizados y simulados.

Algunas de las tecnologías que pueden usarse para generar el API son: NodeJS y FastAPI. Así mismo es posible usar Flask y Django para consumir esta API y generar una plataforma Web que facilite a usuarios finales el uso de la biblioteca.

Al usar estas tecnologías, la aplicación puede ofrecerse como un servicio en la nube usando diferentes plataformas como son Amazon Web Service o Google Cloud, por citar algunas, permitiendo mayor poder de cómputo y que esté disponible para una mayor cantidad de usuarios.

Bibliografía

- [1] J. González-García, Y. Pérez-Maldonado, A. Santiago-Alvarado, A. Cordero-Dávila, G. Castro-González, M. E. Ramírez-Guzmán, and A. S. Cruz-Félix, “Robotic system of two degrees of freedom for the evaluation of concave reflective surfaces using the ronchi test,” *International Journal of Engineering Science and Innovative Technology, IJESIT*, vol. 5, no. 2, pp. 17–30, 2016.
- [2] A. Cordero-Dávila and J. González-García, “Optical surface evaluation by correlating bi-ronchigram images,” *Optical Engineering*, vol. 54, no. 3, Marzo, 2015.
- [3] Pasco, https://cdn2.pasco.com/product_document/Complete-Speed-of-Light-Apparatus-Manual-OS-9261B-OS-9262A-OS-9263B.pdf, *Speed of light apparatus*. Instruction manual and experiment guide for the Pasco Scientific Model OS-9261A, 62 and 63A.
- [4] L. Speacek, “Omnidirectional vision simulation and robot localisation,” *TAROS’06*, pp. 37–34, 2006.
- [5] L. Spacek, “Omnidirectional catadioptric vision sensor with conical mirrors,” *TIMR*, 2003.
- [6] J. C. A. Fernandes and J. A. B. C. Neves, “Using conical and spherical mirrors with conventional cameras for 360° panorama views in a single image,” *accepted for IEEE International Conference on Mechatronics*, july 3-5, 2006.
- [7] V. P. Radionov and V. K. Kisele, “Conical 90° mirrors for terahertz laser resonator,” *Telecommunications and Radio Engineering*, vol. 74, no. 4, pp. 337–343, 2015.
- [8] M. León-Rodríguez, J. A. Rayas, A. Martínez-García, A. Martínez-González, A. T.-Q. nones, and R. Porrás-Aguilar, “Panoramic reconstruction of quasi-cylindrical objects with digital holography and a conical mirror,” *Optics Letters*, vol. 46, 2021.
- [9] C. M. Ortiz-Lima, A. Cordero-Dávila, and J. González-García, “Small petal tools performance for parabolizing optical surfaces,” *Applied Optics*, vol. 55, no. 33, Noviembre 20, 2016.
- [10] D. Malacara, *Óptica básica*. Fondo de cultura económica, 3 ed., 2015.
- [11] E. Hecht, *Óptica*. ADDISON WESLEY IBEROAMERICANA, 2000.
- [12] R. E. Fishcer and B. Tadic-Galeb, *Optical System Design*. McGraw-Hill, 2000.
- [13] V. N. Mahajan, *Optical Imaging and Aberrations: Ray Geometrical Optics*. SPIE Press, 1998.

