



Universidad
Rey Juan Carlos

ESCUELA DE INGENIERÍA DE FUENLABRADA

GRADO EN INGENIERÍA AEROESPACIAL EN
VEHÍCULOS AEROESPACIALES

TRABAJO FIN DE GRADO

DISEÑO DE ALGORITMOS METAHEURÍSTICOS PARA EL
PROBLEMA DE COLOCACIÓN DE TURBINAS EÓLICAS
SOBRE EL TERRENO

Autor: Antonio José Romero Barrera

Tutor: Dr. David Casillas Pérez

Cotutora: Dra. Laura María Cornejo Bueno

Curso académico 2022/2023

Esta obra está bajo una licencia Creative Commons “Atribución-
NoComercial-SinDerivadas 4.0 Internacional”.



Agradecimientos

Tras muchos años de esfuerzo, me llena de orgullo poder cerrar mi ciclo universitario con este trabajo. Me gustaría reconocer a varias personas que me han apoyado durante toda esta trayectoria y que, sin ellas, este trabajo no hubiera podido ser posible.

A todos mis amigos de la universidad que me hicieron disfrutar de esta época, en especial a Juan Francisco Pablos García, Juan José González Reig, Carmen Martínez Beltrán, Irene Soler Antón, Raúl Serrano Rolando y Rubén Jiménez Mejías. . .

A mi compañera de grado, Verónica Vázquez Varela, por invitarme a ir con ella a Orti-gueira, donde me concienció de las problemáticas existentes con los parques eólicos. . .

A mi tutor, David Casillas Pérez, y mi cotutora, Laura Cornejo Bueno, por toda la ayuda y el apoyo dado durante el desarrollo de este trabajo. . .

A Gregorio Robles Martínez por toda la confianza depositada en mi persona y todo lo que me ha enseñado durante estos dos últimos años, además de por la plantilla de LaTeX usada en la memoria. . .

A mi compañero de despacho Sergio Raúl Montes León y mi compañera de trabajo Me-ritxell Díaz Coque, por asesorarme en varios aspectos de la memoria. . .

A todos los mencionados, mis más sinceros agradecimientos.

Finalmente, me gustaría dedicar un párrafo completo a mis padres, José Carmelo Romero Álvarez y María de los Milagros Barrera Esquinas, mi hermano, Héctor Romero Barrera, y mis cuatro abuelos, que me hubiera gustado que pudiesen presenciar este momento, pero sé que se sentirían orgullosos. Sin su aliento, dedicación y sacrificio, seguramente no me hubiera visto con fuerzas para continuar en los malos momentos. A todos ellos especialmente, les agradezco desde lo más profundo de mi corazón por todo lo que han hecho por mí.



PREMIADO COMO MEJOR TFG EN LOS II PREMIOS DE TRABAJOS FIN DE GRADO Y TRABAJOS FIN DE MÁSTER PARA LA TRANSICIÓN ENERGÉTICA

Este trabajo fue premiado como el mejor TFG para la Transición Energética, a nivel nacional, del curso 2022/2023. Este reconocimiento proviene de la Cátedra de Transición Energética, una iniciativa conjunta de la *Universidad de Santiago de Compostela (USC)* y *Genesal Energy*.

SOLICIT.	Título do traballo	Adec. ás liñas (ata 2 puntos)	Ponder ac. Cualific. (ata 3 puntos)	Calid. Académ. (ata 3 puntos)	interese e aplicab. Indust. (ata 2 puntos)	TOTAL
Romero Barrera, Antonio José	Diseño de algoritmos metaheurísticos para el problema de colocación de turbinas eólicas sobre el terreno	1.90	3.00	2.85	2.00	9.75

Comentario de autor: *Quiero expresar mi profundo agradecimiento a la Cátedra por su iniciativa y compromiso con el fomento del desarrollo de estudios relacionados con la generación de energías limpias y renovables, así como el impulso que aportan a la investigación y la innovación en este campo tan crucial para el futuro de nuestro planeta.*

Esta página fue adicionada posteriormente al 22 de mayo de 2023, día de la defensa de este TFG, en cumplimiento de las bases establecidas en la convocatoria de los “II Premio a Trabajos Fin de Grao e Traballos Fin de Máster para a Transición Enerxética”.

Resumen

Este trabajo tiene el propósito de estudiar el comportamiento de 3 algoritmos metaheurísticos basados en poblaciones (Genético Básico, PSO y CRO-SL) para la obtención de soluciones óptimas al problema de distribución de turbinas eólicas sobre superficies planas, teniendo en cuenta variables como el tamaño del terreno, el número total de turbinas a posicionar, el modelo de estela seleccionado para determinar la interrelación de afectación de las turbinas entre sí sobre sus corrientes incidentes, y el tipo de matriz de vientos escogida. En total, se comparan 7 modelos de estela distintos (Jensen, Park2, Bastankhah, Frandsen, Larsen, Ishihara, y Zhang) aplicando dos tipologías de vientos (sintéticos y reales), que determinarán la magnitud total de la energía generada en el plazo temporal establecido y la validez de las soluciones producidas por los algoritmos metaheurísticos para prevalecer a lo largo de las distintas iteraciones y ser tomadas como candidatas a la solución final del problema.

Se han utilizado para los análisis, extensiones discretizadas de terreno cuadradas de proporciones de 20×20 con 20 turbinas y de 50×50 con 50 turbinas, junto con dos tipos de vectores viento, de carácter real y generados sintéticamente, de tamaño 2×8760 , lo cual hace que el estudio abarque una totalidad de 84 experimentos. Para su realización se ha hecho uso del sistema de cómputo numérico MATLAB, el cual nos proporciona una gran ventaja y velocidad de cálculo a la hora de realizar operaciones matriciales.

Finalmente, se representa un cálculo cruzado para determinar qué modelo es más eficaz a la hora de realizar las optimizaciones con cada uno de los algoritmos y sus distintas configuraciones de terreno y vientos, realizando un cálculo de las combinaciones resultantes del producto cartesiano de cada una de las soluciones obtenidas con todos los modelos de estela planteados y haciendo una comparativa masiva de los resultados.

Summary

The purpose of this work is to study the performance of 3 population-based metaheuristic algorithms (BGA, PSO and CRO-SL) for obtaining optimal solutions to the problem of wind turbine distribution over flat surfaces, taking into account variables such as the size of the terrain, the total number of turbines to be positioned, the wake model selected to determine the interrelation of the turbines' effect on each other on their incident currents, and the type of wind matrix chosen. In total, 7 different wake models (Jensen, Park2, Bastankhah, Frandsen, Larsen, Ishihara, and Zhang) are compared applying two types of winds (synthetic and real), which will determine the total magnitude of the energy generated in the established time frame and the validity of the solutions produced by the metaheuristic algorithms to prevail throughout the different iterations and to be taken as candidates for the final solution of the problem.

For the analyses, discretized square plot extensions of 20×20 with 20 turbines and 50×50 with 50 turbines have been used, together with two types of wind vectors, real and synthetically generated, of size 2×8760 , which makes the study cover a total of 84 experiments. For its realization we have made use of the MATLAB numerical computation system, which provides us with a great advantage and speed of calculation when performing matrix operations.

Finally, a cross calculation is performed to determine which model is more efficient when performing the optimizations with each of the algorithms and their different terrain and wind configurations, calculating the resulting combinations of the cartesian product of each of the solutions obtained with all the wake models proposed and making a massive comparison of the results.

Índice general

1. Introducción	1
1.1. Motivación	5
1.2. Objetivos	6
1.3. Estructura de la memoria	7
2. Estado del arte	9
3. Marco Teórico	19
3.1. Algoritmos Metaheurísticos	19
3.1.1. Algoritmo Genético Básico (BGA)	20
3.1.2. Particle Swarm Optimization (PSO)	22
3.1.3. Algoritmo Coral Reefs Optimization (CRO)	24
3.2. Modelos de estela	28
3.2.1. Modelo de Jensen	29
3.2.2. Modelo Park2	30
3.2.3. Modelo de Frandsen	31
3.2.4. Modelo de Larsen	32
3.2.5. Modelo de Bastankhah	34
3.2.6. Modelo de Zhang	35
3.2.7. Modelo de Ishihara	36
3.3. La distribución del viento: Rosa de los vientos	37
3.4. Cálculo de afectación de las turbinas aguas abajo	39
4. Herramientas y desarrollo	43
4.1. Entorno de programación: MATLAB	43
4.2. Estructura de datos de modelos, algoritmos y terreno	45

4.2.1. Campos de la estructura generales	45
4.2.2. Campos de estructura específicos	46
4.3. Estructura de la función de <i>fitness</i>	47
4.4. Implementación de modelos de estela	50
4.5. Proceso de elaboración: Algoritmo Genético Básico	51
4.6. Proceso de elaboración: PSO	54
4.7. Proceso de adaptación: CRO-SL	56
4.8. Extracción de datos	58
5. Experimentos y validación	59
5.1. Hipótesis y simplificaciones	59
5.2. Datos usados para el cálculo experimental	60
5.3. Metodología de experimentos	62
5.3.1. Experimentos de optimización de posicionamiento de turbinas	62
5.3.2. Experimentos para la selección de mejores soluciones	62
5.4. Metodología de validación de procesos y resultados	64
5.4.1. Validación de tiempos de ejecución de los modelos de estela	64
5.4.2. Validación de las magnitudes de energía de las soluciones	65
6. Resultados y comparativas	69
6.1. Valores de energía generados y rendimiento	69
6.2. Evolución de las soluciones de la población	74
6.3. Mejores soluciones halladas para el problema	80
7. Conclusiones	85
7.1. Discusión final general	85
7.2. Impacto social, económico y medioambiental	87
7.3. Consecución de objetivos	90
7.4. Lecciones aprendidas	91
7.5. Trabajos futuros	91
Bibliografía	93
A. Glosario de variables	I

<i>ÍNDICE GENERAL</i>	IX
B. Glosario de acrónimos	III
C. <i>Scripts</i> utilizados para los experimentos	V
D. Valores orientativos de rugosidad del terreno	VII
E. Evolución de valores de la función de <i>fitness</i>	IX
F. Soluciones de posicionamiento de turbinas	XVII
G. Resultados de <i>fitness</i> de las soluciones cruzadas	XXXI
H. Resultados de rendimiento de las soluciones cruzadas	XXXVII

Índice de figuras

1.1. Partes principales de un aerogenerador.	2
1.2. Granja eólica marítima Horns Rev en Dinamarca.	4
2.1. Frecuencia de uso de <i>keywords</i> procedentes del <i>snowballing</i>	10
2.2. Los 10 autores con más publicaciones entre los artículos filtrados.	13
2.3. Relación entre autores de los artículos seleccionados.	14
2.4. Frecuencia de uso de <i>keywords</i> tras el proceso SLR.	14
2.5. Evolución del número de publicaciones en los últimos 10 años.	15
3.1. Tipos de algoritmos metaheurísticos.	20
3.2. Diagrama de funcionamiento de algoritmo genético básico.	22
3.3. Diagrama de funcionamiento del algoritmo CRO.	26
3.4. Esquema del efecto deficitario interno de la estela sobre la velocidad.	28
3.5. Representaciones genéricas de rosa de los vientos.	37
3.6. Representaciones alternativas de rosa de los vientos.	39
4.1. Comprobación de afectación de estela durante la fase de obtención de distancias de la función de <i>fitness</i>	49
4.2. Tipos de geometrías de estelas.	51
4.3. Representación gráfica del proceso de cruce 1Px.	52
4.4. División de los individuos de la población en grupos de partículas.	54
5.1. Curvas de potencia aerogenerador AWEC-60.	61
5.2. Producto cartesiano de conjuntos {WM} con {S} para obtención de soluciones cruzadas.	63
6.1. Resultados Algoritmo Genético Básico para terreno de 20×20 con viento real. .	75

6.2. Resultados Algoritmo Genético Básico para terreno de 50×50 con viento real. . .	76
6.3. Resultados <i>Particle Swarm Optimization</i> para terreno de 20×20 con viento real. . .	77
6.4. Resultados <i>Particle Swarm Optimization</i> para terreno de 50×50 con viento real. . .	78
6.5. Resultados CRO-SL para terreno de 20×20 con viento real.	79
6.6. Resultados CRO-SL para terreno de 50×50 con viento real.	80
6.7. Mapa posicional de las soluciones cruzadas con mejor rendimiento medio global. . .	83
E.1. Resultados Algoritmo Genético Básico para terreno de 20×20 con viento sintético.	X
E.2. Resultados Algoritmo Genético Básico para terreno de 50×50 con viento sintético.	XI
E.3. Resultados <i>Particle Swarm Optimization</i> para terreno de 20×20 con viento sintético.	XII
E.4. Resultados <i>Particle Swarm Optimization</i> para terreno de 50×50 con viento sintético.	XIII
E.5. Resultados CRO-SL para terreno de 20×20 con viento sintético.	XIV
E.6. Resultados CRO-SL para terreno de 50×50 con viento sintético.	XV
F.1. Resultados Algoritmo Genético Básico para terreno de 20×20 con viento sintético.	XVIII
F.2. Resultados Algoritmo Genético Básico para terreno de 20×20 con viento real. . .	XIX
F.3. Resultados Algoritmo Genético Básico para terreno de 50×50 con viento sintético.	XX
F.4. Resultados Algoritmo Genético Básico para terreno de 50×50 con viento real. . .	XXI
F.5. Resultados Algoritmo PSO para terreno de 20×20 con viento sintético.	XXII
F.6. Resultados Algoritmo PSO para terreno de 20×20 con viento real.	XXIII
F.7. Resultados Algoritmo PSO para terreno de 50×50 con viento sintético.	XXIV
F.8. Resultados Algoritmo PSO para terreno de 50×50 con viento real.	XXV
F.9. Resultados Algoritmo CRO-SL para terreno de 20×20 con viento sintético. . .	XXVI
F.10. Resultados Algoritmo CRO-SL para terreno de 20×20 con viento real.	XXVII
F.11. Resultados Algoritmo CRO-SL para terreno de 50×50 con viento sintético. . .	XXVIII
F.12. Resultados Algoritmo CRO-SL para terreno de 50×50 con viento real.	XXIX

Índice de tablas

1.1. Objetivos generales y específicos del proyecto.	6
1.2. Estructura del contenido principal de la memoria.	7
1.3. Estructura del contenido de los apéndices.	8
2.1. Protocolo SLR para el filtrado de artículos.	12
3.1. Valores empíricos de los coeficientes del modelo Larsen.	33
3.2. Métodos de superposición de estelas	40
3.3. Principios de aproximación de estela por superposición.	41
4.1. Comparativa de características entre MATLAB y Python (NumPy) para imple- mentar algoritmos metaheurísticos con matrices.	45
4.2. Estructura y variables de la función de <i>fitness</i>	48
4.3. Campos característicos de ‘data’ para el uso de Algoritmo Genético Básico. . .	53
4.4. Campos característicos de ‘data’ para el uso de algoritmo PSO.	55
4.5. Campos característicos de ‘cn’ para el uso de algoritmo CRO-SL.	57
4.6. Campos característicos de ‘f’ para el uso de algoritmo CRO-SL.	57
4.7. Campos característicos de ‘p’ para el uso de algoritmo CRO-SL.	58
5.1. Parámetros de generación de los vectores velocidad sintéticos.	60
5.2. Características del entorno y geométricas de las turbinas.	61
5.3. Tiempos de ejecución de los modelos de estela y proporciones del terreno. . . .	65
5.4. Energía máxima y mínima generadas anualmente por 20 turbinas eólicas. . . .	66
5.5. Energía máxima y mínima generadas anualmente por 50 turbinas eólicas. . . .	67
6.1. Resultados BGA con distribución de vientos sintética.	71
6.2. Resultados BGA con distribución de vientos real.	71

6.3. Resultados PSO con distribución de vientos sintética.	72
6.4. Resultados PSO con distribución de vientos real.	72
6.5. Resultados CRO-SL con distribución de vientos sintética.	73
6.6. Resultados CRO-SL con distribución de vientos real.	74
6.7. Mejores rendimientos medios de modelo de estela para distribución de vientos sintéticos.	81
6.8. Mejores rendimientos medios de modelo de estela para distribución de vientos real.	82
6.9. Soluciones cruzadas con mejores rendimientos medios.	84
7.1. Metas específicas de los ODS que se adecuan a este estudio.	89
D.1. Valores de rugosidad para distintas tipologías de terreno.	VIII
G.1. Soluciones cruzadas de <i>fitness</i> de BGA con configuración de 20×20 y viento sintético.	XXXI
G.2. Soluciones cruzadas de <i>fitness</i> de BGA con configuración de 20×20 y viento real.	XXXI
G.3. Soluciones cruzadas de <i>fitness</i> de BGA con configuración de 50×50 y viento sintético.	XXXII
G.4. Soluciones cruzadas de <i>fitness</i> de BGA con configuración de 50×50 y viento real.	XXXII
G.5. Soluciones cruzadas de <i>fitness</i> de PSO con configuración de 20×20 y viento sintético.	XXXII
G.6. Soluciones cruzadas de <i>fitness</i> de PSO con configuración de 20×20 y viento real.	XXXIII
G.7. Soluciones cruzadas de <i>fitness</i> de PSO con configuración de 50×50 y viento sintético.	XXXIII
G.8. Soluciones cruzadas de <i>fitness</i> de PSO con configuración de 50×50 y viento real.	XXXIII
G.9. Soluciones cruzadas de <i>fitness</i> de CRO-SL con configuración de 20×20 y viento sintético.	XXXIV
G.10. Soluciones cruzadas de <i>fitness</i> de CRO-SL con configuración de 20×20 y viento real.	XXXIV
G.11. Soluciones cruzadas de <i>fitness</i> de CRO-SL con configuración de 50×50 y viento sintético.	XXXIV
G.12. Soluciones cruzadas de <i>fitness</i> de CRO-SL con configuración de 50×50 y viento real.	XXXV

H.1. Rendimiento cruzado de BGA con configuración de 20×20 y viento sintético.	XXXVII
H.2. Rendimiento cruzado de BGA con configuración de 20×20 y viento real.	XXXVII
H.3. Rendimiento cruzado de BGA con configuración de 50×50 y viento sintético.	XXXVIII
H.4. Rendimiento cruzado de BGA con configuración de 50×50 y viento real.	XXXVIII
H.5. Rendimiento cruzado de PSO con configuración de 20×20 y viento sintético.	XXXVIII
H.6. Rendimiento cruzado de PSO con configuración de 20×20 y viento real.	XXXIX
H.7. Rendimiento cruzado de PSO con configuración de 50×50 y viento sintético.	XXXIX
H.8. Rendimiento cruzado de PSO con configuración de 50×50 y viento real.	XXXIX
H.9. Rendimiento cruzado de CRO-SL con configuración de 20×20 y viento sintético.	XL
H.10. Rendimiento cruzado de CRO-SL con configuración de 20×20 y viento real.	XL
H.11. Rendimiento cruzado de CRO-SL con configuración de 50×50 y viento sintético.	XL
H.12. Rendimiento cruzado de CRO-SL con configuración de 50×50 y viento real.	XLI

Capítulo 1

Introducción

En los últimos años, la importancia de las energías renovables ha crecido exponencialmente, representando un 12.6 % de la energía total consumida en el año 2020 [1]. Una de las formas más populares de obtener energía renovable es mediante el uso de turbinas eólicas, las cuales se posicionan estratégicamente en áreas con vientos fuertes para extraer energía del viento.

La extracción de energía a través del mecanismo cinético descrito depende, principalmente, del módulo efectivo de la velocidad del viento que incide sobre la turbina. Un módulo efectivo del vector viento mayor, provocará que incremente la tracción de las palas de forma cuadrática y la velocidad de rotación del rotor, lo que se traduce en un aumento de la energía generada. El módulo del vector vientos puede verse afectado por diversos factores que alteren la trayectoria del mismo y produzcan una desaceleración del flujo, como pueden ser elementos y relieves del paisaje, o las propias estelas producidas por la rotación de las palas de las turbinas eólicas. Los componentes que forman el sistema de transformación de energía de la turbina eólica se pueden observar en la Figura 1.1.

- **Torre:** es la estructura vertical que soporta todo el conjunto del aerogenerador y lo eleva por encima del nivel del suelo para aprovechar los vientos con mayor velocidad y energía cinética.
- **Góndola:** protección encargada de almacenar todos los componentes mecánicos y electrónicos de la turbina eólica, encargados de la transformación de energía y las mediciones.
- **Rotor:** elemento rotatorio del aerogenerador que aloja las palas y su eje de giro. La rotación del rotor genera energía mecánica que se transmite a través del eje hacia el generador. Está formado por el buje y las palas.

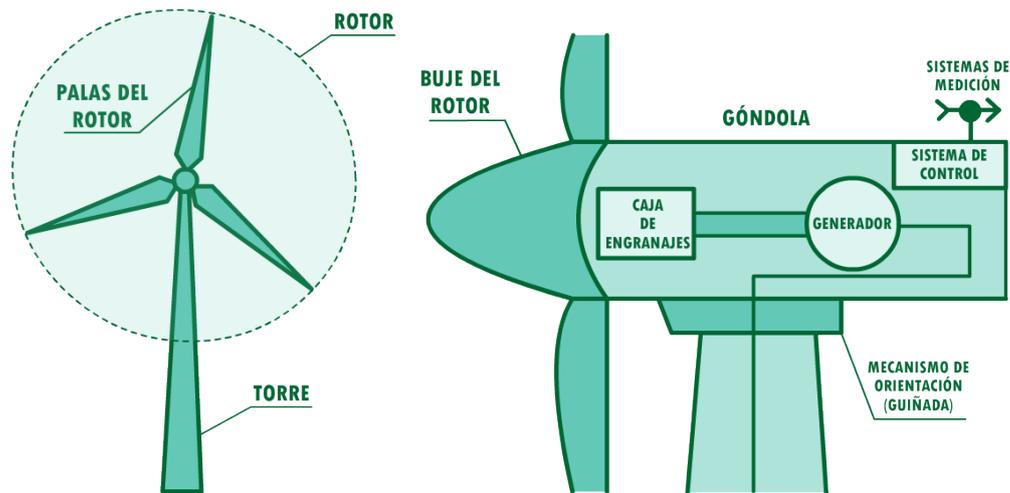


Figura 1.1: Partes principales de un aerogenerador.

- **Palas:** son los elementos aerodinámicos encargados de capturar la energía cinética del viento y transformarla en energía mecánica mediante su rotación.
- **Buje:** componente situado en el centro del rotor, que conecta las palas al eje de transmisión que gira el generador eléctrico.
- **Generador:** dispositivo electromagnético encargado de transformar la energía mecánica en energía eléctrica. El generador está compuesto por un conjunto de imanes y bobinas que generan corriente eléctrica mediante el movimiento relativo entre ellos. Transforma la energía mecánica generada por el movimiento de las palas en energía eléctrica mediante el principio de inducción electromagnética. Básicamente, cuando el eje rotor gira, se desplaza a través de un campo magnético, lo que induce la generación de una corriente eléctrica en el cableado del generador y, por ende, la producción de energía eléctrica.
- **Sistema de control:** es el conjunto de dispositivos electrónicos que monitorizan y regulan el funcionamiento del aerogenerador, ajustando su posición y velocidad para maximizar la producción de energía y garantizar la seguridad y estabilidad del sistema.
- **Sistemas de medición:** compuesto por elementos encargados de medir propiedades del viento, como los anemómetros y las veletas, y transmitir dicha información al sistema de control.
- **Caja de engranajes:** permite adaptar la velocidad de rotación de las palas al generador, lo que permite una operación eficiente y segura de la turbina eólica. Esto se logra gracias a la

combinación interna de engranajes que aumentan la velocidad de rotación de las palas y la adaptan a la velocidad de rotación de operación establecida del generador, protegiéndolo de sobrecargas.

En lo referente al número de palas de los aerogeneradores, actualmente se sigue el diseño establecido por el modelo danés de 3 palas. Un número mayor de palas implicaría la reducción del tamaño de la cuerda de los perfiles alares y el uso de materiales más ligeros y resistentes en la fabricación de la pieza para mantener la integridad estructural y no aumentar el peso del conjunto, lo que se traduce en un aumento notable de los costes. Un número menor, haría que surgieran problemas de estabilidad en el modelo de 1 pala por la falta de contrapeso y simetría de fuerzas, y problemas de vibraciones en el modelo de 2 palas por el aumento que se tendría que hacer en la longitud y cuerda, para que el rotor ejerza la tracción correspondiente. El modelo de 3 palas es, en definitiva, la mejor opción de diseño debido a su estabilidad, eficiencia y relación de costes.

Una vez seleccionada la región para la granja eólica, es importante distribuir las turbinas de manera óptima para maximizar la producción de energía a largo plazo. En este contexto, el uso de metodologías basadas en nuevas tecnologías para el estudio del posicionamiento de turbinas eólicas se ha incrementado y se han desarrollado técnicas para encontrar soluciones óptimas. Se busca realizar un estudio del posicionamiento de turbinas utilizando algoritmos evolutivos, específicamente los genéticos, los cuales son ampliamente utilizados en este campo. Existen múltiples combinaciones entre los algoritmos evolutivos y los modelos de turbulencia aplicados en el estudio para modelar el déficit de velocidad que se genera aguas abajo en la granja, afectando cada elemento de la misma. En la Figura 1.2 se muestra una representación de las estelas de las turbinas eólicas de granja eólica de Horns Rev [2].

En la actualidad existen 2 tipos principales de parques eólicos:

- **Parque eólico terrestre (*onshore*):** enfocados en la generación de electricidad aprovechando la energía cinética del viento capturada por medio de turbinas eólicas instaladas sobre tierra. La conversión de la energía cinética en energía eléctrica se lleva a cabo mediante un proceso de transformación electromecánica que se integra en la red eléctrica.

- **Parque eólico marítimo (*offshore*):** aprovechan la energía cinética del viento en alta mar mediante la instalación de turbinas eólicas en plataformas adaptadas, ancladas sobre el lecho marítimo a través de estructuras especiales de grandes dimensiones.



Figura 1.2: Granja eólica marítima Horns Rev en Dinamarca. Adaptada de [2].

En este trabajo se estudiará el uso de los algoritmos metaheurísticos poblacionales para el posicionamiento óptimo de turbinas eólicas en una granja eólica. Estos algoritmos son un tipo de metaheurística que hacen uso de procesos observados en la naturaleza para buscar soluciones óptimas a un problema. A través de la selección de los individuos más aptos y la generación de otros estados nuevos, estos algoritmos pueden encontrar soluciones óptimas de manera eficiente.

La elección de un algoritmo de optimización adecuado es fundamental para el éxito del estudio, y se considerarán varios de ellos para encontrar el más adecuado para la resolución del problema planteado. Además, se utilizarán varios modelos de estela para simular la evolución de la velocidad del viento sobre la granja eólica, ya que puede haber variaciones relevantes en los resultados de la energía generada por los aerogeneradores en función del modelo escogido, tal y como se menciona en [3].

El estudio de posicionamiento de turbinas eólicas utilizando algoritmos metaheurísticos tiene un gran potencial en el campo de las energías renovables. La capacidad de encontrar soluciones óptimas a un problema complejo como la colocación de turbinas eólicas puede tener un impacto significativo en la producción de energía renovable y, por ende, en la lucha contra el cambio climático.

1.1. Motivación

El cambio climático es uno de los mayores desafíos que enfrenta la humanidad en la actualidad, afectando a los distintos ecosistemas, su biodiversidad y varios sectores económicos como la ganadería o la agricultura [4]. La necesidad de reducir nuestras emisiones de gases de efecto invernadero es cada vez más urgente. En este contexto, la energía renovable, y en particular la energía eólica, juega un papel fundamental en la transición hacia un futuro más sostenible [5], por lo que, instituciones como la Organización de Naciones Unidas (ONU) o la Unión Europea (UE) han marcado respectivamente una serie de objetivos como la Coalición Global por la Neutralidad del Carbono [6] y el Horizonte Europa [7] para reducir las emisiones de efecto invernadero.

Como se menciona en [8], el posicionamiento óptimo de turbinas eólicas sobre el terreno es un problema complejo que puede tener un gran impacto en la producción de energía renovable. La capacidad de encontrar la solución óptima a este problema puede aumentar significativamente la eficiencia y la rentabilidad de la propia granja eólica, lo que se traduce en una mayor producción de energía no contaminante y, por ende, una reducción de las emisiones de gases de efecto invernadero.

Con este estudio se busca contribuir a este importante objetivo, mediante la aplicación de técnicas de Inteligencia Artificial (IA) para el posicionamiento óptimo de turbinas eólicas. Al utilizar algoritmos metaheurísticos y modelos de estela, se busca encontrar soluciones óptimas para el posicionamiento de turbinas eólicas en una granja eólica, lo que puede tener un impacto significativo en la producción de energía renovable.

Este TFG tiene el potencial de contribuir significativamente a la lucha contra el cambio climático, y de ayudar a promover la transición hacia un futuro más sostenible y respetuoso con el medio ambiente. Por lo tanto, la motivación principal es trabajar en la búsqueda de soluciones innovadoras y eficaces para los desafíos energéticos que enfrentamos, y así contribuir a un futuro más sostenible y habitable para todos.

1.2. Objetivos

Este Trabajo Fin de Grado busca encontrar una metodología óptima de posicionamiento de aerogeneradores, sobre distintas tipologías de terreno discretas, a través de 3 algoritmos meta-heurísticos basados en poblaciones, y estudiar la variación de resultados entre ellos en función del modelo de estela utilizado para los cálculos de la función de *fitness* en los algoritmos, con el fin de determinar cuál de ellos resulta más efectivo para llevar a cabo la optimización. Otro punto importante del estudio es precisar qué modelo de estela funciona mejor a la hora de obtener una solución óptima en cada uno de los algoritmos planteados. En la Tabla 1.1 se recogen los objetivos generales y específicos establecidos para este trabajo.

Tabla 1.1: Objetivos generales y específicos del proyecto.

Objetivos Generales (OG)	Objetivos Específicos (OE)
OG1: Creación de un repositorio y documentación de varios modelos de estela usados en el sector de las turbinas eólicas.	OE1.1: Agrupar la información de varios modelos de estela encontrados en la literatura científica y resumir su funcionamiento y las ecuaciones por las que se rigen.
	OE1.2: Creación de <i>scripts</i> de MATLAB que permitan ejecutar fácilmente las ecuaciones de los modelos.
OG2: Realizar una comparativa entre el funcionamiento y los resultados obtenidos de los 3 algoritmos metaheurísticos utilizados.	OE2.1: Programación de un Algoritmo Genético Básico y un algoritmo Particle Swarm Optimization.
	OE2.2: Adaptación de un algoritmo CRO-SL al problema planteado y a la función de <i>fitness</i> programada para la obtención de resultados óptimos.
	OE2.3: Generación gráficas de evolución de las soluciones y de tablas con los datos de energía generada y rendimiento.
OG3: Obtener la mejor colocación de turbinas para cada configuración de turbinas, terreno y vientos planteados.	OE3.1: Cálculo de las soluciones cruzadas, entre todos los modelos de estela, de las soluciones óptimas del proceso de optimización.
	OE3.2: Determinar el modelo de estela preferente a la hora de realizar el proceso de optimización con cada uno de los algoritmos planteados y las distintas configuraciones.

1.3. Estructura de la memoria

En este TFG, se ha estructurado el contenido principal en un total de 7 capítulos que han sido descritos detalladamente en la Tabla 1.2. Adicionalmente, se puede consultar la bibliografía y un total de 8 apéndices que complementan la información y resultados mostrados en la memoria, cuya explicación se encuentra recogida en la Tabla 1.3.

Tabla 1.2: Estructura del contenido principal de la memoria.

Título	Detalles
Capítulo 1: Introducción	Explicación breve de la temática del proyecto, así como de las motivaciones e inquietudes que llevaron a la selección del mismo. También se exponen los objetivos generales y específicos que se buscan completar, y la estructura de la propia memoria.
Capítulo 2: Estado del Arte	Estudio sobre el estado actual de las investigaciones científicas llevadas a cabo desde 2013 hasta la actualidad. Se realiza un análisis de la literatura de una base de datos de artículos extraídos de Google Scholar y Scopus para determinar los focos de desarrollo y el crecimiento de publicaciones a lo largo de los últimos años.
Capítulo 3: Marco Teórico	Definición de los conceptos básicos necesarios para comprender la totalidad del proyecto. Se muestra el funcionamiento teórico de los 3 algoritmos metaheurísticos usados, los 7 modelos de estela incorporados, el concepto de rosa de los vientos y los tipos de superposición de estelas.
Capítulo 4: Desarrollo	Sección que recoge la descripción del desarrollo de los <i>scripts</i> de MATLAB usados para llevar a cabo los experimentos, y de las variables y parámetros necesarios para su funcionamiento.
Capítulo 5: Experimentos y validación	Definición de los distintos experimentos que se van a ejecutar para poder realizar posteriormente las comparativas. Se exponen los datos usados para los cálculos y las configuraciones seleccionadas. También se hallan una serie de valores que sirven como umbrales de validación de energía generada por los distintos modelos.
Capítulo 6: Resultados y comparativas	Debate de los resultados obtenidos, el comportamiento y tendencia de cada uno de los algoritmos usados, diferencias entre sus valores de energía generada y extracción de la mejor solución global al problema.
Capítulo 7: Conclusiones	Visión y planteamientos finales tras la realización de los experimentos, así como la ponderación del cumplimiento de los objetivos planteados al inicio y las posibles vías futuras de investigación relacionadas.
Bibliografía	Recopilación de fuentes consultadas para el desarrollo de la memoria.

Tabla 1.3: Estructura del contenido de los apéndices.

Título	Detalles
Apéndice A: Glosario de variables	Recopilación del nombre de las variables utilizadas en las ecuaciones que aparecen en la memoria.
Apéndice B: Glosario de acrónimos	Recopilación de todos los acrónimos que aparecen en la memoria, junto con el significado de sus siglas.
Apéndice C: <i>Scripts</i> utilizados para los experimentos	Contiene el enlace directo al repositorio de GitHub. En él se pueden consultar varios de los <i>scripts</i> de MATLAB que se han desarrollado para llevar a cabo los experimentos necesarios.
Apéndice D: Valores orientativos de rugosidad del terreno	Tabla que recoge los valores aproximados de rugosidad del terreno, que se le asignan a la variable z_0 , utilizada en algunos modelos de estela para determinar la pendiente del cono de expansión en función del terreno donde se encuentre situada la turbina eólica.
Apéndice E. Evolución de valores de la función <i>fitness</i>	Gráficas que muestran la tendencia de crecimiento de los valores de la función <i>fitness</i> del mejor individuo y de la media poblacional conforme avanza el número de iteraciones del algoritmo.
Apéndice F: Soluciones de posicionamiento de turbinas	Representación gráfica del mapa de posiciones de turbinas sobre el terreno discretizado, en función de sus proporciones, número de turbinas, tipo de viento, algoritmo metaheurístico y modelo de estela.
Apéndice G: Resultados de <i>fitness</i> de las soluciones cruzadas	En este apéndice se muestran todas las tablas de las soluciones cruzadas entre modelos de estela, con sus valores correspondientes de <i>fitness</i> .
Apéndice H: Resultados de rendimiento de las soluciones cruzadas	El apéndice recoge todas las tablas de rendimiento de las soluciones cruzadas entre modelos de estela, junto con los rendimientos medios por solución y modelo de estela.

Capítulo 2

Estado del arte

Los algoritmos metaheurísticos son una herramienta computacional de optimización que se inspira en una serie de procesos y comportamientos observados en la naturaleza para encontrar soluciones óptimas a problemas complejos por medio de iteraciones sucesivas. En el caso de la ubicación de turbinas eólicas, el objetivo es encontrar una distribución que maximice la producción de energía del conjunto (función de *fitness*) y minimizar el impacto ambiental y social que puedan ocasionar.

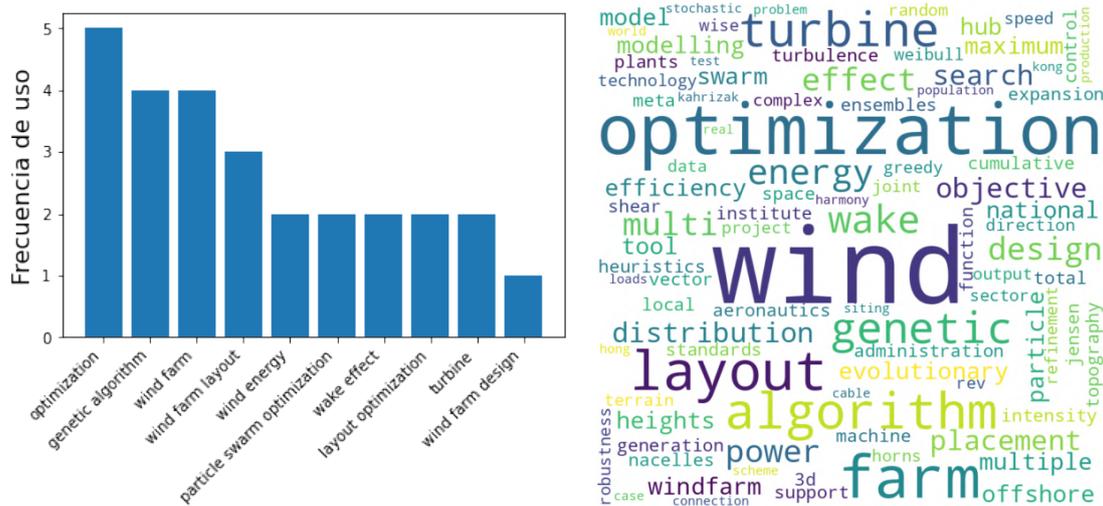
El uso de algoritmos metaheurísticos en este contexto permite explorar una gran cantidad de posibles ubicaciones de turbinas en un tiempo razonable, lo que sería imposible mediante métodos tradicionales. Además, estos algoritmos tienen la capacidad de aprender y adaptarse a medida que se utilizan, mejorando la eficiencia y efectividad del proceso de búsqueda.

En relación con la temática del problema, se ha observado un incremento de la literatura científica en los últimos años, sobre todo debido al gran desarrollo de los distintos campos de la Inteligencia Artificial (IA) y al auge de las energías renovables, las cuales están ganando importancia y cuota de generación de energía con respecto al resto de tecnologías. Se han propuesto diferentes enfoques y metodologías para el uso de algoritmos metaheurísticos en la ubicación de turbinas eólicas. Por ejemplo, algunos estudios se han centrado en la optimización de la disposición de turbinas en parques eólicos teniendo en cuenta la afectación y superposición de las distintas estelas (es el caso de este TFG), mientras que otros se han centrado en la ubicación óptima de turbinas individuales.

Múltiples estudios han demostrado que el uso de algoritmos metaheurísticos son un método eficaz para generar soluciones al problema de posicionamiento de turbinas eólicas. Los resultados de mejora en el uso de algoritmos metaheurísticos con respecto a distribuciones aleato-

rias de la granja de turbinas han llegado a marcar diferencias muy notorias en la generación de energía. Otros autores han hecho uso de esta aleatoriedad para realizar optimizaciones por metaheurísticas poblacionales que llevaban a cabo una generación de soluciones (individuos) pseudoaleatorias, que eran posteriormente sometidas a procesos de selección [9]. Para este trabajo, se ha partido del artículo [10] como referencia para iniciar la metodología de análisis de la literatura. El artículo está focalizado en la optimización de la distribución de aerogeneradores en parques eólicos a través del algoritmo CRO-SL, el cual es comparado, haciendo uso del modelo Jensen, con un algoritmo *pseudorandom* que genera soluciones aleatorios para generar los individuos de la población, y con las soluciones obtenidas a través de otros algoritmos de optimización que compitieron en el *challenge* [11] organizado por el *National Renewable Energy Lab* (NREL) estadounidense junto con la *International Energy Agency* (IEA). La combinación de diferentes métodos de búsqueda en un solo algoritmo y la utilización de una codificación discreta y un procedimiento geométrico convexo son aspectos clave para el buen desempeño del algoritmo CRO-SL.

Se ha realizado una búsqueda de artículos científicos a través de las bases de datos de Google Scholar y Scopus a través del uso condicional de varios de los *keywords* obtenidos del proceso de *snowballing* realizado sobre el artículo [10]. Buscamos artículos centrados en la optimización del posicionamiento de turbinas sobre el terreno, por lo que definimos una serie de *keywords* que guarden relación con esta temática y que son indispensables que contengan los artículos de la búsqueda.



(a) Top 10 keywords más usadas.

(b) Wordcloud de keywords.

Figura 2.1: Frecuencia de uso de *keywords* procedentes del *snowballing*.

Se busca que los resultados de la búsqueda guarden relación con metodologías basadas en IA para llevar a cabo el proceso de optimización, a la vez que sean destinados para distribuir turbinas eólicas, por lo que se introducirán dos condicionales con varios ‘OR’ entre los *keywords* que hacen referencia a términos de estas temáticas, y se fijan las *keywords* ‘*optimization*’ y ‘*wind turbines*’ entre el operador condicional AND, cuyo uso fue bastante frecuente entre los artículos del *snowballing*. La regla introducida de consulta ha sido la siguiente:

wind turbine AND optimization AND (“wake model” OR “wake effect” OR “wind farm design” OR “wind turbine placement” OR “wind farm layout” OR “turbines layout”) AND (“genetic algorithms” OR “evolutionary ensembles” OR “Artificial Intelligence” OR “meta-heuristics”)

El rastreo en las distintas plataformas de estos artículos, a través de palabras clave, arroja un total de 227 resultados en Scopus y 885 en Google Scholar con las mismas restricciones booleanas de búsqueda impuestas en ambas. La información de estos artículos es alojada en un fichero de formato CSV. Posteriormente, se aplica un protocolo *Systematic Literature Review* (SLR) [12], el cual marca las pautas de identificación, selección y valoración crítica de artículos de investigación en función de unas preguntas de interés a resolver. Se emplean los distintos criterios de validez (inclusión, exclusión y calidad), en forma de cuestiones, con el fin de seleccionar los artículos más relevantes del listado y que más se ajusten a la temática que nos atañe.

Tras el filtrado de artículos según los criterios de inclusión y exclusión de la Tabla 2.1, y la aplicación de los criterios de calidad establecidos, pasamos a disponer de un total de 89 artículos para analizar del conjunto total de artículos. Se ha obtenido la relación entre los autores de los distintos artículos identificados y se ha observado una clara tendencia en el uso de algoritmos genéticos como *Particle Swarm Optimization* (12 apariciones), optimización multiobjetivo (8 apariciones), y algunos otros que aparecen en menor medida como *Ant Colony Optimization* (ACO), *Bee Colony Optimization* (BCO) o *Pseudorandom Search Optimization* (PRSO).

Tabla 2.1: Protocolo SLR para el filtrado de artículos.

String de búsqueda (condicional booleano)	<i>wind turbine AND optimization AND (“wake model” OR “wake effect” OR “wind farm design” OR “wind turbine placement” OR “wind farm layout” OR “turbines layout”) AND (“genetic algorithms” OR “evolutionary ensembles” OR “Artificial Intelligence” OR “meta-heuristics”)</i>
Metadatos de la búsqueda	Título, Abstract, Keywords
Bases de datos consultadas	Google Scholar, Scopus
Criterio de inclusión	<ol style="list-style-type: none"> 1. Fecha de publicación posterior a 2013. 2. Documentos centrados en la optimización del rendimiento por posicionamiento de turbinas eólicas. 3. Hacen uso de vertientes de IA para llevar a cabo el proceso de optimización.
Criterios de exclusión	<ol style="list-style-type: none"> 1. Documentos no escritos en inglés. 2. Documentos de menos de 4 páginas. 3. Documentos duplicados en la búsqueda. 4. Otras versiones de estudios ya encontrados. 5. Tipologías de publicaciones que no sean artículos.
Criterios de calidad	<ol style="list-style-type: none"> 1. Grupo de clasificación del sitio donde se realiza la publicación. 2. ¿Hacen uso de Algoritmos Genéticos como herramienta de optimización? 3. ¿Aplican modelos de estela en la optimización? 4. Número de colaboraciones y citaciones de los autores entre artículos seleccionados.

La calidad de los resultados obtenidos por los algoritmos metaheurísticos está muy ligada a la selección apropiada de los hiperparámetros característicos del problema. Algunos artículos como [13] analiza de forma comparativa, entre dos algoritmos PSO, la incidencia en los resultados obtenidos de realizar el proceso de optimización de colocación de turbinas con varias combinaciones de pares de valores [(4,2), (2,4), (2,2), (4,4)] de aceleración personal y social (explicación más detallada de estos conceptos en la Sección 3.1.2 del Capítulo 3). Entre sus resultados se pueden observar diferencias de eficiencia que pueden llegar a rondar el 1 %, aunque esta diferencia puede incrementarse si se optimiza los propios hiperparámetros ajustando de forma precisa la magnitud de dichas constantes, lo que tendería a aumentar la eficiencia máxima conseguida, a pesar de existir un gran componente de aleatoriedad en el cálculo de la velocidad de la partícula.

Durante el análisis de la literatura realizado, se han identificado tres vertientes principales de optimización a través de las distintas vertientes de IA relacionado con el problema de maximización de la obtención de energía en las granjas eólicas.

- **Optimización de distribución de turbinas eólicas sobre el terreno:** obtención de mayor cantidad de energía con el uso de técnicas iterativas de posiciones de turbinas eólicas

sobre un terreno de unas dimensiones dadas. Se hace uso de modelos de estela para simular los déficits de velocidad provocados por la estela turbulenta que generan las turbinas aguas arriba en la dirección de la corriente de aire. En este ámbito, técnicas como el uso de algoritmos metaheurísticos o algoritmos de búsqueda son las más utilizadas [14, 15].

- **Selección de localización para la instalación de parques eólicos:** con la aparición del *Big Data* y la mejora en el tratamiento de datos masivos, se es capaz de analizar con distintas técnicas las posiciones óptimas para la construcción de una determinada granja eólica y la extensión del terreno necesaria para ello [16, 17, 18].
- **Optimización geométrica de turbinas individuales:** uso de técnicas algorítmicas de optimización y distintas metodologías que aportan una mejora de los perfiles alares que forman las palas de las turbinas, así como otros parámetros de torsión de la pala o la ley de cuerdas. Una mejora en los coeficientes de sustentación (en este caso de empuje) de la propia pala, provocará mayor velocidad de giro del rotor, provocando un aumento en la energía generada por la propia turbina para unas velocidades dadas [19, 20, 21, 22].

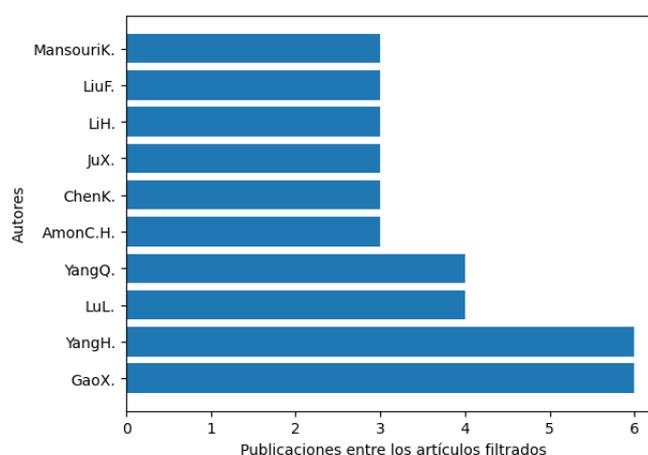


Figura 2.2: Los 10 autores con más publicaciones entre los artículos filtrados.

En referencia a los 89 artículos resultantes tras la aplicación del proceso SLR, observando la Figura 2.2 se puede ver el nombre de los 10 autores con más publicaciones realizadas. En la Figura 2.3 podemos visualizar, a través de los arcos que unen los nodos, las colaboraciones entre los distintos autores de los artículos resultantes del filtrado por los criterios impuestos. Aquellos autores que disponen de un número mayor de publicaciones presentes en la base de datos final van asociados a un nodo de color más oscuro, que, por lo general, suelen ser también los que más número de conexiones por arcos tienen con otros nodos.