- [14] W. J. Smith, *Modern optical engineering: the design of optical systems*. McGraw-Hill Education, 3 ed., 2000.
- [15] D. Malacara and Z. Malacara, *Handbook of Optical Design*. Marcel Dekker, 2003.
- [16] J. Schwiegerling, *Optical specification, fabrication, and testing*. Society of Photo-Optical Instrumentation Engineers (SPIE), 2014.
- [17] D. Malacara, *Optical shop testing*, vol. 59. John Wiley & Sons, 3 ed., 2007.
- [18] L. Lundström and P. Unsbo, “Transformation of zernike coefficients: scaled, translated, and rotated wavefronts with circular and elliptical pupils,” *J. Opt. Soc. Am. A*, vol. 24, pp. 569–577, 2007.
- [19] V. M. Mahajan and D. Malacara, *Zernike Polynomials and Wavefront Fitting in Optical Shop Testing*. Wiley Sons, 3rd ed., 2007.
- [20] A. Cordero-Dávila, D. G. Reyes-Olguín, and J. González-García, “Border of spot diagrams are always caustic surface and/or marginal rays,” *Appl. Opt.*, vol. 61, no. C20-C26, 2022.
- [21] E. L. O’Neill, *Introduction to statistical optics*. Dover Publications Inc., 1992.
- [22] H. H. Hopkins, *Wave Theory of Aberrations*. Clarendon Press, 1950.
- [23] V. Lakshminarayanan and A. Fleck, “Zernike polynomials: a guide,” *Journal of Modern Optics*, 2011.
- [24] R. Sirohi, *Optical Methods of Measurement: Wholefield Techniques*. CRC Press, 2018.
- [25] J. M. Geary, *Introduction to Optical Testing*. The Society of Photo-Optical Instrumentation Engineers, 1993.
- [26] E. P. Goodwin and W. J. C., *Field Guide to Interferometric Optical Testing*, vol. FG10. SPIE Press, 2006.
- [27] R. R. Shannon and J. C. Wyant, *Applied Optics and Optical Engineering*, vol. 9. Academic Press, subsidiary of Harcourt Brace Janovich, Publishers, 1983.
- [28] R. L. Haupt and S. E. Haupt, *Practical Genetic Algorithms*. John Wiley & sons, Inc., 2004.
- [29] S. Mirjalili, *Evolutionary Algorithms and Neural Networks. Theory and Applications*. Springer International Publishing AG, 2019.
- [30] T. Bäck, J. N. Kok, and G. Rozenberg, *Handbook of natural computing*. Springer, Heidelberg, 2012.
- [31] A. Duarte, *Metaheuristics*. Dykinson, 2007.
- [32] O. Bozorg-Haddad, M. Solgi, and H. A. Loáiciga, *Meta-heuristic and evolutionary algorithms for Engineering optimization*. John Wiley & Sons, Inc., 2017.

- [33] E. Cantú-Paz, *Efficient and accurate parallel genetic algorithms*. Kluwer Academic Publishers, 2001.
- [34] S. Mirjalili, *Evolutionary Algorithms and Neural Networks. Theory and Applications*. Springer, 2019.
- [35] A. Q. H. Badar, *Evolutionary Optimization Algorithms*. CRC Press, 2022.
- [36] L. Araujo and C. Cervigón, *Algoritmos evolutivos: Un enfoque práctico*. Alfa Omega, 2009.
- [37] T. Bäck, C. Foussette, and P. Krause, *Contemporary Evolution Strategies*. Springer-Verlag Berlin Heidelberg, 2013.
- [38] G. van Rossum and the Python development team, *Python Frequently Asked Questions*. Python Software Foundation, 2023.
- [39] A. M. and Hugo van Kemenade and wiredfool, J. A. C. (Alex), A. Karpinsky, O. Baranovič, C. Gohlke, J. Dufresne, Yay295, DWesl, D. Schmidt, K. Kopachev, A. Houghton, S. Mani, S. Landey, J. Ware, vashek, Piolie, J. Douglas, S. T., D. Caro, U. Martinez, S. Kossouho, R. Lahd, A. Lee, E. W. Brown, O. Tonnhofer, and M. Bonfill, *python-pillow/Pillow: 10.0.0*. Zenodo, 2023.
- [40] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, “Array programming with NumPy,” *Nature*, vol. 585, pp. 357–362, Sept. 2020.
- [41] J. Hunter and M. Droettboom, *The Architecture of Open Source Applications (Volume 2) matplotlib*. <https://aosabook.org/en/v2/matplotlib.html>. Visitado el 26/08/23.
- [42] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, B. Matthew, J. Wilson, K. J. Millman, A. R. J. Mayorov, Nikolay Nelson, E. Jones, R. Kern, L. Eric, C. J. Carey, Í. Polat, Y. Feng, M. E. W., J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and SciPy 1. Contributors, “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python,” *Nature Methods*, vol. 17, pp. 261–272, 2020.
- [43] R. J. Salazar, C. R. Sánchez, C. M. Fabián, F. G. Sánchez, and L. A. Aguilar, “Generalized phase-shifting interferometry by parameter estimation with the least squares method,” *Optics and Lasers in Engineer*, 2013.
- [44] A. S. Alvarado, F. S. G. Agustín, M. E. P. Zacarias, O. H. Carranza, and V. I. M. Oliva, “Evaluation of aluminum molds using the null screens test,” in *Modeling Aspects in Optical Metrology IX* (B. Bodermann and K. Frenner, eds.), vol. 12619, p. 126190D, International Society for Optics and Photonics, SPIE, 2023.