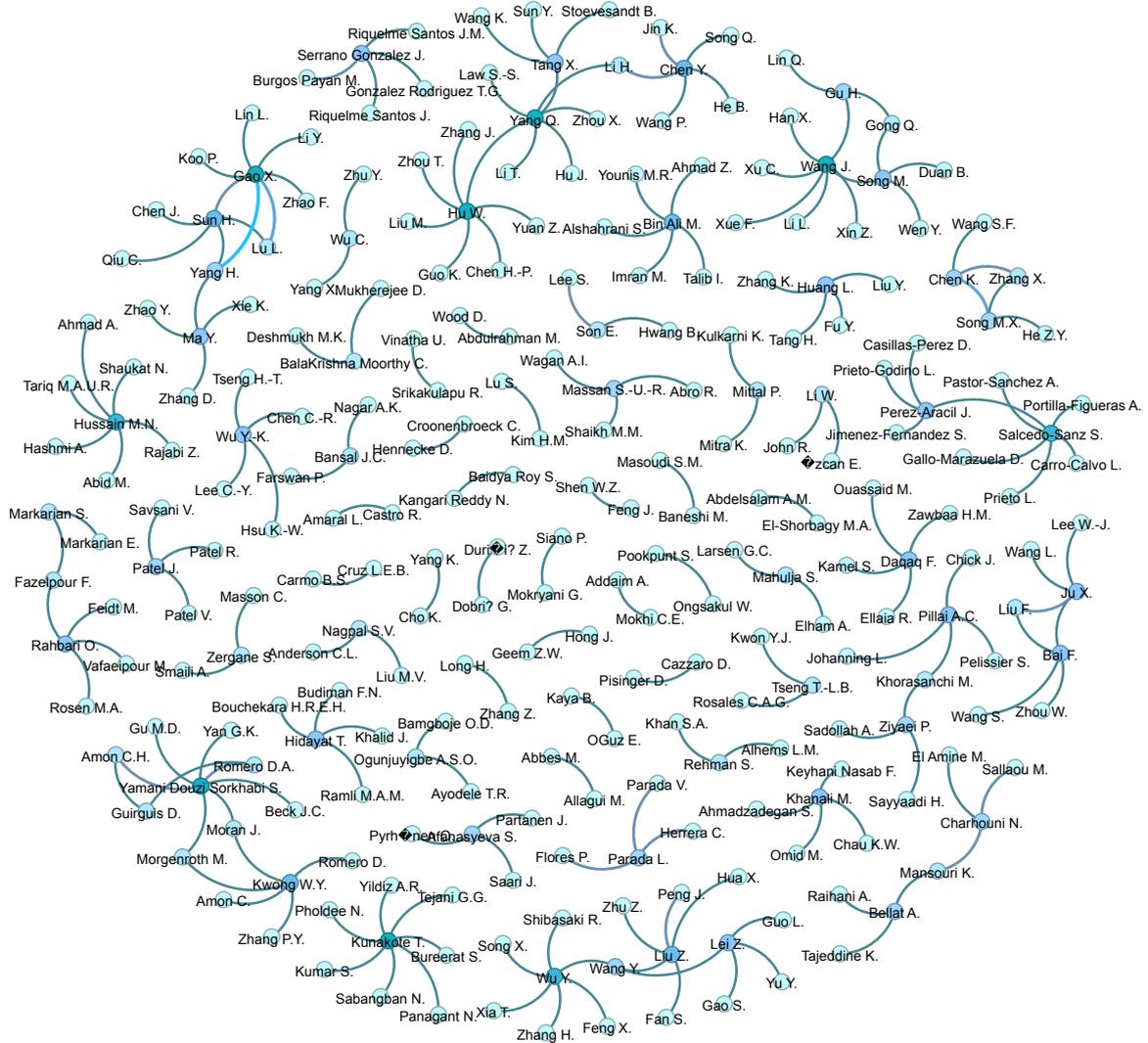
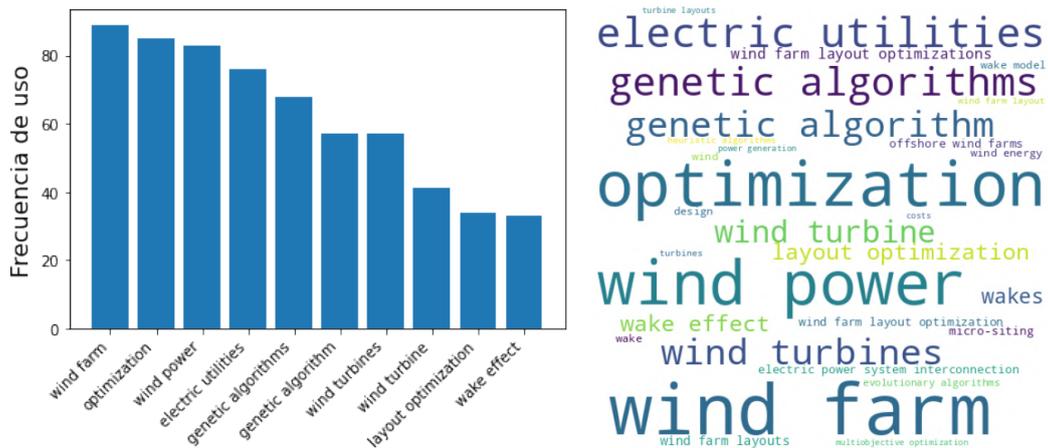


Figura 2.3: Relación entre autores de los artículos seleccionados.



(a) Top 10 keywords más usadas.

(b) Wordcloud de keywords

Figura 2.4: Frecuencia de uso de keywords tras el proceso SLR.

Después de realizar la selección de artículos anterior, se recopilan todas las *keywords* que aparecen en ellos y se vuelve a calcular nuevamente la frecuencia de utilización de los distintos términos. Al haber realizado un filtrado de artículos para la selección haciendo uso de una serie de *keywords* preseleccionados, es lógico que el nuevo resultado arroje palabras similares que ya habían sido detectadas previamente. Sin embargo, vemos la desaparición de ‘*wind energy*’ y la aparición de ‘*wind power*’, la eliminación de terminología específica como ‘*particle swarm optimization*’ y la aparición del plural/singular de algunos de los términos, lo que incrementa su importancia en caso de tratarlos como una única *keyword* y sumar sus valores de frecuencia de uso.

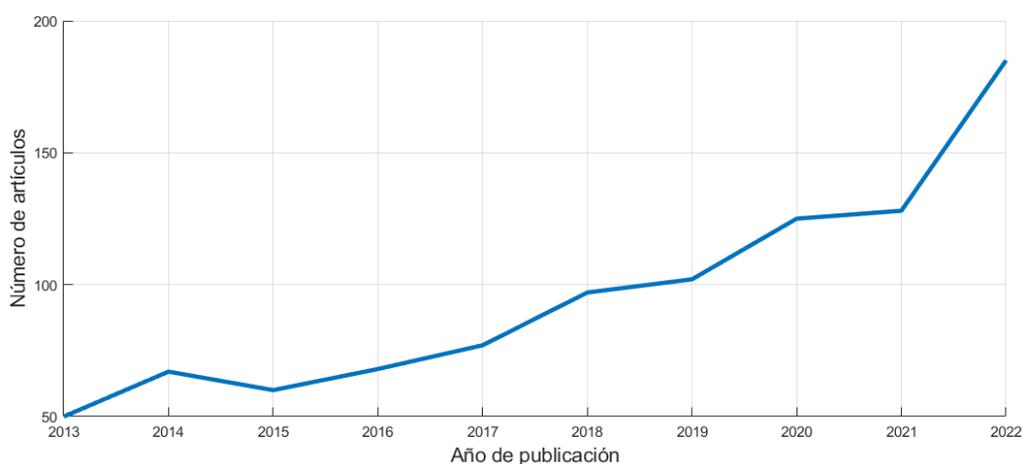


Figura 2.5: Evolución del número de publicaciones en los últimos 10 años.

Haciendo uso de la gráfica mostrada en la Figura 2.5 y atendiendo a los valores anuales de publicaciones sacados entre los artículos obtenidos de la búsqueda principal, podemos observar que el número de publicaciones realizadas en los últimos años triplican, en cuanto a cantidad, a los números de mediados de la década pasada. Debido a esto, podemos concluir que este campo de estudio está siendo muy desarrollado en la actualidad por causas como el auge de las energías renovables y el desarrollo masivo de las distintas ramas y tecnologías de la IA. También se ha observado una amplia participación del componente investigador asiático en este campo, identificados a través del gráfico de colaboración entre autores, la procedencia de las instituciones que redactaban los artículos, el filtro por idioma (la búsqueda sólo arrojó artículos escritos en chino e inglés) y la contabilización de las repeticiones de *keywords* (se han localizado *keywords* como ‘China’ o ‘Hong Kong’, tras el filtrado SLR, que se repetían en varias ocasiones). Ejemplo de esto pueden ser artículos encontrados tras la selección SLR como [23]

en el que hacen uso de algoritmos genéticos de población múltiple para maximizar la cantidad de energía generada por las granjas eólicas de la zona de Hong Kong.

Finalmente, se ha realizado un segundo filtrado de artículos tras la aplicación del procedimiento SLR, buscando aquellos que presentaban metodologías más innovadoras e interesantes, para reducir todavía más nuestro repertorio de artículos y poder analizarlos más en profundidad. Se han seleccionado 4 artículos que se ha considerado que definen de forma fidedigna el estado actual de las investigaciones en el campo tratado.

Entre los enfoques de investigación más abarcados, podemos destacar las presentaciones de modificaciones en algoritmos ya existentes para mejorar su funcionamiento, y su posterior comparativa con otros métodos y algoritmos de optimización existentes para cuantificar el grado de mejora. En [24] se hace uso del algoritmo *Coral Reefs Optimization with Substrate Layers* (CRO-SL), el cual implementa la división de la población en distintos sustratos sobre los que actúa un operador distinto, para intentar hallar la distribución de turbinas ideal sobre un terreno marino (granjas eólicas *offshore*) de carácter irregular que es discretizado. La división de la población en distintos sustratos marca una evolución de los valores de *fitness* de la población muy positivos, llegando a alcanzar soluciones de posicionamiento más adecuadas que otros métodos como las Aproximaciones Evolutivas, el algoritmo *Harmonic Search* (HS) y el algoritmo *Differential Evolution* (DE). En esta línea, el artículo [25] presenta una modificación del algoritmo *Biogeography-based Optimization* (BBO) inspirado en la migración de especies entre islas. El algoritmo es denominado *Fitness Difference Based BBO* (FD-BBO) e introduce pequeñas modificaciones de parámetros en la función de *fitness*, obteniéndose así una mayor exploración de las soluciones. Para verificar el correcto funcionamiento de esta modificación, el algoritmo es comparado con la versión clásica del BBO y una versión avanzada denominada *Blended BBO* (BBBO). Una de las características más importantes de este artículo es el hecho de tratar con turbinas con valores de altitud del buje y radio del rotor no homogéneos entre sí, es decir, con turbinas de distintas características geométricas. El hecho de realizar optimizaciones geométricas de las turbinas conforme se posicionan, podría llevar a incrementos muy notables en la energía generada por las turbinas eólicas. Entre el resto de los artículos revisados, se encontró que ciertos investigadores realizaron mejoras sobre otros algoritmos como el *Bionic Optimization* (BO) [26] o el *Chaotic Manta Ray Foraging Algorithm* (CMRFA) [27].

Uno de los parámetros más relevantes a la hora de colocar las turbinas en granjas eólicas terrestres son los cambios de altitud que se dan sobre la superficie, debido a los desniveles

de la topografía de la zona, y que inciden notablemente sobre los resultados obtenidos tras la aplicación de los distintos modelos de estela como se explica en la Sección 3.2 del Capítulo 3. En el artículo [28], los autores hacen uso de un *Cluster Identification Algorithm* (CIA) y un *Multi-Objective Genetic Algorithm* (MOGA) para resolver el problema de posicionamiento.

Otro gran problema que se presenta al intentar simular el funcionamiento de las turbinas eólicas es el tratamiento de los valores de las mediciones reales del viento obtenidos por los anemómetros, utilizados para el cálculo de la energía generada, debido a que el módulo de la velocidad que recogen son valores continuos. En [29] se optimiza el posicionamiento de aerogeneradores con el uso de un Algoritmo Genético, focalizando el estudio en la aplicación de un método de discretización de los módulos de velocidad y la generación de vectores velocidad sintéticos con el uso de las probabilidades recogidas en la rosa de vientos y una distribución de Weibull. Esta técnica ha sido utilizada en los experimentos de este TFG y se puede ver su aplicación en la Sección 5.2 del Capítulo 5.

Capítulo 3

Marco Teórico

Este capítulo recoge toda la información básica necesaria para comprender los experimentos planteados en el Capítulo 5. Se divide en un total de 4 secciones principales, empezando por la Sección 3.1 donde se explica el funcionamiento de los tres algoritmos metaheurísticos utilizados en los experimentos: BGA, PSO y CRO. A continuación, se detallan todos los procesos relacionados con los 7 modelos de estela en la Sección 3.2, una serie de conceptos sobre la forma de representación de los módulos y direcciones del viento a través de las rosas de los vientos en la Sección 3.3, y los distintos tipos de superposición que se pueden aplicar sobre los modelos para el cálculo de velocidad en un punto afectado por más de una estela en la Sección 3.4.

3.1. Algoritmos Metaheurísticos

Los algoritmos metaheurísticos son procedimientos de optimización iterativos inspirados en procesos y comportamientos observados en la naturaleza. Su objetivo es la obtención de soluciones óptimas (o subóptimas que se encuentren cerca de ellas), para problemas de gran complejidad que serían difíciles de abordar con otras técnicas como la programación lineal/no lineal. En función del tipo de problema a resolver, existen dos vertientes distintas de algoritmos metaheurísticos que se pueden usar, cuya elección dependerá de la complejidad que se vaya a abordar. En la Tabla 3.1 se pueden observar las dos ramas de los algoritmos metaheurísticos existentes, en este trabajo se hará uso de los algoritmos metaheurísticos basados en poblaciones.

Algoritmos metaheurísticos basados en trayectorias

Este tipo de algoritmos inicializan una única solución y trabajan de forma iterativa con ella, con

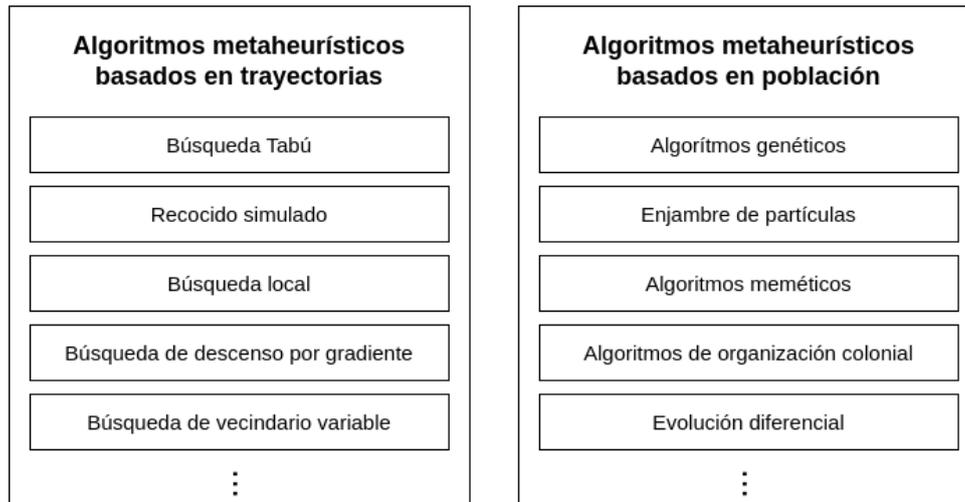


Figura 3.1: Tipos de algoritmos metaheurísticos.

el fin de explorar soluciones que estén cerca de la solución inicial que den mejores resultados. Esta metodología es usada para problemas que no son muy complejo y que no requieren buscar un gran abanico de alternativas para hallar un resultado óptimo o cercano a él.

Algoritmos metaheurísticos basados en poblaciones

Los algoritmos de optimización metaheurísticos basados en poblaciones (p-metaheurísticos), son aquellos en los que se trabaja con un conjunto de soluciones y una función de *fitness*, la cual, se busca maximizar o minimizar en función de la problemática a resolver y funciona como medida de idoneidad de la solución para el problema tratado. Este tipo de algoritmos cuentan con una serie de variables que van modificando y alterando al conjunto de soluciones con los que se trabaja, provocando cambios internos dentro del conjunto tras cada iteración.

3.1.1. Algoritmo Genético Básico (BGA)

La algoritmia genética es una rama de los denominados algoritmos evolutivos, la cual fue concebida por John Henry Holland [30], durante la década de los años 70 y cuya metodología de búsqueda probabilística se basa en la evolución de los individuos a través de mecanismos de selección natural, comportamientos en la naturaleza y la genética-molecular.

El Algoritmo Genético Básico (BGA) es la forma referencial en la que se apoyan muchas de las ramas existentes, las cuales han modificado la idea principal que propone. Está basado en el comportamiento generalizado de las distintas especies que definió Charles Darwin, en el cual, los individuos de una especie que mejor se adapten a un acontecimiento que afecta direc-

tamente en su supervivencia, serán los que prevalecerán. En este caso, se van generando nuevos miembros de la población a través de procesos de mezcla genética de individuos ya existentes (reproducción) y alterando aleatoriamente un número reducido de los genes resultantes (mutación) de algunos de ellos. Se somete a toda la población resultante a un criterio de selección que provocará las pertinentes defunciones según la selección realizada, dejando únicamente a los individuos que mejor se adaptan a la restricción establecida. Los componentes básicos del algoritmo son los siguientes:

- **Genes:** variables del problema de optimización que se alteran para dar lugar a una solución.
- **Cromosoma:** conjunto determinado de genes de un individuo.
- **Individuo:** cada posible candidato a la solución del problema planteado y que representan las distintas variables del problema en forma de cadena de genes.
- **Población:** conjunto de individuos/cromosomas.
- **fitness:** calidad de un individuo como solución al problema.
- **función de fitness:** problema que se busca resolver.

En la Figura 3.2 podemos observar el proceso que lleva a cabo un algoritmo genético básico para optimizar, a lo largo de varias generaciones, las poblaciones de individuos con el fin de obtener una solución próxima a la más ideal. Los algoritmos evolutivos no aseguran que la solución obtenida sea la mejor posible, pero sí garantizan una solución cercana a ella en un tiempo de análisis inferior a los procedimientos de fuerza bruta, los cuales analizan todas las soluciones posibles para hallar la mejor.

Partiendo de una población de partida (1ª generación), se someten a estos individuos a un análisis de su idoneidad para ser solución al problema a través de la función de *fitness* definida previamente. Tras ponderar su adecuación al problema, se pueden dar dos casos: que no cumpla el criterio de detención y que, por tanto, la población tenga que ser sometida a operaciones genéticas que darán lugar a una población nueva alterada con individuos distintos (nueva generación), o que cumpla el criterio de detención, lo cual provocará la selección de la población actual como solución al problema.

Entre las operaciones de los algoritmos genéticos básicos podemos distinguir 3 tipos:

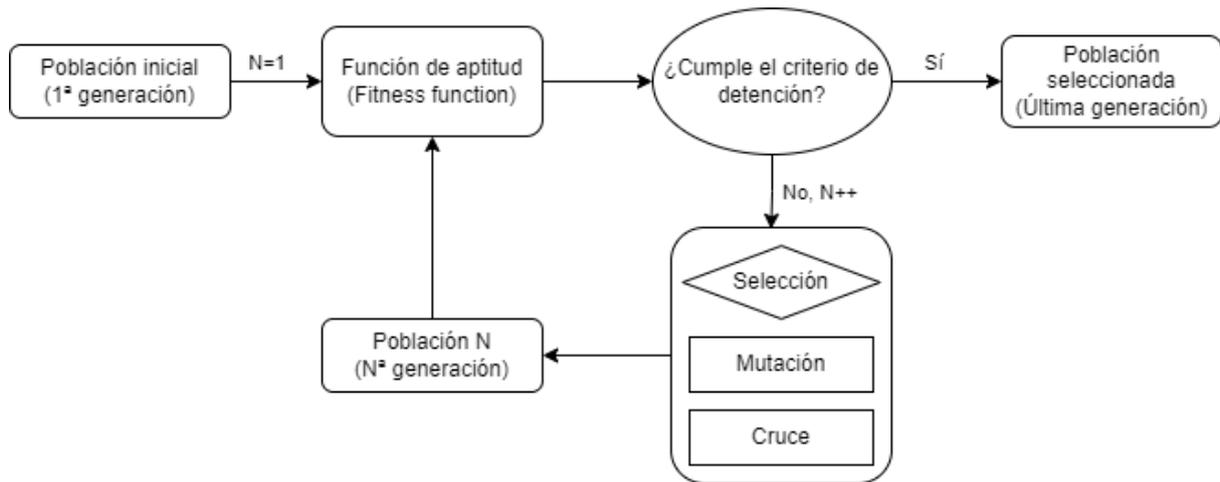


Figura 3.2: Diagrama de funcionamiento de algoritmo genético básico.

- **Selección:** operación mediante la cual se escoge, por medios de aleatoriedad, a los individuos de una población que serán partícipes en las modificaciones genéticas.
- **Mutación:** modificación aleatoria de determinados genes de un individuo para producir otro distinto al original.
- **Cruce:** generación de dos descendientes como resultado de la mezcla de los genes de dos progenitores seleccionados.
- **Defunción:** proceso final por el que los miembros del conjunto poblacional con peor *fitness* son eliminados, manteniendo un tamaño fijo de individuos.

3.1.2. Particle Swarm Optimization (PSO)

La Optimización por Enjambre de Partículas o *Particle Swarm Optimization* (PSO) es un algoritmo metaheurístico basado en poblaciones, en el cual cada solución (individuo) de la población se comprende como una partícula que ocupa una posición concreta del espacio de estados, y modifica dicha situación desplazándose en dicho espacio siguiendo a una partícula líder con mejor valor de *fitness* que el resto del conjunto poblacional.

Este algoritmo está basado en el comportamiento de enjambre de varios animales como pueden ser las bandadas de pájaros o los bancos de peces. Haciendo esta analogía, los distintos miembros de la población (partículas) siguen a un individuo alfa, el cual es el mejor posicionado para cumplir la meta final. Los enjambres se pueden caracterizar por las siguientes propiedades:

- Los individuos siguen una tendencia homogénea.

- Enjambre compuesto de gran cantidad de individuos.
- Interacción local basada en reglas simples.
- Autoorganización: velocidades adaptativas y evitar colisiones.

La actualización de la velocidad y posición de cada partícula en la población se realiza mediante las siguientes fórmulas:

$$v_{i+1} = w \cdot v_i + c_1 \cdot r_1 \cdot (x_p^{Best} - x_i) + c_2 \cdot r_2 \cdot (x_g^{Best} - x_i) \quad (3.1)$$

$$x_{i+1} = x_i + v_{i+1} \quad (3.2)$$

En las ecuaciones de movimiento de partículas definidas, podemos encontrar los siguientes hiperparámetros:

- c_1 (coeficiente cognitivo / aceleración personal): este parámetro controla la influencia de la mejor posición personal de cada partícula en su movimiento. Se recomiendan unos valores típicos para c_1 pertenecientes al rango [1.5, 2.5].
- c_2 (coeficiente/aceleración social): este parámetro controla la influencia de la mejor posición global encontrada por la colonia de partículas en el movimiento de cada partícula. Los valores típicos para c_2 están comprendidos en el rango [1.5, 2.5].
- w (factor de inercia): controla la relevancia de la velocidad actual de cada partícula en su movimiento. Un valor alto de w ($w > 1$) dará a cada partícula un movimiento más exploratorio (la partícula acelerará y buscará nuevas soluciones, no convergiendo en el óptimo temporal), mientras que un valor bajo de w ($w < 1$) dará un movimiento más explotativo (la partícula desacelerará y tenderá a estabilizarse). Valores típicos para w están en el rango de [0.8, 1.2].
- r_1 y r_2 : valores aleatorios uniformemente distribuidos, que comprenden unas magnitudes recogidas en un rango de [0, 1].

En resumen, el algoritmo PSO es un método de optimización iterativo que utiliza una población de partículas para buscar la solución óptima a un problema. Cada partícula se mueve a través del espacio de búsqueda, y su posición y velocidad se actualizan en función de su propia mejor solución encontrada hasta el momento y la mejor solución global encontrada por cualquier individuo histórico de la población hasta el momento.

3.1.3. Algoritmo Coral Reefs Optimization (CRO)

El algoritmo CRO, es un algoritmo metaheurístico evolutivo, cuyo funcionamiento simula los procesos de reproducción que se dan en los corales y sus acciones competitivas a la hora de buscar espacio en los arrecifes, los cuales se encuentran bastante saturados a nivel general. Se aplican modificaciones sustanciales al algoritmo genético clásico buscando incorporar nuevas herramientas a la hora de buscar soluciones y seleccionar los individuos de la población final.

Con el fin de mejorar la eficiencia y la obtención de resultados del algoritmo CRO tradicional, CRO-SL se creó con la idea de particionar las distintas poblaciones en grupos de un tamaño determinado, sobre los cuales se aplicaría un patrón de búsqueda concreto que diferiría de los aplicados en el resto de los fragmentos poblacionales.

Conceptos básicos sobre corales y arrecifes

Reproducción de los corales

Los corales son una especie capaz de reproducirse de forma sexual y asexual, ambos métodos pueden ser usados a lo largo de su vida debido a que estas especies generan gametos tanto femeninos como masculinos. La reproducción sexual se puede dar de forma externa o interna.

- **Reproducción sexual externa (*broadcast spawning*):** método de reproducción más usado por los corales, basados en la fecundación de huevos, expulsados al medio por un determinado pólipo, con espermatozoides a través de vía acuática. Esto genera las larvas que se desplazan por el arrecife hasta encontrar un lugar adecuado para arraigarse y crecer.
- **Reproducción sexual interna (*brooding*):** debida a la fecundación interna de huevos que no han sido expulsados por el pólipo, y que, por tanto, se encuentran en su interior. De igual forma que en la reproducción sexual externa, el espermatozoide de otro pólipo viaja por el agua hasta toparse con los huevos y los fecunda, con la diferencia de que los huevos residen en el interior de otro pólipo. Al estar protegidas las larvas por el pólipo les es más fácil sobrevivir ante posibles depredadores y arraigarse en el arrecife una vez son expulsadas al medio.
- **Reproducción asexual (*budding*):** se generan nuevos individuos genéticamente iguales, a partir de pólipos que han crecido lo suficiente, buscando expandir o colonizar nuevas

zonas. También se puede dar este tipo de reproducción por la fragmentación de un determinado pólipa debido a causas externas del medio.

Defunción y sustitución de individuos

Los corales se caracterizan por ser una de las especies más longevas existentes, pudiendo vivir centenares de años. A pesar de esto, a lo largo de su vida tienen que enfrentarse a numerosos peligros para poder sobrevivir. Una de las etapas en las que están más amenazados, es en su estado de larva, en la cual se ven acechados por varios depredadores que rondan los arrecifes y gran parte de ellos son consumidos por estos antes de encontrar un sitio donde arraigarse.

Funcionamiento CRO básico

Este algoritmo está desarrollado siguiendo los comportamientos explicados en la Sección 3.1.3. En un principio, se inicia una población aleatoria que ocupa determinadas posiciones de una cuadrícula de las distintas posiciones discretas posibles del arrecife modelado, con una tasa de ocupación establecida por el usuario, que dará el número de huecos que deben quedar libres.

Posteriormente se realiza una simulación de la reproducción de los corales usando distintos operadores para cada tipo de reproducción.

1. Reproducción sexual externa (*Broadcast spawning*):

- Selección uniforme de una fracción aleatoria de corales existentes para llevar a cabo este tipo de reproducción. Aquellos corales no seleccionados, serán escogidos para la reproducción asexual posteriormente.
- Con los genes de los corales seleccionados para esta forma reproductiva, se generan larvas a través de mecanismos y estrategias de mezcla de genes de los progenitores, surgiendo nuevos individuos de genética distinta.

2. Reproducción sexual interna (*Brooding*): formación de una larva a través de mecanismos de alteración de genes por mutación con la fracción de individuos restantes no seleccionados en el procedimiento anterior. Se considera la autofecundación de huevos interna en los corales hermafroditas.

3. **Posicionamiento de nuevos individuos:** las distintas larvas generadas en los procesos de reproducción sexual son sometidas a la función de fitness para competir por una posición aleatoria dentro del arrecife (representado por una cuadrícula de varias posiciones discretas) con el fin de ver si sustituye al individuo que ostenta esa posición o no. En caso de que la posición escogida por la larva esté vacía, no será necesario competir con otro individuo y se establecerá en ella directamente. El número de intentos de una larva para establecerse se limita de forma arbitraria por el usuario.
4. **Reproducción asexual (*Budding*):** una vez establecidas las larvas dentro del arrecife, se ordenan los individuos según los valores obtenidos de su función fitness y un número determinado de ellos son duplicados y alterados genéticamente con el fin de obtener variabilidad con respecto al individuo original. Posteriormente, se repite el proceso de posicionamiento explicado previamente para establecer los nuevos corales dentro del arrecife.
5. **Fase de depredación:** tras cada iteración de los procesos de generación de nuevos individuos, un número determinado de corales puede desaparecer para liberar huecos dentro del coral para que las larvas de futuras generaciones puedan establecerse. Se establece una probabilidad inferior de morir por depredación a los corales con peor función de *fitness*.

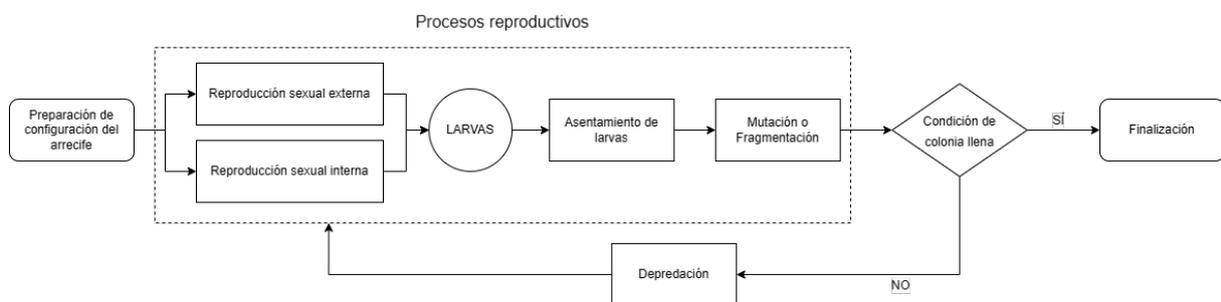


Figura 3.3: Diagrama de funcionamiento del algoritmo CRO.

Modificación CRO: *Coral Reefs Optimization with Substate-Layers* (CRO-SL)

Esta modificación parte de la idea de mejorar el funcionamiento del algoritmo CRO. CRO-SL es un algoritmo evolutivo competitivo, el cual divide la población en distintas capas de sustratos, las cuales hacen uso de distintos métodos de exploración.

Cada coral se asignará a un determinado sustrato y será sometido a un proceso reproductivo y selectivo distinto en función de la capa a la que pertenezcan. Sin embargo, la función de *fitness* será la misma en todos los casos.

En el primer CRO-SL se definieron 5 capas de sustratos distintas con sus respectivos operadores de búsqueda:

- **Búsqueda Harmónica (*Harmony Search* - HS):** método metaheurístico que emula el comportamiento improvisativo de los músicos de una orquesta durante el proceso de composición de una melodía agradable y armónica.
- **Mutación media por Evolución Diferencial (*Means Evolution from Differential Evolution* - DE):** método evolutivo que optimiza el proceso de obtención de las distintas soluciones a través de la comparación con una media de calidad de población. Almacena una serie de soluciones existentes, y las combina entre ellas a través de una serie de ecuaciones diferenciales, con el fin de obtener una solución óptima.
- **Cruce de 2 puntos (*2 Points Crossover* - 2Px):** el operador combina de forma aleatoria los genes de dos individuos seleccionan dos puntos de corte en su genoma. En el CRO-SL, se selecciona uno de los individuos de la propia capa de sustratos asignada al 2Px, y otro individuo de cualquier otra parte del arrecife, independientemente de si forma parte del mismo sustrato o no.
- **Cruce multipunto (*Multi Point Crossover* - MPx):** similar al 2Px, pero en este caso, se seleccionan varios puntos a través de un operador binario que marca los intervalos de corte dentro del genoma de un individuo y se solapan con los de otro.
- **Mutación Gaussiana (*Gaussian Mutation* - GM):** a través de una variable aleatoria de carácter gaussiano, se generan nuevos individuos que dependerán del valor de la misma. La función gaussiana de probabilidad de densidad se puede expresar como:

$$f_{G(0,\sigma^2)} = \frac{e^{-\frac{x^2}{2\sigma^2}}}{\sigma\sqrt{2\pi}} \quad (3.3)$$

Los valores máximos y mínimos de la desviación típica (σ), serán $\sigma = 0.2(A - B)$ y $\sigma = 0.02(A - B)$ respectivamente, siendo $[B,A]$ el dominio de búsqueda. Un mayor valor de σ provoca mutaciones más agresivas, por lo que este valor será mayor durante las primeras iteraciones e irá disminuyendo conforme avancen.

3.2. Modelos de estela

En el estudio de posicionamiento de turbinas eólicas es muy importante conocer cómo afecta el posicionamiento de cada uno de los elementos al resto de posibles posiciones aguas abajo que pueden ocupar los otros componentes de la granja eólica.

Para obtener, de manera aproximada, el comportamiento del viento afectado por el paso a través de una turbina eólica, se utilizan los denominados ‘modelos de estela’, los cuales tratan de expresar cómo disminuye el módulo de la velocidad respecto a la velocidad incidente aguas arriba debido a la turbulencia generada por la rotación de sus palas. En los siguientes Capítulos procedemos a explicar de forma breve las bases teóricas de varios modelos conocidos.

Durante el cálculo de los efectos de la estela, se tienen en cuenta una serie de parámetros generales, comunes a todos los modelos: la longitud de estela, su intensidad y el vector dirección del viento. La intensidad del viento está relacionada con la estabilidad de la propia atmósfera. En ocasiones, una intensidad alta que provoque una alta inestabilidad atmosférica puede conllevar a efectos positivos, como una disminución de la longitud de estela deficitaria propagada. En la Figura 3.4 se muestra un esquema de la disminución de velocidad del viento que se produce tras el paso del aire a través del rotor de la turbina eólica.

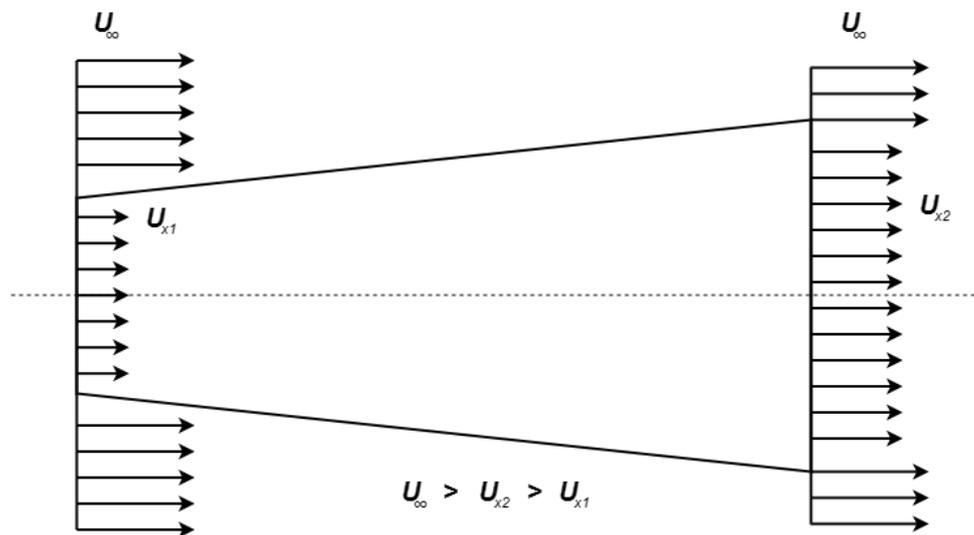


Figura 3.4: Esquema del efecto deficitario interno de la estela sobre la velocidad.

3.2.1. Modelo de Jensen

Es un modelo empírico propuesto por N.O. Jensen en [31], que hace una representación lineal de la estela, por lo que, al ser de primer orden, su implementación a la hora de iterar distintas configuraciones de granjas eólicas da unos tiempos de procesado relativamente bajos en comparación con otros modelos.

Jensen describe un perfil de velocidades escalonado y uniforme (*top hat*) y considera la aparición de un volumen de control inmediatamente aguas abajo de la turbina que provoca un decremento de la velocidad incidente en el resto de los puntos que se encuentran en su interior, que va perdiendo efecto conforme más alejadas quedan las posiciones internas respecto a la proyección entre el vector velocidad y la distancia entre la turbina afectante y la afectada.

Entre los parámetros utilizados por este modelo podemos citar: el diámetro del rotor (D_r), el coeficiente de empuje (c_T), la velocidad aguas arriba no alterada (U_∞), la velocidad en una determinada posición de la estela (U_x), el coeficiente de disminución de estela (k), la altitud (z) y el coeficiente de rugosidad del terreno (z_0).

$$U_x = U_\infty \cdot \left(1 - \frac{1 - \sqrt{1 - c_T}}{\left(1 + \frac{2 \cdot k \cdot x}{D_r}\right)^2} \right) \quad (3.4)$$

Este modelo hace uso del Teorema de la Superposición para el cálculo total del déficit de velocidad en un punto dado debido a la interacción con las múltiples turbinas colocadas en el terreno aguas arriba.

$$U_x = U_\infty \cdot \left(1 - \frac{1 - \sqrt{1 - c_T}}{\sqrt{\sum \left(1 + \frac{2 \cdot k \cdot x}{D_r}\right)^4}} \right) \quad (3.5)$$

Cabe resaltar que, si queremos conocer la potencia de una determinada turbina, en las Ecuaciones (3.4) y (3.5), el módulo de velocidad aguas arriba U_∞ puede adquirir diversos valores en función de si se ve afectada una zona o no por la presencia de aerogeneradores previos, o lo que es lo mismo, el valor de U_x solo corresponderá con U_∞ si el viento que baña un determinado punto se encuentra libre de perturbaciones y turbulencias.

Definimos el coeficiente de disminución de estela (k) mostrado en las dos ecuaciones previas como:

$$k = \frac{1}{2 \ln\left(\frac{z_h}{z_0}\right)} \quad (3.6)$$

La longitud de rugosidad (z_0) es una unidad de distancia que nos proporciona información sobre la altura sobre el terreno en la cual se da una velocidad media nula, siempre que el viento tenga una variación logarítmica dependiente de ella. Haciendo uso del área de elementos rugosos que están de cara al viento (S), el área horizontal media correspondiente a los diversos obstáculos (A_h) y la altitud respecto al terreno del elemento rugoso (z_e), podemos definir la longitud de rugosidad como se expresa en la Ecuación (3.7).

$$z_0 = \frac{1}{2} \frac{S \cdot z_e}{A_h} \quad (3.7)$$

En la Tabla D.1 del Apéndice D, podemos ver algunos ejemplos de valores característicos de rugosidades en función de la zona y los elementos ubicados sobre el terreno. En el caso de terrenos homogéneos sin mezcla de varias especificaciones, estos valores pueden ser tomados para mantener dicho valor constante en los cálculos y agilizar el proceso de cálculo.

3.2.2. Modelo Park2

En el artículo [32], se describe Park2 como una modificación del modelo Jensen, explicado en la Sección 3.2.1, que toma de él su perfil de velocidades escalonado (*top hat*) y su teoría de expansión lineal. La diferencia principal de Park2 con su modelo predecesor (Park1), es que considera que el efecto parcial de una determinada turbina equivale al efecto de la estela aguas abajo de la estela generada por ella en una determinada situación condicionada:

- No haya otras turbinas que afecten a la turbina estudiada.
- La turbina reciba el viento incidente con módulo igual al viento sin perturbar.

Otra característica relevante en el cálculo de la pérdida de velocidad conforme se interponen obstáculos en el transcurso del aire, es que, al contrario que en el primer modelo Park, éste omite el efecto reflectivo que supuestamente ejercería el suelo al recibir el impacto del aire, el cual se añadiría por superposición al viento incidente en otro elemento. Se tomó esta decisión debido a que, según varios análisis CFD, no se describía correctamente el efecto interactivo entre el suelo y la estela. La capa límite del suelo hace que la velocidad del aire justo en la superficie sea nula, condición que no cumplía el modelo reflectivo utilizado en Jensen o Park1.

$$\Delta U_x = U_\infty \cdot \left(1 - \sqrt{1 - c_T}\right) \cdot \left(\frac{D_r}{D_r + 2k \cdot (x - x_i)}\right)^2 \quad (3.8)$$

Con el fin de estimar el efecto del solapamiento de estelas en una determinada posición donde se encuentre situada una turbina eólica, Park2 hace uso de la Teoría de Perturbación clásica para encontrar la velocidad incidente en dicho punto.

$$U_x = U_\infty - \sum_i^N \Delta U_x \frac{A_{x,i}^{overlap}}{A_{rotor}} \quad (3.9)$$

Donde N es el número total de turbinas aguas arriba cuya estela llega al punto x , $A_{x,i}^{overlap}$ es la superficie parcial de la estela generada por la turbina i que baña el rotor de la posición x y A_{rotor} es el área del propio rotor. Resaltar que la metodología de superposición propuesta por el modelo Park2 es de tipo sumatorio lineal.

3.2.3. Modelo de Frandsen

El modelo de estela propuesto por S. Frandsen en [33], es muy utilizado en parques eólicos situados cerca de territorios marinos que abarcan grandes extensiones de superficie. Para que este modelo sea realmente representativo, es necesario que el territorio a estudiar cumpla con unas determinadas características:

- La zona debe ser una geometría rectangular.
- Aerogeneradores deben estar posicionados en filas.
- El espacio debe ser equidistante entre las filas y sus elementos.
- El flujo de viento debe ser paralelo a la orientación de las filas escogida.

El autor tuvo en cuenta una serie de consideraciones para simplificar las ecuaciones utilizadas: despreció la fuerza cortante generada por las turbulencias, la aceleración, la presión y la fuerza de la gravedad. Podemos hallar la velocidad en incidente en una turbina afectada por otra a través de la siguiente relación:

$$U_x = U_\infty \cdot \left(\frac{1}{2} - \frac{1}{2} \cdot \sqrt{1 - 2c_T \frac{A_r}{A_w}} \right) \quad (3.10)$$

Siendo A_r el área del disco del rotor, la cual puede ser calculada a través de las propiedades geométricas de las palas de la turbina ($A_r = \pi(D_r/2)^2$). Para calcular el área instantánea de la estela en un punto, debemos utilizar el diámetro instantáneo de la estela en dicho punto

($A_w(x) = \pi(D_w(x)/2)^2$), el cual se puede relacionar con el diámetro del rotor a través de la Ecuación (3.11).

$$D(x) = D_r \cdot \left(\gamma^{k/2} + \alpha \cdot s \right)^{k-1} \quad (3.11)$$

$$\gamma = \frac{1 + \sqrt{1 - c_T}}{2 \cdot \sqrt{1 - c_T}}, \gamma > 1 \quad (3.12)$$

$$\alpha = \gamma^{\frac{k}{2}} \cdot \frac{(1 + 2\alpha_{noj} \cdot s)^k - 1}{s} \quad (3.13)$$

Para realizar el cálculo de estas variables es necesario establecer el valor a las constantes no definidas. En función del autor y sus estudios, se le pueden asignar distintos valores experimentales a la constante k , por lo que, si utilizamos los resultados de Frandsen, $k = 2$, mientras que, si cogemos los de Schlichting, $k = 3$. Se tendrá en cuenta la distancia adimensional en las ecuaciones vistas previamente para el cálculo de α y γ , $s = x/D_r$, y se tomará el valor de $\alpha_{noj} = 0.05$.

3.2.4. Modelo de Larsen

El modelo presentado por G.C. Larsen en [34], hace uso de las ecuaciones de turbulencia sobre la capa límite descritas por Prandtl, y supone que el flujo inicialmente se considera incompresible ($Mach < 0.3$), estacionario y simétrico axialmente. El modelo de Larsen, al igual que muchos otros modelos, hace uso de la ecuación de Euler, la cual es una forma simplificada de la ecuación de Navier-Stokes que desprecia la viscosidad (cosa que puede asumir al restringir su uso a régimen incompresible como hemos indicado previamente). Este modelo fue recomendado por *Netherlands Energy Research Foundation* en su informe *European Wind Turbine Standards II* (EWTS II) de 1999 [35].

La Ecuación (3.14) que describe la pérdida de velocidad depende de varios factores que se desglosan a continuación:

$$U_x = U_\infty - \frac{U_\infty}{9} (c_T \cdot A_r \cdot x^{-2})^{1/3} \left(r^{3/2} (3c_1^2 \cdot c_T \cdot A_r \cdot x)^{-1/2} - \left(\frac{35}{2\pi} \right)^{3/10} (3c_1^2)^{-1/5} \right)^2 \quad (3.14)$$

El radio de la estela a una distancia determinada de la turbina que la genera es:

$$R_w(x) = \left(\frac{35}{2\pi} \right)^{1/5} (3c_1^2)^{1/5} (c_T \cdot A_r \cdot x)^{1/3} \quad (3.15)$$

Para definir el valor de la variable c_1 es necesario atender a las siguientes definiciones:

$$c_1 = \left(\frac{k \cdot D_r}{2} \right)^{5/2} \cdot \left(\frac{105}{2\pi} \right)^{-1/2} \cdot (c_T \cdot A_r \cdot x_0)^{-5/6} \quad (3.16)$$

$$x_0 = \frac{9.6D_r}{\left(\frac{2R_{9.6}}{k \cdot D_r} \right)^3 - 1} \quad (3.17)$$

$$k = \sqrt{\frac{m+1}{2}} \quad (3.18)$$

$$m = \frac{1}{\sqrt{1-c_T}} \quad (3.19)$$

$$R_{9.6} = a_1 \cdot \exp(a_2 \cdot c_T^2 + a_3 \cdot c_T + a_4) \cdot (b_1 \cdot I_a + 1) \cdot D_r \quad (3.20)$$

El radio de la estela que genera una turbina a una distancia de 9.6 veces el diámetro del rotor ($R_{9.6}$) se puede obtener de la Ecuación (3.20), la cual depende de una serie de coeficientes constantes de ajuste hallados de forma empírica. El valor de los coeficientes se encuentra recogido en la Tabla 3.1.

Tabla 3.1: Valores empíricos de los coeficientes del modelo Larsen.

Coeficiente	Valor
a_1	0.435449861
a_2	0.797853685
a_3	-0.124807893
a_4	0.136821858
b_1	15.6298

La intensidad de turbulencia ambiental juega un papel crucial en la interacción de la estela de un objeto en movimiento con el flujo de aire en la atmósfera circundante. Debido a que las perturbaciones turbulentas pueden afectar significativamente a la eficiencia de las turbinas eólicas que se ven inmersas en el mencionado flujo turbulento, por lo que es esencial tener en cuenta la intensidad de la turbulencia ambiental (I_a) al modelar la dinámica de la estela. En particular, la intensidad de turbulencia de la velocidad del viento en una zona se utiliza comúnmente como un parámetro cuantitativo para describir la magnitud de las perturbaciones turbulentas en

el flujo de aire. Este parámetro es un factor crítico de gran importancia que debe ser considerado cuidadosamente al modelar la estela de un objeto en movimiento, en nuestro caso, del giro de los rotores de las distintas turbinas eólicas. Larsen definió la intensidad ambiental para su modelo a través de la Ecuación (3.21), que depende de la distancia entre el punto de interés y la turbina que genera la perturbación, a la vez que de factores geométricos de la propia turbina como son el coeficiente de tracción (c_T) o el diámetro del rotor (D_r).

$$I_a = 0.29 \left(\frac{x}{D_r} \right)^{-1/3} \sqrt{1 - \sqrt{1 - c_T}} \quad (3.21)$$

3.2.5. Modelo de Bastankhah

Bastankhah propone una metodología simple y eficiente de cálculo de modelo de estela analítica para la predicción de la velocidad aguas abajo del viento. Para ello hace uso de un perfil de velocidades definido a través de una función de carácter gaussiano al contrario que los modelos descritos previamente, los cuales tenían un perfil lineal y constante a lo largo de la dirección radial de la estela.

Este modelo despreja los términos viscosos y de presión en la ecuación de momentos para encontrar el déficit de velocidad.

$$U_x = U_\infty \cdot (1 - C(x) \cdot f(r/\delta(x))) \quad (3.22)$$

Donde $C(x)$ hace referencia al vector normalizado del déficit de velocidad correspondiente a las distintas posiciones aguas abajo en el centro de la estela, r es la distancia radial desde el centro de la estela hasta el objeto inmerso en la corriente turbulenta y $\delta(x)$ representa la anchura radial total de la estela a una distancia x del punto desde el que se genera. La Ecuación ((3.22)) se puede reescribir como:

$$U_x = U_\infty \cdot (1 - C(x) \cdot e^{-\frac{r^2}{2\sigma^2}}) \quad (3.23)$$

$$C(x) = 1 - \sqrt{1 - \frac{c_T}{8(\sigma/D_r)^2}} \quad (3.24)$$

El autor define su modelo completo, con una ecuación que depende de un vector de componentes tridimensionales. En caso de querer simplificar la fórmula o analizar un caso bidimensional, bastaría con establecer componentes fijas de z (equivalente a analizar los resultados en

un plano paralelo a XY), aunque por lo general la metodología apropiada sería escoger el plano que hace equivalente z a ' z_h ' para anular el término de la ecuación y obtener simetría axial.

$$U_x = U_\infty \cdot \left(1 - \sqrt{1 - \frac{c_T}{8(k^* \cdot x/D_r + \varepsilon)^2}} \right) \cdot \exp \left(- \frac{1}{2(k^* \cdot x/D_r + \varepsilon)^2} \cdot \left(\left(\frac{z - z_h}{D_r} \right)^2 + \left(\frac{y}{D_r} \right)^2 \right) \right) \quad (3.25)$$

Desglosando algunos de los parámetros de la Ecuación (3.25), podemos encontrar una relación entre la desviación típica (σ) de la velocidad dentro del cono de la estela y el ratio de crecimiento (k^*).

$$\frac{\sigma}{D_r} = k^* \frac{x}{D_r} + \varepsilon = \frac{\partial \sigma}{\partial x} \frac{x}{D_r} + \left(\frac{\sigma}{D_r} \right)_{x \rightarrow 0} \quad (3.26)$$

3.2.6. Modelo de Zhang

El modelo de Zhang descrito en [36], toma de referencia el modelo de Jensen y lo modifica aplicándole una estela de carácter gaussiano a través de la utilización de la desviación estándar (σ), usada por Bastankhah en su modelo, la cual es una variable crucial a la hora de definir la geometría de distribución en el interior de la estela. Haciendo uso de la Ecuación (3.27) se puede calcular el radio de la estela a una determinada distancia gracias a la pendiente de la estela turbulenta de Zhang (k).

$$\frac{R_w}{D_r} = k_w \frac{x}{D_r} + 0.5 \quad (3.27)$$

La pendiente del cono de expansión de la estela se calcula a través de la fórmula tradicional del modelo de Jensen, realizando un pequeño ajuste a través de los coeficientes de intensidad de turbulencia total de la estela (I_w) y la intensidad de turbulencia de atmósfera en calma (I_0).

$$k = \frac{0.5}{\ln(z_h/z_0)} \quad (3.28)$$

$$k_w = k_t \frac{I_w}{I_0} \quad (3.29)$$

La Ecuación (3.30) define la velocidad en un determinado punto bañado por la estela de otra turbina según el modelo descrito por Zhang, el cual hace uso de una metodología no lineal para el cálculo.

$$U_x = U_\infty \left[1 - \left(\frac{\pi^2 - 4}{3\pi^2 - 16} - \sqrt{\left(\frac{\pi^2 - 4}{3\pi^2 - 16} \right)^2 - \frac{\pi^2 \cdot c_T}{3\pi^2 - 16} \left(\frac{D_r/2}{R_w} \right)^2} \right) \cdot \left(\cos \left(\frac{r}{R_w} \pi \right) + 1 \right) \right] \quad (3.30)$$

3.2.7. Modelo de Ishihara

Este modelo presentado en [37], cuenta con fórmulas muy similares a las vistas en modelo de Zhang visto en la Sección 3.2.6, debido a que ambos hacen uso de los modelos de Jensen y Bastankhah para llevar a cabo sus modificaciones. En este caso se relaciona de manera directa la pendiente de la estela con el cálculo de la desviación típica como se puede observar en la Ecuación (3.31). Aun así, existen una serie de equivalencias que pueden relacionar directamente el valor del radio de la estela a una determinada distancia (R_w) con el valor de la desviación típica (σ), en la Ecuación (3.32) se muestra una de ellas.

$$\frac{\sigma}{D_r} = k_w \frac{x}{D_r} + \varepsilon \quad (3.31)$$

$$\sigma = \frac{R_w}{2\sqrt{2\ln(2)}} \quad (3.32)$$

Los valores de la pendiente de estela (k_w) y la intersección de la función lineal (ε) se pueden obtener de forma directa a través de los valores de intensidad atmosférica sin perturbar (I_0) y el coeficiente de tracción de la turbina (c_T) a través de las siguientes ecuaciones:

$$k_w = 0.11c_T^{1.07} \cdot I_0^{0.2} \quad (3.33)$$

$$\varepsilon = 0.23c_T^{-0.25} \cdot I_0^{0.17} \quad (3.34)$$

Con todos estos valores, se calcula la velocidad en un determinado punto bañado por la estela de una turbina posicionada aguas arriba a través de la Ecuación (3.35).

$$U_x = U_\infty \cdot \left(1 - \frac{c_T}{\left[a + b + \frac{c}{(1+x/D_r)^2} \right]^2} \exp \left(\frac{r^2}{2\sigma^2} \right) \right) \quad (3.35)$$

Los parámetros a , b y c pueden hallarse a través de las Ecuaciones (3.36), (3.37) y (3.38) respectivamente, los cuales dependen de la intensidad de turbulencia en atmósfera en calma (I_0) y del valor del coeficiente de tracción de la turbina (c_T).

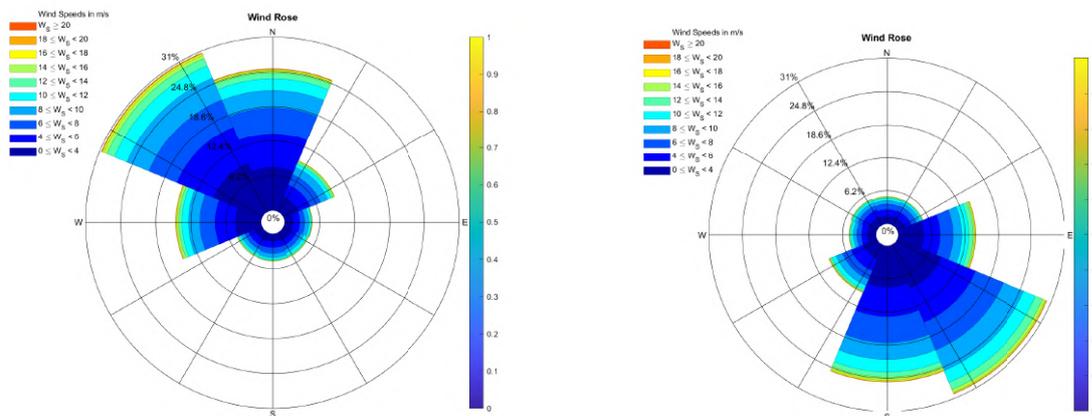
$$a = 0.93c_T^{-0.75} \cdot I_0^{0.17} \quad (3.36)$$

$$b = 0.42c_T^{0.6} \cdot I_0^{0.2} \quad (3.37)$$

$$c = 0.15c_T^{-0.25} \cdot I_0^{-0.7} \quad (3.38)$$

3.3. La distribución del viento: Rosa de los vientos

La rosa de los vientos es una representación gráfica de las frecuencias, intensidades y direcciones que toma el viento en una determinada zona, y a través de la cual, se puede llegar a analizar la velocidad del mismo. Es una herramienta muy útil a la hora de representar parte de la climatología de un determinado lugar, y es usado como método para orientar las pistas de los aeropuertos de forma que las aeronaves tengan vientos favorables durante las actuaciones de despegue y aterrizaje. Para elaborar la representación geométrica es necesario tomar muestras de los distintos valores previamente comentados de forma recurrente y en intervalos temporales muy amplios. La Figura 3.5 muestra la rosa de los vientos, de 8 direcciones, de los datos sintéticos y reales que se usarán para la realización de los experimentos descritos en la Sección 5.2 del Capítulo 5.



(a) Rosa de vientos con datos sintéticos.

(b) Rosa de vientos con datos reales.

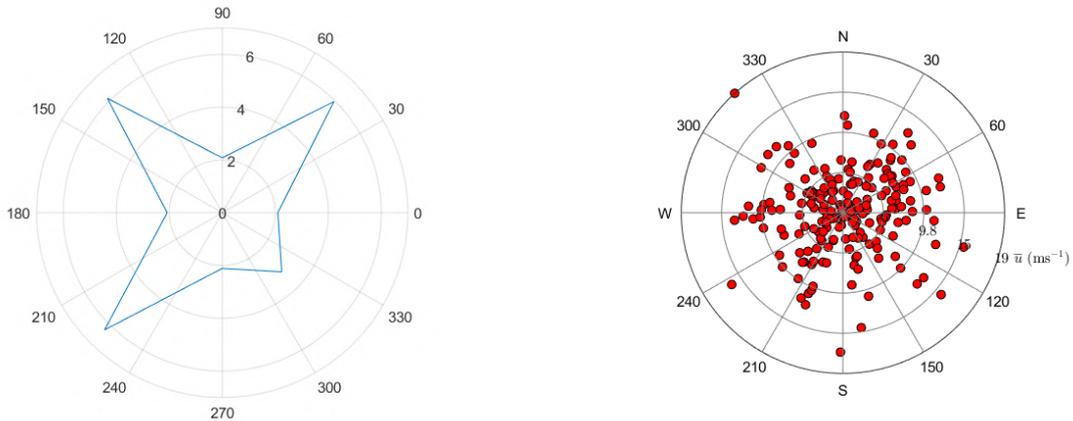
Figura 3.5: Representaciones genéricas de rosa de los vientos.

La forma de interpretar estos diagramas es muy simple, para ello, se hace uso de tres parámetros importantes a identificar en las barras: dirección, longitud y colores. Su longitud indicará cuánto de frecuentes son los vientos en la dirección a la que apunte, mientras que el color dará información sobre las velocidades que alcanzan. Este tipo de gráfica es capaz de informar de hasta 32 rumbos del viento, aunque en muchas ocasiones se hace uso de menos. En función de la precisión que tengan a la hora de representar los datos, podemos distinguir varios tipos distintos:

- **Rosa de los vientos clásica** [38]: modelo tradicional de rosa de los vientos, usado en la antigüedad, que recoge 12 direcciones cardinales que son norte, norte-noreste, noreste, este, sureste, sur-sureste, sur, sur-suroeste, suroeste, oeste, noroeste, norte-noroeste.
- **Rosa de los vientos de Vitruvius** [39]: representación de los vientos en 24 puntos cardinales principales equidistanciados en 15° , pensada para facilitar su dibujo a mano sin perder información respecto al modelo clásico.
- **Rosa de los vientos de 4 puntos cardinales**: contiene los 4 puntos principales de orientación con una diferencia de 90° entre ellos, que son el norte, el sur, el este y el oeste.
- **Rosa de los vientos de 8 puntos cardinales**: representación que recoge 8 puntos principales con una diferencia de 45° entre sí, que incluyen los 4 puntos cardinales principales, así como los puntos intermedios adicionales que marcan el noreste, el sureste, el suroeste y el noroeste.
- **Rosa de los vientos de 16 puntos cardinales**: representación con 16 puntos principales diferenciados en 22.5° , que agrupan los 8 puntos cardinales mencionados anteriormente, así como otros puntos añadidos, como el norte-noreste, el este-noreste, el este-sureste, el sur-sureste, el sur-suroeste, el oeste-suroeste, el oeste-noroeste y el norte-noroeste.
- **Rosa de los vientos de 32 puntos cardinales**: es la más detallada y precisa, ya que tiene 32 puntos principales que indican direcciones específicas con una diferencia de 11.25° , permitiendo mayor precisión en la representación de las propiedades del viento.

Tal y como se muestra en la Figura 3.6 hay dos formas alternativas a las presentadas en la Figura 3.5. Podemos representar la rosa de los vientos en forma de gráfico de dispersión, lo cual nos ofrece la posibilidad de añadir mayor número de variables de posicionamiento en el propio

gráfico y disponer así de más información. Otra forma de representar los datos menos habitual, es con una geometría en forma de estrella cuya distancia entre el origen polar de coordenadas y cada uno de los puntos que forman parte del contorno, cuantifican la frecuencia del viento en una determinada dirección, y el gradiente interno que rellena de la geometría da información sobre la magnitud de los módulos de la velocidad.



(a) Rosa de los vientos en gráfico de estrella

(b) Rosa de los vientos en gráfico de dispersión

Figura 3.6: Representaciones alternativas de rosa de los vientos.

3.4. Cálculo de afectación de las turbinas aguas abajo

En un parque eólico se sitúan multitud de aerogeneradores, lo que hace que la velocidad del viento incidente se vea afectada por los obstáculos situados aguas arriba y ciertos elementos más retrasados vean una velocidad inferior del viento, lo cual afecta a la energía generada.

Para estimar el efecto de la estela generado en un punto concreto donde se sitúa una turbina, Lissaman en [40], propuso el siguiente modelo de superposición de tipo lineal para el cálculo de la velocidad en un determinado punto debido a la estela generada por turbinas en otras posiciones:

$$U_x = U_\infty - \sum_i^N (U_\infty - U_{x,i}) \quad (3.39)$$

Donde N es el número de turbinas situadas aguas arriba cuya estela afecta a la zona de estudio, U_∞ es la velocidad sin perturbar y $U_{x,i}$ es la velocidad perturbada en ese punto por la estela asociada a una turbina i .

En los distintos modelos de estela consultados, los autores utilizan diversas metodologías de superposición de estela para averiguar la velocidad incidente sobre la turbina en un punto concreto. Estas expresiones pueden variar en función del método por el que se realice la diferencia entre la velocidad incidente aguas arriba menos el déficit de velocidad que provoca la estela de las distintas turbinas. En la Tabla 3.2 se ha recogido los distintos tipos y sus expresiones.

Tabla 3.2: Métodos de superposición de estelas. Información obtenida de [41].

Tipo	Ecuación
Lineal	$\left(1 - \frac{U_x}{U_\infty}\right) = \sum_N \left(1 - \frac{U_{x,i}}{U_\infty}\right)$
Cuadrática	$\left(1 - \frac{U_x}{U_\infty}\right)^2 = \sum_N \left(1 - \frac{U_{x,i}}{U_\infty}\right)^2$
Geométrica	$\frac{U_x}{U_\infty} = \prod_N \left(\frac{U_{x,i}}{U_\infty}\right)$
Balance de energías	$U_\infty^2 - U_x^2 = \sum_N (U_\infty^2 - U_{x,i}^2)$
Estela dominante	$\frac{U_x}{U_\infty} = \min_N \left(\frac{U_{x,i}}{U_\infty}\right)$

A la hora de realizar la superposición de estelas generadas por distintas turbinas, cuya área de expansión se solapa en un determinado punto de interés en el que centramos el análisis, es importante tener en cuenta cómo realizar la composición de pérdida de velocidad y cómo influyen entre sí las desaceleraciones del aire. Para ello, se suelen usar diversas metodologías de superposición que hacen variar los resultados finales, en cierta medida, según la técnica escogida. En este trabajo, los modelos de estela han sido programados con metodologías de superposición cuadrática.

La Tabla 3.3 muestra los métodos de superposición orientados a turbinas. Estas metodologías cuentan con unos acrónimos que hacen referencia al efecto de aplicar una técnica u otra para el cálculo de la velocidad final que percibe un punto del plano, o en este caso una determinada turbina aguas abajo. A continuación, se describen brevemente cada uno de ellos:

- **Ambient-based velocity deficit linear sum (AVDLS)**: método que considera un decremento en la velocidad del viento axial detrás del rotor de la turbina primaria, lo que resulta en un cambio en la velocidad del viento en la ubicación de la turbina secundaria. Esta metodología utiliza el ajuste de mínimos cuadrados para encontrar los coeficientes de superposición que mejor se ajustan a los datos experimentales del modelo.

- **Rotor-based velocity deficit linear sum (RVDLS)**: técnica de carácter similar al AVDLS, pero en lugar de considerar solo la reducción de la velocidad axial, también tiene en cuenta la reducción de la velocidad radial del viento detrás del rotor de la turbina primaria.
- **Ambient-based velocity deficit root sum square (AVDRSS)**: es un método que utiliza una fuente de Rankine para modelar la estela de la turbina primaria y considera una reducción en la velocidad axial del viento en la ubicación de la turbina secundaria. También utiliza el método de superficie de fuente para modelar la estela.
- **Rotor-based velocity deficit root sum square (RVDRSS)**: metodología parecida al AVDRSS, pero que también tiene en consideración el déficit de velocidad radial del viento detrás del rotor de la turbina primaria.
- **Ambient-based energy deficit linear sum (AEDLS)**: es una técnica que considera tanto la reducción de la velocidad axial como la turbulencia generada por la turbina primaria en la estela. Utiliza el método de mínimos cuadrados para ajustar los coeficientes de superposición.
- **Rotor-based energy deficit linear sum (REDLS)**: método de superposición que considera los efectos de la turbulencia en la estela de la turbina primaria y utiliza el método de mínimos cuadrados para ajustar los coeficientes de superposición.

Tabla 3.3: Principios de aproximación de estela por superposición. Información obtenida de [42].

Principio de Superposición	Ecuación
AVDLS	$U_x = U_\infty - \sum_{i=1}^n (U_\infty - U_i)$
RVDLS	$U_x = U_\infty - \sum_{i=1}^n (U_{x,i} - U_i)$
AVDRSS	$U_x = U_\infty - \sqrt{\sum_{i=1}^n (U_\infty - U_i)^2}$
RVDRSS	$U_x = U_\infty - \sqrt{\sum_{i=1}^n (U_{x,i} - U_i)^2}$
AEDLS	$U_x = \sqrt{U_\infty^2 - \sum_{i=1}^n (U_\infty^2 - U_i^2)}$
REDLS	$U_x = \sqrt{U_\infty^2 - \sum_{i=1}^n (U_{x,i}^2 - U_i^2)}$

Capítulo 4

Herramientas y desarrollo

En este capítulo, se explica de forma concisa, los procesos de selección de tecnologías para llevar a cabo los experimentos y el funcionamiento de los *scripts* creados. Es muy recomendable consultar los ficheros alojados en el repositorio de GitHub, que se referencia en el Apéndice C, para seguir el hilo de desarrollo de código. El capítulo se divide en un total de 7 secciones principales, iniciándose en la Sección 4.1, donde se explica qué es MATLAB y los motivos por los que se decidió escoger esta herramienta en lugar de otras tecnologías. Posteriormente, en la Sección 4.2 se detalla el funcionamiento de la estructura de datos que almacena la información sobre las configuraciones de terreno, modelos de estela y algunos parámetros específicos del algoritmo seleccionado para el análisis. La estructura de datos es usada en todos los cálculos realizados por la función de *fitness*, la cual se explica en la Sección 4.3. Una serie importante de conceptos clave para el entendimiento de los cálculos son los tipos de estela existentes, el cómo se lleva a cabo la identificación de turbinas afectadas y cómo se calcula la energía generada por cada una de las turbinas, todo ello recogido en la Sección 4.4. Finalmente, las Secciones 4.5, 4.6 y 4.7 especifican el funcionamiento de los algoritmos programados, BGA, PSO y CRO-SL respectivamente, junto con sus parámetros específicos, y en la Sección 4.8 se expone el proceso por el que se recopilan los valores de interés de los ficheros de resultados tras la finalización del proceso de optimización.

4.1. Entorno de programación: MATLAB

MATLAB (MATrix LABoratory) es un software de programación numérica, creado por la compañía Mathworks, ampliamente utilizado en diversas disciplinas científicas e ingenieriles.

Se trata de un entorno de programación interactivo de alto nivel que destaca por su capacidad en análisis numérico, cálculo matricial y visualización de datos. Gracias a esas funcionalidades, MATLAB se ha convertido en una herramienta esencial para resolver problemas complejos que implican grandes conjuntos de datos y cálculos matemáticos intensivos. En particular, su capacidad para trabajar con matrices de gran tamaño lo convierte en una herramienta esencial para el análisis de datos y la simulación de sistemas.

- **Interfaz de usuario:** cuenta con una interfaz intuitiva y fácil de usar. Además, ofrece herramientas para facilitar la programación, como la consola de comandos, el editor de códigos y la ventana de visualización.
- **Lenguaje de programación:** utiliza un lenguaje de programación de alto nivel que permite la creación de *scripts* y funciones con una sintaxis simple y fácil de aprender. También ofrece una amplia gama de funciones y herramientas integradas para realizar cálculos matemáticos y manipulaciones de datos. Sus archivos se denotan con las extensiones ‘.m’ para *scripts*, ‘.mat’ para ficheros de datos y ‘.mlx’ para código en vivo.
- **Bibliotecas y paquetes:** dispone de gran cantidad de bibliotecas y paquetes que ofrecen una variedad de herramientas para aplicaciones específicas, como el procesamiento de imágenes, análisis de señales y simulación de sistemas.
- **Simulink:** herramienta que permite la creación de modelos de sistemas dinámicos y la simulación de su comportamiento. Esta herramienta es especialmente útil en aplicaciones de ingeniería y ciencias físicas, para realizar los llamados diagramas de bloques.
- **Visualización de datos:** ofrece una amplia gama de herramientas para la visualización de datos, como pueden ser los gráficos 2D y 3D, diagramas de barras e histogramas. Estas herramientas son útiles para la representación y la comunicación de resultados.

Se ponderaron dos alternativas para el desarrollo del código necesario para los experimentos, realizarlas en MATLAB o en Python, haciendo uso de las librerías de NumPy, Pandas y Matplotlib. Tras analizar las ventajas de ambos lenguajes, y debido a la naturaleza matricial del problema a la hora de realizar los cálculos de energía generada, se optó finalmente por MATLAB como opción preferente para el desarrollo de los distintos *scripts*.

Tabla 4.1: Comparativa de características entre MATLAB y Python (NumPy) para implementar algoritmos metaheurísticos con matrices.

Características	MATLAB	Python (NumPy)
Rendimiento	Optimizado para cálculos numéricos y matriciales, lo que resulta en una mayor eficiencia para operaciones de matrices de gran tamaño.	NumPy también está optimizado para operaciones de matrices, pero puede ser menos eficiente que MATLAB en ciertos casos.
Velocidad de cálculo	MATLAB es generalmente más rápido que NumPy en operaciones de matrices de gran tamaño.	NumPy es menos eficiente que MATLAB en algunas operaciones de matrices, pero su velocidad de cálculo se puede mejorar con el uso de bibliotecas adicionales como Numba y Cython.
Funciones y paquetes integrados	Amplio conjunto de funciones integradas para operaciones matriciales, estadísticas y visualización de datos.	NumPy también tiene una amplia variedad de funciones integradas para operaciones matriciales, pero puede tener menos funciones integradas para visualización de datos y estadísticas.

4.2. Estructura de datos de modelos, algoritmos y terreno

Antes de ejecutar la fase de optimización de cada uno de los algoritmos metaheurísticos, es necesario establecer una serie de parámetros que definan el funcionamiento del mismo. Para facilitar la activación del procedimiento por parte del usuario, se le pide por consola una serie de variables correspondientes al modelo, tamaño de terreno y número de turbinas. Los valores otorgados se introducen en la función ‘genData()’ que devolverá una estructura de parámetros, tanto generales de terreno y viento, como característicos del modelo de estela seleccionado.

4.2.1. Campos de la estructura generales

Las variables generales de la estructura ‘data’, son aquellas que se definirán siempre, independientemente del modelo de estela escogido, por ser parámetros comunes y usados por todos ellos. Corresponden a características geométricas de las turbinas como el coeficiente de tracción (‘data.CT’), el diámetro de la turbina (‘data.D_r’) y el área del rotor (‘data.A_r’); propiedades del terreno como la distancia de seguridad entre turbinas (‘data.dSec’), número de posiciones de

un lado de la *grid* ('data.Kgr'), número de turbinas a colocar (data.Nturb) o número de puntos de ploteo para estelas suavizadas ('data.numPoints'); y propiedades del viento a través de la declaración de la intensidad de turbulencia estacionaria ('data.I0').

4.2.2. Campos de estructura específicos

En función del nombre del modelo de estela que se le pase como argumento a la función 'genData()', ésta determinará a través de una serie de condicionantes, el tipo de campos de estructura que debe definir para el correcto funcionamiento de las funciones asociadas.

Algunos campos son compartidos por varios modelos, sobre todo los del modelo Jensen, sobre el cual se fundamentan la mayoría de los modelos actuales. El modelo de Zhang, por ejemplo, hace uso de las mismas definiciones y valores del modelo de Jensen, y le añade otras nuevas que permiten obtener mayor precisión a la hora de estimar las velocidades.

Campos del modelo Jensen: recoge los campos de altitud sobre el terreno ('data.Z'), rugosidad del terreno (data.Z0), pendiente de la estela ('data.k') y ángulo de la estela ('data.a').

Campos del modelo Park2: se definen de manera directa los campos de pendiente de estela ('data.k') y ángulo de estela ('data.a').

Campos del modelo Frandsen: agrupa entre sus campos el valor directo de la pendiente de estela ('data.k'), la constante α_{noj} ('data.alpha_noj') y la distancia adimensional para ploteo de la estela ('data.s') evaluada para un total de puntos dado por 'data.numPoints'.

Campos del modelo Bastankhah: para el uso del modelo completo extendido es necesario la creación de los campos de desviación típica ('data.sigma'), valor de la desviación típica cuando 'x' tiende a 0 ('data.epsilon'), la tasa de crecimiento ('data.growth') y la pendiente de la estela ('data.k'). En caso de querer hacer uso de la versión simplificada bidimensional, únicamente se generará el campo de la pendiente de estela transversal ('data.ky').

Campos del modelo Larsen: este es el modelo de mayor complejidad y que necesita más número de campos. Recoge una serie de constantes ('data.m', 'data.k', 'data.a1', 'data.a2', 'data.a3', 'data.a4', 'data.b1', 'data.x0', 'data.c1'), la intensidad de turbulencia añadida ('data.Ia'), la intensidad de turbulencia de estela ('data.Iw'), la distancia adimensional evaluado en

un número de puntos equivalente a la magnitud de numPoints ('data.s') y el radio de la estela en cada uno de los puntos evaluados ('data.R_x').

Campos del modelo Ishihara & Quian: para este modelo, es necesario definir como campos una serie las constantes 'a', 'b' y 'c' ('data.a', 'data.b' y 'data.c'), la intersección de la función lineal ('data.epsilon') y la pendiente de la estela ('data.k').

Campos del modelo Zhang: agrupa en sus parámetros la altitud sobre el terreno ('data.Z'), la rugosidad del terreno ('data.Z0'), la pendiente de estela Jensen ('data.k.t'), la pendiente modificada de Zhang ('data.k'), la intensidad de turbulencia añadida ('data.Ia'), la intensidad de turbulencia total de la estela ('data.Iw'), la distancia adimensional para los puntos de evaluación ('data.s') y el radio de estela en los mismos puntos evaluados ('data.R_x').

4.3. Estructura de la función de *fitness*

La función de *fitness* de nuestro problema nos dará la energía generada por todas las turbinas, para una distribución de vientos horaria, en un marco temporal especificado. Esta función recibe como hiperparámetros la población total (cada individuo es un vector con las posiciones ocupadas por las turbinas en la matriz), el vector de vientos y la estructura de datos 'data', que almacena los parámetros del modelo de estela y tamaño del terreno.

Se ha modificado la estructura de las funciones usadas en los análisis llevados a cabo en el artículo [10], reestructurando el flujo de trabajo y las variables involucradas, de manera que, en función del modelo de estela seleccionado, genere las constantes correspondientes a dicho modelo y haga uso de unas funciones de representación de límites de estela y cálculo de velocidades incidentes en cada una de las turbinas. La Tabla 4.2 recoge de forma resumida la estructura utilizada en el cálculo del *fitness* de los individuos.

La nueva estructura de flujo de trabajo requiere de una serie de *scripts* que no se usaban en la estructura original, y que han sido elaborados para tener una mejor organización y disponer de una estructura que sea válida para el cálculo del *fitness* de varios modelos de estela distintos sin necesidad de cambiar de *script* principal, ni realizar distintas estructuras de código en función del modelo. Se utiliza la función 'f_powerPlantsT_fast.m' para administrar las dos fases del proceso, en las cuales se obtienen las distancias entre turbina afectante y afectada, y se calcula la velocidad incidente sobre las turbinas.

Tabla 4.2: Estructura y variables de la función de *fitness*.

Función	Inputs	Outputs	Descripción
f_powerPlants_T	vVec gr data	pwr_T gan_T cost_T obj_T dt	Función principal encargada de analizar los datos de la estructura 'data' y gestionar el uso de unos u otros valores transmitidos a unas funciones concretas del modelo.
f_powerPlants_f1	vVec gr data	xUDef yUDef	Función encargada de detectar el modelo de estela y llamar a la función 'getDist' correspondiente.
f_powerPlants_f2	vVec gr ppPower xUDef yUDef data	pwr_t pwrGen Ux gan cost obj	Función que detecta el modelo de estela, llama a la función 'wm_' correspondiente para calcular los déficit de velocidades y calcula la energía generada por la granja en la hora analizada.
getDistTrapz getDistSmooth	vDir gr data	Trapz: xUDef Smooth: xUDef yUDef	Calcula la distancia entre la turbina que genera la estela y la turbina que se encuentra afectada por ella.
wm_jensen wm_park2 wm_frandsen wm_bastankhah wm_larsen wm_ishihara wm_zhang	m vMod xUDef yUDef data	Ux	Calcula la velocidad incidente sobre cada una de las turbinas posicionadas en el terreno.

En primera instancia, es necesario conocer la afectación mutua entre las estelas de cada una de las turbinas eólicas para averiguar el déficit de velocidad producido. Las estelas cambiarán de dirección y geometría en función de la matriz de vientos utilizada y el modelo de estela seleccionado. Para averiguar qué turbinas se encuentran sometidas a la estela de otras, es necesario conocer el modelo de estela para estimar los límites de la estela por uno de los dos métodos explicados en la Sección 4.4. Haciendo uso de la función ‘f_powerPlants_f1’, se obtienen dos matrices de afectación que recogen las distancias longitudinales (‘xUDef’) y transversales (‘yUDef’) desde la turbina que genera la estela hasta la turbina sometida a ella.

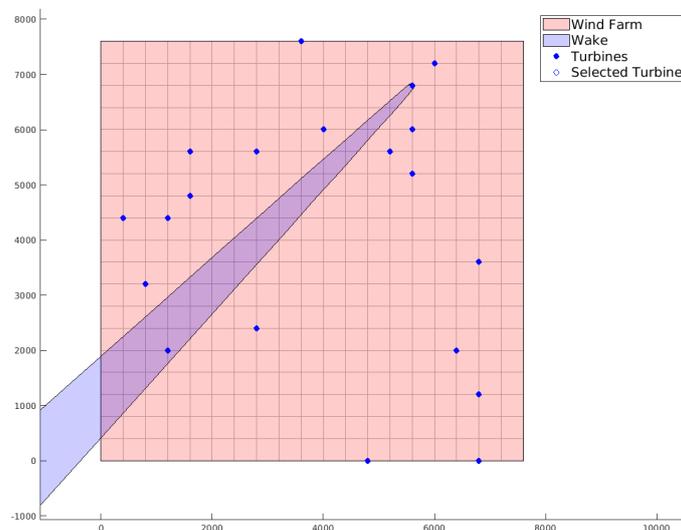


Figura 4.1: Comprobación de afectación de estela durante la fase de obtención de distancias de la función de *fitness*.

En segundo lugar, haciendo uso de las ecuaciones del modelo de estela seleccionado, llamados desde la función ‘f_powerPlants_f2’, se calcula la pérdida de velocidad en cada una de las turbinas por las distintas estelas afectantes gracias a las matrices ‘xUDef’ e ‘yUDef’ halladas en el paso anterior. Esta velocidad deficitaria, permite calcular la velocidad final incidente en cada una de las turbinas para las distintas horas analizadas, y poder así poder estimar la energía generada para cada hora por ellas.

Finalmente, tras haber recibido el gestor principal (‘f_powerPlantsT_fast’) de vuelta el módulo de velocidad que incide sobre cada una de las turbinas eólicas posicionadas en el terreno procede a realizar el cálculo de la energía generada a lo largo del periodo temporal establecido. Como *inputs* para el cálculo será necesario introducir la matriz ‘gr’ que recoge la distribución

de turbinas sobre el terreno, la matriz 'vVec' que recoge los vectores viento de cada hora y la estructura 'data'. Como *output* devolverá la energía total generada 'pwr_T' y una estructura 'dt' con los datos desglosados de la energía total generada por cada hora para el vector vVec correspondiente.

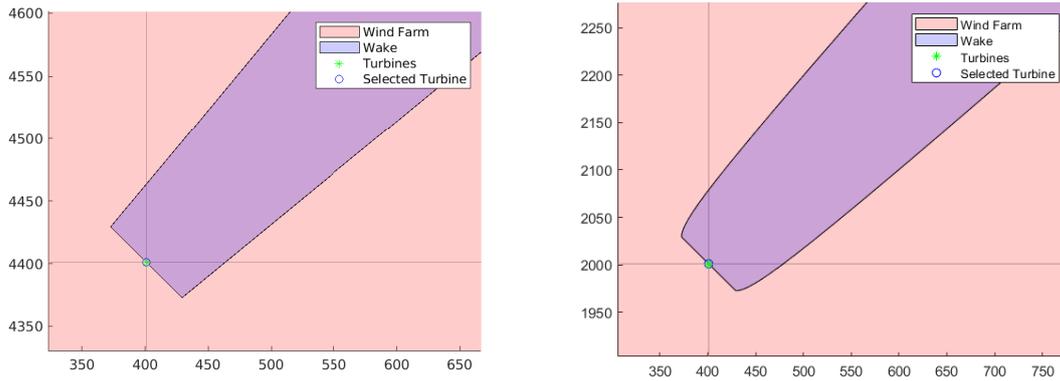
4.4. Implementación de modelos de estela

Inicialmente, se disponía de base con dos modelos ya implementados (Jensen y Bastankhah) en el artículo de referencia [10], los cuales, sólo hizo falta reordenarlos y adaptarlos a la nueva estructura de código impuesta. Más allá de Jensen y Bastankhah, se han implementado otros 5 modelos de estela, lo que hace un cómputo global de 7 modelos de estela distintos para la realización de las pruebas experimentales.

Durante el cálculo de la función de *fitness*, los modelos de estela son llamados en la segunda fase para la obtención de la energía final generada a lo largo del periodo de tiempo establecido. En función del modelo seleccionado que disponga el *struct* 'data', se llamará a uno u otro en el proceso, alterando en cierta medida las potencias obtenidas y los valores correspondientes a los distintos algoritmos genéticos de la función de *fitness*.

Durante la primera fase, en la cual hay que determinar las turbinas que se ven afectadas por las distintas estelas, es necesario establecer los puntos delimitantes de ella con el uso de dos métodos distintos:

- **Estela trapezoidal:** el método más rápido, es utilizar un polyshape de cuatro lados, lo cual nos deja con una estela con forma trapezoidal que se utiliza para los modelos de Jensen, Park2 y Bastankhah, como se muestra en la Figura 4.2a.
- **Estela suavizada:** basado utilizar un polyshape multipunto que te genera una geometría de estela con forma similar a un cáliz de copa, haciendo uso de la función "logspace()" de MATLAB para determinar los puntos de interpolación de la geometría. Se utiliza para los modelos de Frandsen, Larsen, Ishihara y Zhang, y se muestra representada en la Figura 4.2b.



(a) Nacimiento de la estela trapezoidal.

(b) Nacimiento de la estela suavizada.

Figura 4.2: Tipos de geometrías de estelas.

En la segunda fase, las funciones encargadas del cálculo de déficit de velocidad de cada uno de los modelos de estela serán llamadas por el segundo gestor 'f_powerplants_f2.m' que identificará el modelo según el valor introducido en el campo 'data.model' y llamará a la función 'wm_('data.model').m' pertinente. Estas funciones recibirán como *inputs* las matrices de afectación longitudinal y transversal de las estelas provocadas por las otras turbinas ('xUDef' e 'yUDef'), el índice de la turbina afectada ('m'), el módulo del vector vientos ('vMod') y la estructura 'data'. En este caso, podemos distinguir nuevamente dos tipos de modelos de estela, aquellos que sólo tienen en cuenta la afectación longitudinal de la estela que únicamente recibirán de *input* 'xUDef' (Jensen, Frandsen y Park2) y aquellos que hacen uso tanto de la afectación transversal y longitudinal (Larsen, Bastankhah, Ishihara y Zhang) que recibirán tanto 'xUDef' como 'yUDef'.

4.5. Proceso de elaboración: Algoritmo Genético Básico

El funcionamiento general del Algoritmo Genético Básico se explica en la Sección 3.1.1. En nuestro caso, hemos hecho uso de los procesos de cruce (recombinación de genes) y mutación (alteración aleatoria de genes) para producir los nuevos individuos, candidatos a ocupar parte de la población sustituyendo a otros individuos que formaban parte de ella previamente. Dado que se busca elaborar el algoritmo genético más básico posible, el proceso de recombinación programado establece un solo punto de corte aleatorio (1Px) en el genoma de los progenitores, generando así 4 partes de tamaño complementario que se combinan entre sí, dando lugar

a dos nuevos individuos (descendientes). Existen otras alternativas como el cruce de 2 puntos (2Px) o multipunto (MPx), las cuales puede ser conveniente aplicar en caso de que se trabaje con genomas de gran extensión o se busque encontrar soluciones de carácter más exploratorio.

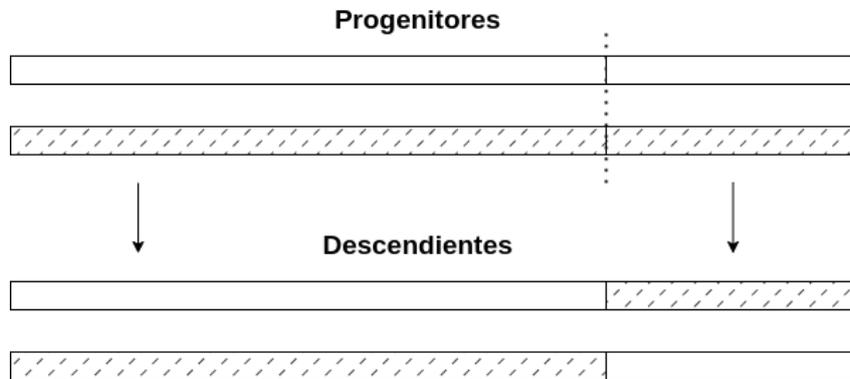


Figura 4.3: Representación gráfica del proceso de cruce 1Px.

Para hacer uso de este algoritmo, es necesario definir una serie de variables que incluiremos en la propia estructura de datos ('data') definida en el Capítulo 4.2.

El *script* principal del algoritmo se encarga, inicialmente, de preguntar al usuario por consola el nombre del modelo de estela que busca utilizar, la longitud del cuadrilátero que define la superficie y el número de turbinas a posicionar sobre el plano. A través de estos datos, genera la parte de la estructura del modelo de estela y el terreno, de forma directa desde los propios inputs introducidos por consola y con el uso de la función 'genData()'.

Posteriormente, se añaden de forma directa las variables de control del propio algoritmo que definen hiperparámetros como el tamaño poblacional, número de genes, probabilidades de mutación, número de cruces y número de generaciones. Después de esto, es necesario inicializar una población, la cual se generará de forma aleatoria. Esta población estará formada por individuos en formato de vector que contendrán las posiciones ocupadas por las turbinas en formato de número entero comprendidos entre los valores $[1, Kgr^2]$, siendo 'Kgr' el valor de la longitud del lado del terreno introducido por consola por el usuario al comienzo del proceso. Una vez ya dentro del bucle generacional, el cual es un bucle *for* que se repetirá los distintos procesos establecidos dentro del mismo el número de veces que se haya establecido como valor de la variable 'nGenerations'. En la Tabla 4.3 se pueden ver todos los campos añadidos a las estructuras con sus respectivas descripciones.

Tabla 4.3: Campos característicos de ‘data’ para el uso de Algoritmo Genético Básico.

Campo	Descripción
‘data.crossParam’:	Parámetro de selección de individuos que se reproducen (<i>float</i>).
‘data.crossTimes’	Cantidad de individuos que se reproducen en una generación (<i>integer</i>).
‘data.mutationGenProb’	Probabilidad de que un gen del individuo se altere aleatoriamente (<i>float</i> ≤ 1).
‘data.mutationIndProb’	Probabilidad de que se someta al proceso de mutación a un individuo (<i>float</i> ≤ 1).
‘data.populationSize’	Número total de individuos de una población (<i>integer</i>).
‘data.nGenes’	Número de genes de un individuo, con un valor igual al número de turbinas a posicionar descrito por el valor de ‘data.Nturb’ (<i>integer</i>).
‘data.nGenerations’	Número total de generaciones que se evaluarán (<i>integer</i>).

El primer proceso que se lleva a cabo es el cruce entre individuos. Se anida un bucle *for* al principal que repetirá las acciones descritas tantas veces como sea el valor de la variable ‘nCross’. En él, se seleccionarán dos individuos por iteración del bucle, y se recombinarán entre sí por metodología 1Px, generando dos nuevos individuos ciclo que se irán almacenando en una nueva población de carácter temporal y auxiliar, compuesta únicamente por esos descendientes.

La nueva población de individuos, producidos por el cruce de genes entre miembros de la población principal, se someterá posteriormente a un proceso de mutación, el cual modificará una serie de genes de determinados individuos de dicha población según las probabilidades de mutación de individuo y probabilidades de mutación de genes establecidos en la inicialización de hiperparámetros del algoritmo a través de las variables ‘mutationIndProb’ y ‘mutationGenProb’. Este proceso selecciona un número determinado de configuraciones de turbinas de forma aleatoria según la probabilidad establecida por ‘mutationIndProb’, y dentro de las configuraciones escogidas, cambiará de posición una serie de turbinas aleatorias, según las probabilidades definidas en ‘mutationGenProb’, a huecos no ocupados por otras turbinas del *grid* que define el terreno.

Finalmente, se realiza el proceso de selección, el cual realiza la comparativa de la población de progenitores con la nueva población de descendientes, y selecciona, entre el conjunto completo de las dos poblaciones, los individuos con mayor valor de *fitness*, produciendo la desaparición (defunción) de los individuos que no cojan en el cupo de individuos establecido y

manteniendo así el tamaño de población inicial para todas las generaciones. Se almacenarán en *arrays* los valores media poblacional ('avgFitness') y el máximo valor de fitness registrado por un individuo ('maxFitness') por cada generación.

4.6. Proceso de elaboración: PSO

Como podemos observar en la Figura 4.4, se ha propuesto un algoritmo PSO que divide cada configuración de turbinas, en tantos grupos de partículas como número de turbinas se tengan que posicionar ('data.Nturb'), y cada grupo estará compuesto de tantas partículas como tamaño se asigne a la población ('data.populationSize'). Esto nos deja como resultado 'data.Nturb' *arrays* de partículas de tamaño 'data.populationSize' \times 1, sobre los cuales se aplicarán las reglas de movimiento características de este modelo que se describen en la Sección 3.1.2.

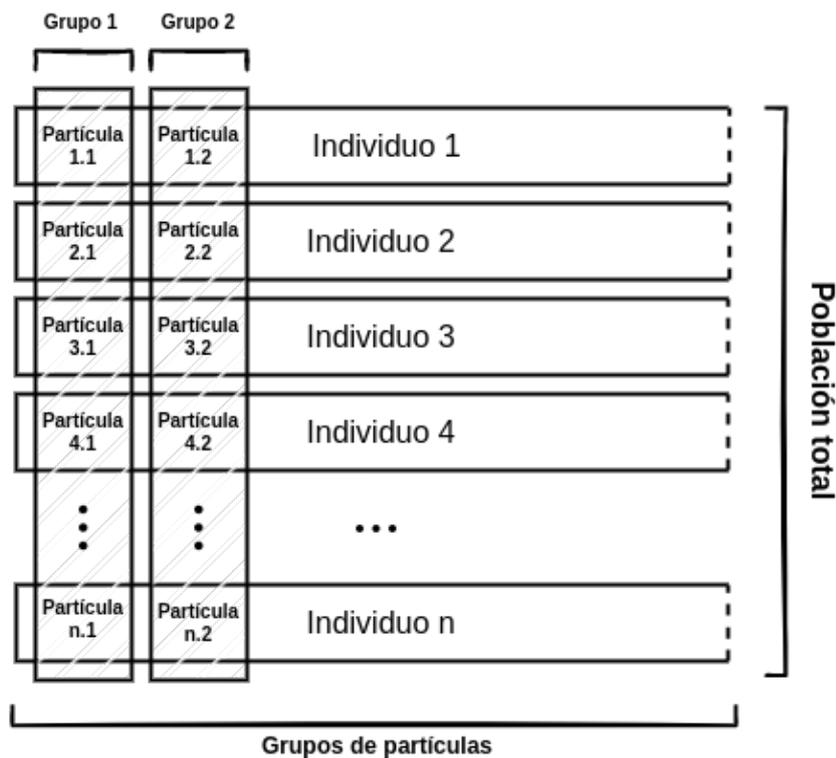


Figura 4.4: División de los individuos de la población en grupos de partículas.

Primero, es necesario generar la estructura 'data' a través de 3 parámetros que se introducen por el usuario a través de la consola (modelo de estela, tamaño del terreno y número de turbinas). Una vez generada la estructura 'data' general, se añaden a ella los campos específicos del modelo que se pueden consultar en la Tabla 4.4. Tal y como se analiza en [13] (mencionado

más extensamente en el Capítulo 2), estos campos específicos, correspondientes a los distintos hiperparámetros del algoritmo, incidirán de forma decisiva en el correcto funcionamiento del proceso de optimización y la calidad de resultados obtenidos, definiendo comportamientos cruciales como la posibilidad de intercambio de partículas de un individuo entre grupos, la capacidad de exploración de soluciones o el peso que tengan la mejor solución local y global en el desplazamiento de cada partícula.

Tabla 4.4: Campos característicos de ‘data’ para el uso de algoritmo PSO.

Campo	Descripción
‘data.t’	Tiempo por cada fase de movimiento (<i>float</i>).
‘data.w’	Coefficiente de inercia (<i>float</i>).
‘data.c1’	Coefficiente de aceleracion personal (<i>float</i>).
‘data.c2’	Coefficiente de aceleracion social (<i>float</i>).
‘data.v0’	Velocidad inicial (<i>float</i>).
‘data.particles’	Número de partículas, equivalente a ‘data.populationSize’ (<i>integer</i>).
‘data.nGroups’	Número de grupos de partículas, con un valor igual al número de turbinas a posicionar descrito por el valor de ‘data.Nturb’ (<i>integer</i>).
‘data.nIterations’	Número de fases de movimiento (<i>integer</i>).

Será necesario almacenar en paralelo una población adicional, a modo de historial de mejores resultados propios, que alojará las distribuciones de turbinas que alcanzaron los mejores resultados de *fitness* en el histórico de cada uno de los individuos. Las distribuciones de turbinas de esta nueva población solo se actualizarán con los valores de la población real en el caso de que el nuevo valor de *fitness* conseguido por el individuo sea superior al conseguido en iteraciones previas por su distribución almacenada en población auxiliar. Al comienzo del algoritmo, en el proceso de inicialización de variables, los valores de ambas poblaciones serán idénticos por falta de historial previo.

Usando la población auxiliar, podemos identificar la mejor distribución personal cada individuo (lo cual nos dará la posición de la mejor posición histórica personal de una partícula) y la mejor distribución global de toda la población (que identificará la mejor posición histórica global alcanzada entre todas partículas). Estos valores serán utilizados en la función de movimiento para estimar la velocidad de la partícula (ver Ecuación (3.1)), que determinará la distancia a la

que se desplazará cada partícula con respecto a su posición actual.

Una vez se consiguen las nuevas distribuciones de turbinas de la población en una iteración determinada, tras ejecutar el proceso de movimiento, se procede a realizar la comparativa entre el valor obtenido y los valores de mejor *fitness* histórico de cada individuo y el mayor *fitness* global alcanzado. En caso de que el nuevo valor de *fitness* hallado sea superior al asociado a la distribución correspondiente en la población auxiliar, esta pasa a sustituirla y se almacena allí como mejor solución local histórica de dicho individuo. Si se da la circunstancia de que también es mayor que el máximo valor de *fitness* obtenido por todos los individuos entre todas las iteraciones, esta distribución pasará a ser la mejor solución histórica global también, y marcará la tendencia de movimiento en las iteraciones posteriores hasta que sea sustituida.

4.7. Proceso de adaptación: CRO-SL

Para realizar la comparativa con un algoritmo metaheurístico con más tiempo de desarrollo y estudio que los dos que se han programado anteriormente (Algoritmo Genético Básico y PSO), se va a modificar y adaptar la estructura de un algoritmo CRO-SL cedido por Jorge Pérez Aracil, profesor e investigador de la Universidad de Alcalá de Henares. Se ha realizado una modificación y adaptación al problema de colocación de turbinas sobre el terreno, de un algoritmo CRO-SL que dispone de 5 sustratos incorporados: Búsqueda Armónica (HS), Evolución Diferencial (DE), Cruce Multipunto (MPx), Cruce de 2 puntos (2Px) y Mutación Gaussiana (GM), las cuales hacen su búsqueda de soluciones más efectiva, alterando el peso que tiene cada sustrato conforme va avanzando el número de generaciones del arrecife. Los *scripts* usados para el funcionamiento del algoritmo ya incorporaban tres estructuras que recogían los campos necesarios para la correcta ejecución del programa. Dado que es necesario hacer uso de la función de *fitness* descrita en la Sección 4.3, se generará la estructura 'data' usada en los otros dos algoritmos programados, pero en este caso, no será necesario incorporarle parámetros específicos del algoritmo a esta estructura ya que el CRO-SL ya traía las suyas propias ('cn', 'p' y 'f') que recogen todos los hiperparámetros necesarios para el proceso de optimización. La estructura 'cn', que hace referencia al coral, recoge todos los hiperparámetros del proceso y acciones de optimización del algoritmo, 'p' aloja los valores de la propia población de larvas y 'f' agrupa todos los datos del *fitness* del coral. En las Tablas 4.5, 4.6 y 4.7 se describe la composición de las tres estructuras de datos (*struct*) utilizadas en el algoritmo.

En cada una de las iteraciones, se evalúan las larvas generadas por cada uno de los sustratos y se comparan con un número determinado aleatorio de posiciones del coral que pueden estar vacías u ocupadas por otra larva. Si se da la situación de que la posición evaluada del coral se encuentra vacía, la larva se acopla en ese lugar directamente, mientras que, si se encuentra ocupado, se somete a un proceso de disputa en el que se compara el valor de *fitness* entre ambas larvas y se decide cuál es más apta para ocupar ese hueco. Si la nueva larva no consigue encontrar un espacio al que adherirse en un número establecido de saltos, ésta termina desapareciendo.

Conforme vayan pasando las generaciones, el coral se llenará con las larvas que mejor *fitness* tengan, y los sustratos predominantes harán uso de los genes de esas larvas para generar nuevas de ellas que puedan desafiar a las larvas ya posicionadas para sustituirlas. De los tres procedimientos de optimización es el que contiene una estructura más compleja, requiriendo más recursos y tiempo para hallar las soluciones.

Tabla 4.5: Campos característicos de ‘cn’ para el uso de algoritmo CRO-SL.

Campo	Descripción
cn.size	Tamaño del coral (<i>integer</i>).
cn.filledRate	Porcentaje del coral ocupado (<i>float</i>).
cn.n_rep	Número de repeticiones (<i>integer</i>).
cn.p_d	Probabilidad de defunción de una larva (<i>float</i>).
cn.holes	Número de huecos del arrecife (<i>integer</i>).
cn.substrates	Vector que recoge los sustratos utilizados (<i>integer</i>).
cn.lB	Límite inferior que puede recibir un gen de larva (<i>float</i>).
cn.uB	Límite superior que puede adquirir un gen de larva (<i>float</i>).

Tabla 4.6: Campos característicos de ‘f’ para el uso de algoritmo CRO-SL.

Campo	Descripción
f.coral	<i>Fitness</i> general de la población del coral.
f.larvae	<i>Fitness</i> general de las larvas generadas.
f.bestHistory	Historial del <i>fitness</i> máximo del coral en cada generación.
f.best	Mejor <i>fitness</i> histórico alcanzado en el coral.

Tabla 4.7: Campos característicos de ‘p’ para el uso de algoritmo CRO-SL.

Campo	Descripción
p.coral	Conjunto de larvas del coral (<i>integer</i>).
p.larvae	Conjunto de genes de la larva (<i>integer</i>).
p.coral_filled	Parte del coral que está ocupada (<i>integer</i>).
p.betterSubs	Indica el sustrato de la mejor larva (<i>integer</i>).
p.operator2ind	Sustrato al que pertenece la larva (<i>integer</i>).
p.best	Genes de la mejor larva (<i>integer</i>).
p.paramFilled	Número de larvas del coral (<i>integer</i>).
p.paramIdxSubstrate	Tamaño del coral para cada sustrato (<i>integer</i>).

4.8. Extracción de datos

Los datos resultantes tras alcanzar el número total de generaciones/iteraciones de los distintos algoritmos de optimización poblacionales, ejecutando las configuraciones correspondientes de modelos, tipo de viento y distribuciones, se guardan en un fichero ‘.mat’, con un nombre identificativo, el cual se genera a través de un *string* declarado con la concatenación de los *inputs* introducidos al comienzo del algoritmo, según el siguiente formato *modelo_algoritmo_tamaño_viento*.

Los nombres de los ficheros almacenados en la carpeta de resultados son revisados y almacenados en un *array* por un *script*, encargado de importar al entorno de trabajo los valores alojados en cada archivo y graficar los valores de *fitness* máximo y promedio de la población a lo largo de cada generación. Posteriormente, se toma la distribución de turbinas asociada al individuo que ostenta el *fitness* máximo histórico de la población, y se genera la figura que muestra la colocación de turbinas sobre el terreno discretizado, según los ‘data.Nturb’ valores de posición que contenga dicho individuo. Todas las gráficas y figuras creadas por el *script* son guardadas en formato ‘.eps’ para su posterior uso en la memoria. Este *script* de procesado de resultados se encarga de ir almacenando los valores de *fitness* máximo de la población para cada modelo de estela, tamaño de terreno y tipología de viento, para posteriormente generar una tabla que incluya el rendimiento a través de la comparativa con los valores de energía anual generada en condiciones ideales (todas las turbinas sin solapamiento de estela) de cada modelo.

Capítulo 5

Experimentos y validación

Este capítulo cuenta con un total de 4 secciones, donde se describen los distintos experimentos que se van a realizar y se detallan los valores referenciales de ciertas constantes utilizadas para analizar las distintas soluciones conseguidas y verificar que sean correctas. En la Sección 5.1 se enumeran las hipótesis simplificadoras tomadas en cuenta en los experimentos, la Sección 5.2 recoge todos los datos utilizados en los experimentos, en la Sección 5.3 se detalla en qué consisten los experimentos a realizar y en la Sección 5.4 se exponen todos los parámetros y datos que se utilizan para comprobar que las soluciones se han calculado correctamente.

5.1. Hipótesis y simplificaciones

A continuación, se enumeran las 5 hipótesis principales que se han tomado en cuenta para la realización de los cálculos de disminución de velocidad sobre las turbinas debido a las estelas, y la energía anual generada por ellas.

- Las turbinas eólicas giran de forma que el vector de la corriente incidente siempre sea perpendicular al plano del rotor (ángulo de resbalamiento del ala nulo, $\beta = 0$).
- Todas las turbinas son del mismo modelo y con las mismas características geométricas.
- Se considera una tipología de terreno despejada sin desniveles ni obstáculos externos que interfieran en las estelas.
- Atmósfera en calma y sin cambios apreciables en la densidad del aire que afecten significativamente a la curva del coeficiente de tracción de las palas del rotor.
- Flujo incompresible y estacionario.

5.2. Datos usados para el cálculo experimental

Se han tomado, a modo de referencia, los parámetros y datos usados en [10]. En lo referente al vector de vientos real, se contaban con tres *datasets*, recogidos en ficheros ‘.mat’, que marcaban la velocidad y dirección del viento en una determinada zona durante 1, 2 y 3 años. Para que se analizaran el mismo número de horas en los dos tipos de vientos y reducir el tiempo de cálculo requerido para obtener el *fitness* de las soluciones, se decidió optar por el uso del vector de vientos correspondiente a 1 año, de tamaño 2×8760 , a pesar de que sea menos indicativo de las tendencias de viento de la zona.

De igual forma, con una rosa de los vientos con probabilidades establecidas en un total de 8 direcciones principales, definidas por los ángulos $[0^\circ, 45^\circ, 90^\circ, 135^\circ, 180^\circ, 225^\circ, 270^\circ, 315^\circ]$, y fijando la semilla de aleatoriedad en 1 para que genere siempre el mismo vector para todos los experimentos, formamos una nueva matriz del mismo tamaño. Los módulos aplicados a las direcciones normalizadas de los distintos vectores velocidad de esta nueva matriz son generados sintéticamente a través de una distribución de Weibull, cuyos parámetros de escala y forma fueron obtenidos de datos de medición reales obtenidos por un anemómetro. Para agilizar el proceso de cálculo y reducir el tiempo por iteración de los algoritmos metaheurísticos utilizados, se han discretizado en 128 valores acotados (*bins*), los módulos de la velocidad de dichas matrices de vectores viento, lo cual nos reduce el tamaño de la matriz en torno a un 30 % (elimina los valores menores a 4 m/s ya que no generan energía según la gráfica de potencia de la Figura 5.1).

Tabla 5.1: Parámetros de generación de los vectores velocidad sintéticos. Visto en [10].

Forma	Escala	Rosa de vientos (E,NE,N,NW,W,SW,S,SE)	Número de Muestras
1.8	7	(0.15, 0.05, 0.05, 0.05, 0.05, 0.1, 0.25, 0.3)	8760

Para la obtención de la energía generada en función de los módulos de la velocidad incidente que llegan a la turbina se ha hecho uso de la gráfica de potencia del aerogenerador AWEC-60, la cual fue extraída de [43]. Haciendo uso de la imagen de la gráfica que se encuentra en el interior del documento, se trasladaron los datos a MATLAB, a través de un ajuste por *splines* de la gráfica original, para su posterior uso en la última fase del cálculo de la función de *fitness*, cuyo funcionamiento se explica en la Sección 4.3 del Capítulo 4.

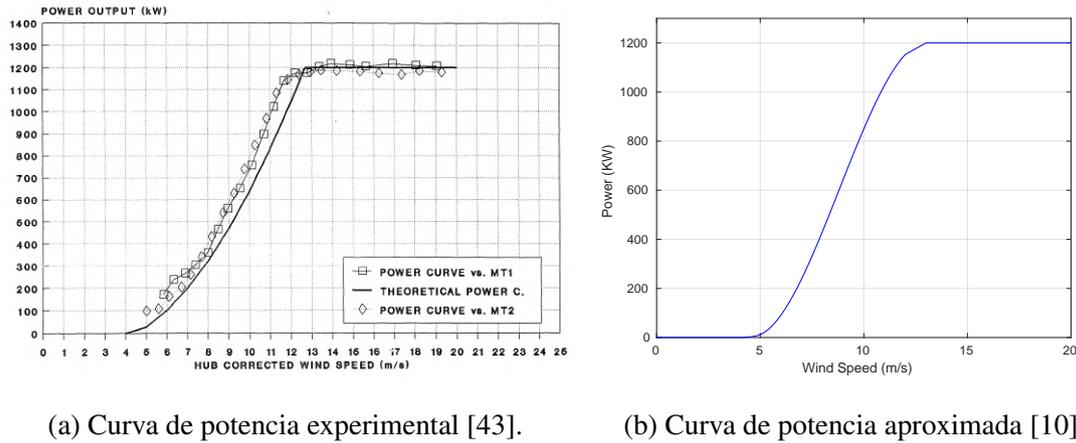


Figura 5.1: Curvas de potencia aerogenerador AWEC-60.

En la Tabla 5.2 se recogen los valores de las propiedades geométricas de las turbinas eólicas (altura de la turbina, diámetro del rotor, coeficiente de tracción, área del rotor, distancia de seguridad) y las características ambientales (rugosidad del terreno, intensidad de turbulencia estacionaria). Estos parámetros son usados posteriormente por los modelos de estela para realizar el cálculo de la pérdida de velocidad que sufre una turbina que se encuentra afectada por la estela de otra. Los valores geométricos de las turbinas eólicas son equivalentes a los mostrados en [9], sin embargo, en este estudio utilizan una gráfica de potencia distinta que la usada en este trabajo, representada en la Figura 5.1. La turbina AWEC-60 escogida para elaborar nuestra gráfica de potencia cuenta con una altura al buje de 60 m y diámetro de rotor de 60 m, sin embargo, se optó por tomar el diámetro del rotor de 80 m para jugar en las mismas condiciones que en la publicación mencionada. La elección del diámetro del rotor es de suma importancia en la delimitación de estela turbulenta que generan las turbinas, afectando a un mayor o menor número de turbinas aguas abajo, por lo que se llegó a la conclusión de que sería de interés mantenerlo para tomar como referencia ciertas magnitudes en los pasos de validación. En [10], además de tomar las mismas características geométricas de turbinas que las recogidas en la Tabla 5.2, usan la misma gráfica de potencia que la mostrada en la Figura 5.1, por lo que será más fácil la comparación de resultados, ya que nos permitirá comparar también magnitudes de energía producida.

Tabla 5.2: Características del entorno y geométricas de las turbinas.

Z (m)	D_{rotor} (m)	c_T	A_r (m ²)	Distancia seguridad (m)	z_0 (m)	I_0
60	80	0.88	$5.0265 \cdot 10^3$	400	0.3	0.035

5.3. Metodología de experimentos

5.3.1. Experimentos de optimización de posicionamiento de turbinas

Inicialmente, para cada algoritmo metaheurístico usado, será necesario realizar 4 optimizaciones de posicionamiento de turbinas eólicas por modelo de estela, en terrenos discreto de proporciones de 20×20 y 50×50 con 20 y 50 aerogeneradores respectivamente, según el tipo de vector viento utilizado (generación de vector de forma sintética o a través de datos reales). Para la generación sintética del vector viento, se tomará una rosa de los vientos de 8 ángulos y se generarán de forma probabilística, siguiendo la distribución de Weibull obtenida, los módulos de la velocidad asociados a dichas direcciones.

Se han propuesto 3 algoritmos metaheurísticos distintos para la comparativa de resultados de energía generada según el modelo de estela seleccionado y sus resultados finales a la hora de posicionar las turbinas. Los 3 algoritmos escogidos son el Algoritmo Genético Básico (BGA), el *Particle Swarm Optimization* (PSO) y el *Coral Reefs Optimization with Substrate Layers* (CRO-SL).

5.3.2. Experimentos para la selección de mejores soluciones

Tras obtener las distribuciones de turbinas sobre el terreno, procedemos al análisis cruzado de soluciones entre modelos, para cada uno de los distintos algoritmos genéticos utilizados, con el fin de averiguar qué modelo de estela realiza mejor la tarea de optimización de colocación de turbinas eólicas para cada uno de los tres algoritmos metaheurísticos utilizados en el estudio. Para cada uno de los ellos, dispondremos de 4 soluciones óptimas (20×20 sintético, 20×20 real, 50×50 sintético y 50×50 real) por cada uno de los 7 modelos de estela usados (28 configuraciones de turbinas por algoritmo metaheurístico). El cálculo de energía anual se realizará bajo las mismas condiciones de distribuciones de vientos que se impusieron durante el proceso de optimización, por lo que inicialmente se dispondrá ya de 84 resultados de los 588 análisis cruzados necesarios.

Sea $\{WM\}$ el conjunto de modelos de estela usados en el cálculo de la función de *fitness* y $\{S\}$ el conjunto de todas las soluciones obtenidas para un número determinado de turbinas y matriz de vientos, ambos con 7 elementos cada uno. Se dispone de un total de 12 conjuntos $\{S\}$ distintos tras los experimentos de optimización, por lo que realizando el producto cartesiano de conjuntos de $\{WM\}$ con los distintos conjuntos $\{S\}$, $\{WM\} \times \{S\}_i$, se obtienen 4 conjun-

tos cruzados por cada uno de los 3 algoritmos de optimización planteados, lo cual nos deja 12 conjuntos de 49 elementos. Siendo cada uno de los 12 conjuntos, las soluciones cruzadas pertenecientes a las diferentes combinaciones planteadas de terreno, número de turbinas, modelo de estela y algoritmos metaheurísticos. En la Figura 5.2 se representa el producto cartesiano del conjunto de los modelos de estela con uno de los conjuntos de soluciones obtenidas del proceso de optimización a través de un algoritmo metaheurístico para la configuración de parámetros seleccionados.

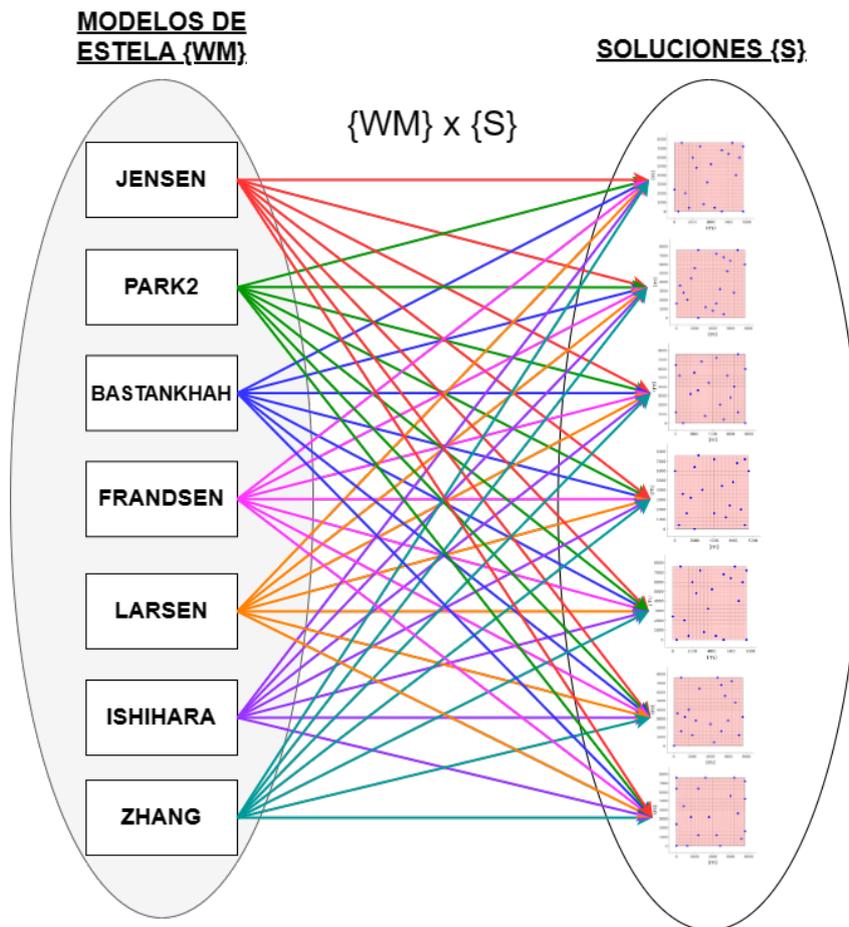


Figura 5.2: Producto cartesiano de conjuntos $\{WM\}$ con $\{S\}$ para obtención de soluciones cruzadas.

5.4. Metodología de validación de procesos y resultados

5.4.1. Validación de tiempos de ejecución de los modelos de estela

Para realizar una comparativa de los tiempos de computación, así como de las magnitudes de los resultados y la diferencia de valores de energía generada entre cada uno de los modelos de estela, se ha creado una tabla comparativa con cada uno de los 7 modelos de estela sobre terrenos discretizados en *layouts* de dimensiones 20×20 con 20 turbinas y 50×50 con 50 turbinas. Se realiza una media entre 20 iteraciones del mismo modelo sobre la misma distribución de turbinas y vientos. Con estos datos podemos estimar el tiempo aproximado de cálculo que nos llevará realizar el total de experimentos en un determinado dispositivo (los tiempos de computación variarán en función de las especificaciones técnicas de la máquina que realice los cálculos). También, nos sirve para observar las diferencias de tiempo en la obtención de resultados entre los distintos modelos de estela programados y estimar un número determinado de iteraciones razonable para su uso en el proceso de optimización mediante algoritmos metaheurísticos. Se le introduce de manera directa el vector de vientos para los 8 ángulos que recoge la rosa de vientos, mencionados en el Capítulo 5.3, con un módulo de 7 m/s, es decir, lo que nos deja un rango temporal de análisis de generación de energía de 160 horas entre todas las iteraciones. Finalmente se hace el promedio de tiempo de procesamiento por hora de producción de las turbinas eólica analizada, dividiendo el tiempo total de cálculo entre el número de velocidades que contiene el vector viento y el número de iteraciones realizadas. Los resultados de los tiempos de ejecución para ambas configuraciones pueden verse en la Tabla 5.3.

Para estimar el número de iteraciones factibles para los análisis, es necesario tomar la referencia de mayor tiempo de ejecución, en este caso la configuración de 50 turbinas en un terreno de proporción 50×50 . Como podemos observar en la Tabla 5.3, los tiempos de ejecución son suficientemente bajos, como para aplicar una metodología de optimización iterativa por procesos metaheurísticos poblacionales con una cantidad de individuos considerable. Estos cálculos realizados de generación de energía por las turbinas a lo largo de un periodo de tiempo determinado, correspondiente a la función de *fitness* que se busca maximizar, deberán repetirse en múltiples ocasiones a lo largo del proceso iterativo de mejora de individuos de la población. Analizando las cotas temporales que podemos aproximar con estos datos aplicados a cada uno de los algoritmos metaheurísticos programados, se ha decidido establecer en 1000 el número total de iteraciones (llamadas generaciones en los algoritmos Genético Básico y CRO-SL).

Tabla 5.3: Tiempos de ejecución de los modelos de estela y proporciones del terreno.

Terreno 20×20 con 20 turbinas		
Modelo	Tiempo ejecución total (s)	Tiempo ejecución promedio (s)
Jensen	12.9105	0.6455
Park2	11.2677	0.5633
Bastankhah	12.8388	0.6419
Frandsen	14.7392	0.7369
Larsen	15.3227	0.7661
Ishihara	12.3965	0.6198
Zhang	13.4323	0.6716
Terreno 50×50 con 50 turbinas		
Modelo	Tiempo ejecución total (s)	Tiempo ejecución promedio (s)
Jensen	30.9483	0.6189
Park2	30.4420	0.6088
Bastankhah	41.5677	0.8313
Frandsen	35.0018	0.7000
Larsen	46.9688	0.9393
Ishihara	38.3057	0.7661
Zhang	40.6227	0.8124

5.4.2. Validación de las magnitudes de energía de las soluciones

Con el fin de percatarnos del correcto funcionamiento del código antes de su utilización con los diferentes algoritmos metaheurísticos diseñados, se han elaborado una serie de pruebas que verifiquen si las modificaciones e implementaciones realizadas en los *scripts* funcionan de forma adecuada y dan resultados realistas. En total se han programado un total de 7 modelos de estela para el análisis, cada uno de ellos con unas características particulares de superposición y de límites de estela, que perjudicará en mayor o menor medida a la corriente incidente sobre cada una de las turbinas del terreno.

Se debe marcar los umbrales máximos y mínimos de energía generada por cada uno de los modelos para las distribuciones de vientos con las que se van a llevar a cabo los distintos

análisis; se han realizado cuatro tipologías distintas de pruebas para cada uno de los modelos de estela. En cuanto a posicionamiento de las turbinas, se estudia una configuración ideal en paralelo, en la que las estelas de las turbinas no se solapan en ninguna ocasión, por lo que siempre verán la corriente incidente sin perturbar, y una colocación en fila en la que el conjunto de turbinas se vea sometido al mayor solapamiento de estelas posible.

Tabla 5.4: Energía máxima y mínima generadas anualmente por 20 turbinas eólicas.

Viento sintético		
Modelo	Energía máxima (kWh)	Energía mínima (kWh)
Jensen	$5.1283 \cdot 10^7$	$2.7156 \cdot 10^7$
Park2	$5.1283 \cdot 10^7$	$1.4337 \cdot 10^7$
Bastankhah	$5.1283 \cdot 10^7$	$0.8298 \cdot 10^7$
Frandsen	$5.1283 \cdot 10^7$	$0.6773 \cdot 10^7$
Larsen	$5.1283 \cdot 10^7$	$0.9325 \cdot 10^7$
Ishihara	$5.1283 \cdot 10^7$	$0.3334 \cdot 10^7$
Zhang	$5.1283 \cdot 10^7$	$1.6105 \cdot 10^7$
Viento real		
Modelo	Energía máxima (kWh)	Energía mínima (kWh)
Jensen	$5.2043 \cdot 10^7$	$2.7733 \cdot 10^7$
Park2	$5.2043 \cdot 10^7$	$1.4704 \cdot 10^7$
Bastankhah	$5.2043 \cdot 10^7$	$0.8557 \cdot 10^7$
Frandsen	$5.2043 \cdot 10^7$	$0.6936 \cdot 10^7$
Larsen	$5.2043 \cdot 10^7$	$0.9535 \cdot 10^7$
Ishihara	$5.2043 \cdot 10^7$	$0.3396 \cdot 10^7$
Zhang	$5.2043 \cdot 10^7$	$1.6509 \cdot 10^7$

Realizando una comparativa de valores entre las energías máximas que se pueden llegar a obtener entre una distribución de tipo sintética a través distribución de Weibull, cuyos parámetros se pueden revisar en la Tabla 5.1, y una distribución de vientos real recogidos de una determinada zona, se observa que en el caso de 20×20 , la energía generada en el mismo rango temporal para las dos distribuciones es aproximadamente un 1.5 % superior en el caso real respecto al sintético. Esto nos hace ver que los módulos son mayores para el registro de vientos

reales, que la distribución generada aleatoriamente por una distribución de Weibull y con una semilla generadora de números aleatorios (*random number generator* - 'rng()') inicializada en un valor unitario. En caso de modificar esta semilla, también lo harán los vientos generados de forma aleatoria, pudiendo alterar esta tendencia.

Tabla 5.5: Energía máxima y mínima generadas anualmente por 50 turbinas eólicas.

Viento sintético		
Modelo	Energía máxima (kWh)	Energía mínima (kWh)
Jensen	$1.2821 \cdot 10^8$	$6.5516 \cdot 10^7$
Park2	$1.2821 \cdot 10^8$	$2.3519 \cdot 10^7$
Bastankhah	$1.2821 \cdot 10^8$	$1.6722 \cdot 10^7$
Frandsen	$1.2821 \cdot 10^8$	$0.7231 \cdot 10^7$
Larsen	$1.2821 \cdot 10^8$	$0.9325 \cdot 10^7$
Ishihara	$1.2821 \cdot 10^8$	$0.3334 \cdot 10^7$
Zhang	$1.2821 \cdot 10^8$	$2.6896 \cdot 10^7$
Viento real		
Modelo	Energía máxima (kWh)	Energía mínima (kWh)
Jensen	$1.3011 \cdot 10^8$	$6.6939 \cdot 10^7$
Park2	$1.3011 \cdot 10^8$	$2.4213 \cdot 10^7$
Bastankhah	$1.3011 \cdot 10^8$	$1.7285 \cdot 10^7$
Frandsen	$1.3011 \cdot 10^8$	$0.7411 \cdot 10^7$
Larsen	$1.3011 \cdot 10^8$	$0.9535 \cdot 10^7$
Ishihara	$1.3011 \cdot 10^8$	$0.3397 \cdot 10^7$
Zhang	$1.3011 \cdot 10^8$	$2.7670 \cdot 10^7$

La energía máxima generada en función del tipo de viento establecido para terrenos de un mismo tamaño y con un mismo número de turbinas puede variar notoriamente. La diferencia de energía máxima entre las dos distribuciones de viento en terrenos de 50×50 es de $1.9 \cdot 10^6$ kWh, mientras que para las distribuciones mencionadas en terrenos de 20×20 es de $7.6 \cdot 10^5$ kWh, esto hace que, para ambos terrenos, la diferencia entre el viento sintético y el real se sitúe en $3.8 \cdot 10^3$ kWh/turbina. Observando las energías mínimas generadas por los modelos, se puede concluir que el modelo de estela de Ishihara es el más restrictivo a la hora de superponer las estelas y que

lleva a una mayor penalización de las turbinas que se encuentran afectadas por estelas cercanas, prediciendo únicamente alrededor del 1.2-5 % de la energía generada que se obtendría con el uso del modelo Jensen, el cual sería el menos limitativo en este caso.

Dado que en el artículo [10] tomado de referencia, se compara el método de optimización a través de la generación de números pseudorandom (propuesto en [9]) con la optimización usando el algoritmo CRO-SL, y se han hecho análisis de índole similar con los mismos datos de vientos, dimensiones de terreno, número de turbinas y curvas de potencia, podemos usar estos resultados para comparar que nuestras soluciones se encuentren cercanas a los mismos rangos de magnitud que muestran. En ese estudio, obtienen valores de energía anual generada respectivamente tras el proceso de optimización, usando el modelo de estela Jensen, de $5.2900 \cdot 10^7$ - $5.3116 \cdot 10^7$ kWh para un terreno de dimensiones 20×20 con 20 turbinas posicionadas y de $1.3197 \cdot 10^8$ - $1.3273 \cdot 10^8$ kWh para el caso de 50 turbinas posicionadas en un *layout* de 50×50 . También estiman unos valores para el modelo Bastankhah de $5.2975 \cdot 10^7$ kWh en distribuciones de 20 turbinas sobre un terreno de 20×20 y de $1.2991 \cdot 10^8$ kWh para 50 turbinas en un terreno de 50×50 . Mencionar que el número de iteraciones que se han ejecutado en el proceso de optimización algorítmica del estudio mencionado es 10 veces mayor que el que número de iteraciones que se han ejecutado en este trabajo (establecidas en la Sección 5.4.1), por lo que el valor de energía generada que obtengamos en los experimentos tenderá a ser menor debido a que se ha realizado una inversión menor de iteraciones en mejorar el posicionamiento de las turbinas y a que la matriz de vientos utilizada tiene distintas direcciones y magnitudes.

Capítulo 6

Resultados y comparativas

En la Sección 5.3.2 del Capítulo 5, se han descrito los distintos experimentos a realizar para mejorar la colocación de las turbinas sobre un terreno discreto de geometría cuadrada/rectangular. Se han realizado 84 experimentos de optimización por algoritmia metaheurística, los cuales han arrojado resultados de energía anual generada comprendidos entre los valores discutidos en la Sección 5.4.2 del Capítulo 5. Se observa gran variabilidad de los resultados en función de la matriz de vientos utilizada, siempre siendo notablemente inferior la energía calculada a través de la matriz de vientos sintéticos. De igual forma, otros factores como el algoritmo metaheurístico por el que se haya realizado la optimización y el modelo de estela también inciden de manera muy considerable en la magnitud de las soluciones.

Se ha recogido la energía anual generada para los distintos modelos de estela en función del algoritmo metaheurístico utilizado, el tipo de matriz de vectores viento generada, el tamaño de terreno y número de turbinas. Gracias a los datos de energía máxima y mínima, se añade a las distintas tablas de resultados una última columna que marca el rendimiento de producción respecto a una colocación ideal en la que no se superpongan estelas, cuyos valores se recogieron en las Tablas 5.4 y 5.5.

6.1. Valores de energía generados y rendimiento

Podemos observar que, entre los distintos algoritmos metaheurísticos diseñados, se mantiene la misma tendencia de comportamiento y crecimiento de la función de *fitness* independientemente del modelo de estela que se haya utilizado para el proceso de mejora de la colocación de las turbinas eólicas. Cada uno de los tres algoritmos tiene una metodología propia e hiper-

parámetros que definen de forma característica su conducta de optimización y su capacidad de obtener mejores o peores resultados en un número determinado de iteraciones.

Como era de esperar, dado que hay mayor cantidad de turbinas eólicas en funcionamiento, los valores de energía obtenidos para las distribuciones de 50×50 con 50 turbinas son superiores a los realizados para el caso de terreno de tamaño 20×20 con 20 turbinas. El hecho de contar con un número superior de turbinas y una cantidad mayor de espacios de colocación (2500 posiciones disponibles para el caso de 50×50 en comparación con las 400 que había en el caso de 20×20), provoca que las turbinas se puedan separar y posicionar de forma que exista una distancia mayor entre ellas, lo cual reduciría notablemente el efecto de la estela sobre las turbinas, reduciendo así el déficit de velocidad sobre el módulo de la corriente incidente y aumentando la energía generada. Haciendo uso de la Ecuación (6.1) se puede calcular la tasa de rendimiento de las soluciones, donde E_{max} y E_{min} corresponden a los valores de energía máxima y mínima de cada modelo que se pueden encontrar en las Tablas 5.4 y 5.5, y E_{gen} es la energía anual generada por la distribución resultante del proceso de optimización.

$$Ren(\%) = \frac{E_{gen} - E_{min}}{E_{max} - E_{min}} \cdot 100 \quad (6.1)$$

En las Tablas 6.1 y 6.2 se pueden ver los valores correspondientes a la optimización de posicionamiento realizada mediante el BGA para el caso sintético y real, respectivamente. Las tasas de rendimiento y energía generada son bastante altas en la mayoría de los casos, viendo como las predicciones más bajas corresponden al modelo de Larsen para ambas matrices de viento usadas, rondando el 92-93 %, aunque para la configuración de 20×20 con vientos sintéticos consigue una solución con un rendimiento de 97.45 %, el cual es un valor que está más cercano a la media de rendimiento que consiguen el resto de los modelos. El modelo Zhang es el segundo modelo con resultados de energía generada más bajos, hecho que choca en cierta medida con las conclusiones extraídas de las Tablas 5.4 y 5.5 del Capítulo 5, que establecían al modelo de Ishihara como más restrictivo que el modelo de Zhang a la hora de predecir la energía generada. Esto puede deberse a que el modelo de Ishihara perjudica, en mayor medida que las turbinas se encuentren posicionadas a distancias cortas unas de otras y que se superpongan un gran número de estelas en el punto de análisis, pero se atenúe este efecto cuando aumente la distancia y no exista tanto solapamiento de estelas.

Los datos de energía obtenidos por los modelos Jensen y Park2 se encuentran muy cercanos, siendo los del modelo Park2 levemente superiores. Como se explicó en la Sección 3.2.2 del

Capítulo 3, este modelo es una modificación directa y que cuenta con un gran parecido con el modelo Jensen, por lo que era esperable este tipo de similitud en los resultados. Los modelos de Bastankhah y Frandsen muestran resultados bastante estables para todas las configuraciones, con valores de rendimiento que se sitúan en torno al 99 % y 99.5 % respectivamente.

Tabla 6.1: Resultados BGA con distribución de vientos sintética.

Modelo	20×20 con 20 turbinas		50×50 con 50 turbinas	
	Energía (kWh)	Rendimiento (%)	Energía (kWh)	Rendimiento (%)
Jensen	5.1161·10⁷	99.49	1.2777·10⁸	99.27
Park2	5.1163·10⁷	99.68	1.2782·10⁸	99.59
Bastankhah	5.0723·10⁷	98.94	1.2649·10 ⁸	98.81
Frandsen	5.1135·10⁷	99.67	1.2758·10 ⁸	99.41
Larsen	5.0213·10⁷	97.45	1.2325·10 ⁸	92.73
Ishihara	5.1154·10⁷	99.73	1.2781·10⁸	99.63
Zhang	4.9933·10⁷	96.16	1.2591·10 ⁸	97.40

Tabla 6.2: Resultados BGA con distribución de vientos real.

Modelo	20×20 con 20 turbinas		50×50 con 50 turbinas	
	Energía (kWh)	Rendimiento (%)	Energía (kWh)	Rendimiento (%)
Jensen	5.1895·10⁷	99.40	1.2964·10 ⁸	99.24
Park2	5.1940·10⁷	99.73	1.2964·10 ⁸	99.51
Bastankhah	5.1572·10⁷	99.17	1.2846·10⁸	98.93
Frandsen	5.1892·10⁷	99.67	1.2954·10 ⁸	99.48
Larsen	4.7814·10 ⁷	92.02	1.2469·10 ⁸	93.19
Ishihara	5.1890·10⁷	99.69	1.2962·10⁸	99.58
Zhang	5.0774·10⁷	96.43	1.2759·10 ⁸	97.27

El segundo algoritmo utilizado es el PSO, cuyos resultados de energía anual generada para los vientos sintéticos y reales quedan recogidos en las Tablas 6.3 y 6.4, respectivamente. Los resultados del algoritmo PSO marcan cotas de energía generadas inferiores a las obtenidas mediante el BGA pese a que es necesario la inversión de un tiempo mayor por iteración, haciendo que este algoritmo sea menos eficiente en todos los aspectos analizados. Los algoritmos PSO requieren de un estudio muy detallado y un ajuste de los hiperparámetros que definen su funcio-

namiento. Tras realizar los experimentos asociados a este algoritmo, se ha concluido que, para su uso en la resolución del problema planteado, dado que es necesario evaluar distintos tamaños de terrenos y número de turbinas, sería de interés establecer los hiperparámetros asociados a las aceleraciones, como valores porcentuales de las proporciones del terreno para limitar de manera efectiva el movimiento de las partículas, y no como valores constantes.

Tabla 6.3: Resultados PSO con distribución de vientos sintética.

Modelo	20×20 con 20 turbinas		50×50 con 50 turbinas	
	Energía (kWh)	Rendimiento (%)	Energía (kWh)	Rendimiento (%)
Jensen	$4.9921 \cdot 10^7$	94.36	$1.2666 \cdot 10^8$	97.53
Park2	$4.9985 \cdot 10^7$	96.49	$1.2593 \cdot 10^8$	97.82
Bastankhah	$4.7489 \cdot 10^7$	91.17	$1.2223 \cdot 10^8$	94.64
Frandsen	$4.9333 \cdot 10^7$	95.62	$1.2509 \cdot 10^8$	97.42
Larsen	$4.7022 \cdot 10^7$	89.84	$1.1665 \cdot 10^8$	90.28
Ishihara	$4.9593 \cdot 10^7$	96.48	$1.2452 \cdot 10^8$	97.05
Zhang	$4.874 \cdot 10^7$	92.77	$1.2404 \cdot 10^8$	95.89

Tabla 6.4: Resultados PSO con distribución de vientos real.

Modelo	20×20 con 20 turbinas		50×50 con 50 turbinas	
	Energía (kWh)	Rendimiento (%)	Energía (kWh)	Rendimiento (%)
Jensen	$5.0032 \cdot 10^7$	91.73	$1.2814 \cdot 10^8$	96.88
Park2	$5.0776 \cdot 10^7$	96.61	$1.2731 \cdot 10^8$	97.36
Bastankhah	$4.8991 \cdot 10^7$	92.98	$1.2333 \cdot 10^8$	93.99
Frandsen	$5.0714 \cdot 10^7$	97.05	$1.2716 \cdot 10^8$	97.60
Larsen	$4.7555 \cdot 10^7$	89.44	$1.1701 \cdot 10^8$	89.13
Ishihara	$4.9015 \cdot 10^7$	93.78	$1.2653 \cdot 10^8$	97.17
Zhang	$4.9941 \cdot 10^7$	94.08	$1.2592 \cdot 10^8$	95.91

Uno de los principales inconvenientes de la metodología del CRO-SL es que la obtención de las soluciones se demora más tiempo que con los otros dos algoritmos desarrollados. El tiempo de resolución del proceso de optimización puede variar en función de la cantidad de población asignada a cada sustrato y cuál vaya consiguiendo mayor predominancia conforme avance el proceso.

Las Tablas 6.5 y 6.6 recogen los valores de energía producida por la distribución estimada a través del algoritmo CRO-SL, tanto en viento sintético como real respectivamente, donde se puede observar de nuevo valores de rendimiento muy elevados, todos ellos superiores al 93 %. El rendimiento de optimización por modelo conseguido por el algoritmo CRO-SL se encuentra muy cercano al obtenido por el BGA, superando a éste en ciertos modelos de cada configuración analizada. Un comportamiento destacable de este algoritmo es el hecho de que obtiene mejores soluciones para configuraciones de terreno de grandes extensiones, ya que únicamente supera al BGA en un caso, usando el modelo de estela Larsen en la configuración de 20×20 con viento real.

A pesar de estos casos concretos, el BGA muestra mejores rendimientos en la mayoría de los modelos, y los obtiene en una cantidad de tiempo muy inferior a la requerida por el CRO-SL dado el mismo número de generaciones poblacionales.

Tabla 6.5: Resultados CRO-SL con distribución de vientos sintética.

Modelo	20×20 con 20 turbinas		50×50 con 50 turbinas	
	Energía (kWh)	Rendimiento (%)	Energía (kWh)	Rendimiento (%)
Jensen	$5.1076 \cdot 10^7$	99.14	$1.2771 \cdot 10^8$	99.20
Park2	$5.1122 \cdot 10^7$	99.57	$1.2772 \cdot 10^8$	99.54
Bastankhah	$5.0666 \cdot 10^7$	98.56	$1.2723 \cdot 10^8$	99.12
Frandsen	$5.1002 \cdot 10^7$	99.37	$1.2766 \cdot 10^8$	99.55
Larsen	$4.9699 \cdot 10^7$	96.23	$1.2127 \cdot 10^8$	94.16
Ishihara	$5.0931 \cdot 10^7$	99.26	$1.2773 \cdot 10^8$	99.62
Zhang	$4.9900 \cdot 10^7$	96.07	$1.2561 \cdot 10^8$	97.43

Tabla 6.6: Resultados CRO-SL con distribución de vientos real.

Modelo	20×20 con 20 turbinas		50×50 con 50 turbinas	
	Energía (kWh)	Rendimiento (%)	Energía (kWh)	Rendimiento (%)
Jensen	$5.1846 \cdot 10^7$	99.19	$1.2967 \cdot 10^8$	99.31
Park2	$5.1904 \cdot 10^7$	99.63	$1.2966 \cdot 10^8$	99.58
Bastankhah	$5.1603 \cdot 10^7$	98.99	$1.2889 \cdot 10^8$	98.91
Frandsen	$5.1831 \cdot 10^7$	99.53	$1.2955 \cdot 10^8$	99.54
Larsen	$5.0278 \cdot 10^7$	95.85	$1.2225 \cdot 10^8$	93.48
Ishihara	$5.1806 \cdot 10^7$	99.51	$1.2954 \cdot 10^8$	99.55
Zhang	$5.0668 \cdot 10^7$	96.13	$1.2759 \cdot 10^8$	97.54

6.2. Evolución de las soluciones de la población

Las Figuras 6.1 y 6.2 muestran la evolución temporal de las iteraciones del BGA para el caso de vientos reales en los terrenos 20×20 y 50×50, respectivamente. En el Apéndice E se encuentran las Figuras E.1 y E.2 correspondientes a los vientos generados sintéticamente.

Las gráficas mostradas en la Figura 6.2 muestran un salto muy acentuado en una generación determinada, lo cual provoca una subida muy notable en la magnitud de los valores de *fitness* máximo y medio de la población. En la mayoría de las gráficas vemos como este punto marca una separación entre dos zonas de bajo ritmo de crecimiento de producción energía, que prácticamente se podrían considerar zonas estacionalidad. Dado que los valores de energía producida que se recogen en las gráficas de la Figura 6.1 son de menor magnitud que las observadas en la Figura 6.2, se puede percibir una pendiente más progresiva y acentuada, con incrementos de energía generada menos puntuales, debido a que muestra un ritmo más paulatino y constante.

En la mayoría de las gráficas, salvo en casos puntuales como la Figura 6.2d, se observa que el crecimiento más importante se da siempre antes de llegar a la generación 400. Este comportamiento se debe al uso del operador de cruce representado en la Figura 4.3. Este modo de combinación de genes de individuos de la población para producir descendencia tiene un carácter bastante conservador. A pesar de establecer una probabilidad de mutación relativamente alta en comparación con los parámetros recomendados, los individuos de cada generación tenderán a ir adquiriendo una gran cantidad de genes comunes y producir valores de *fitness* similares. Esto implica un descenso en el ritmo de crecimiento poblacional, ya que se irán asentando individuos de altos valores de *fitness* que serán difíciles de sustituir por los nuevos descendientes.

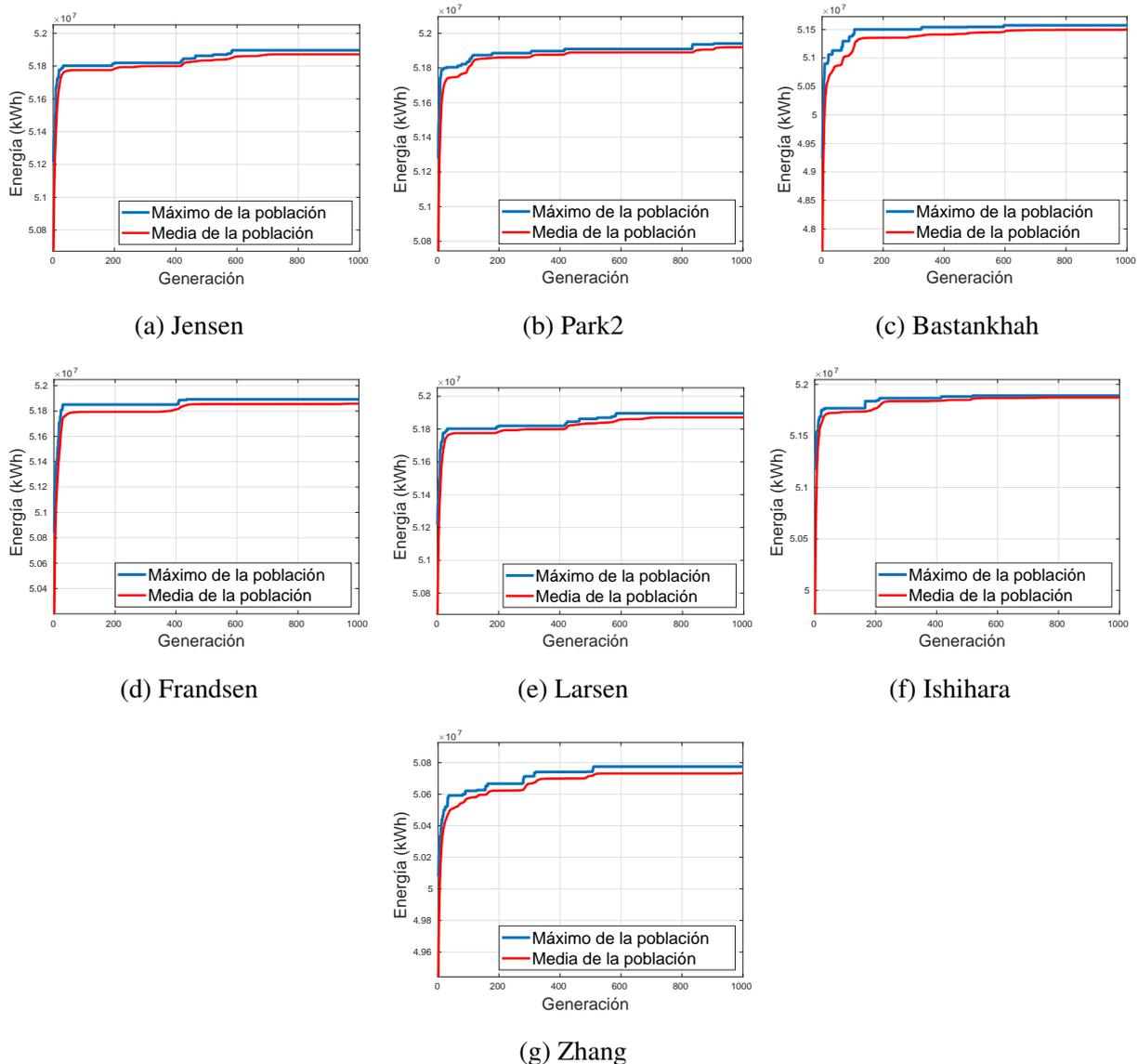


Figura 6.1: Resultados Algoritmo Genético Básico para terreno de 20×20 con viento real.

En las Figuras 6.3 y 6.4 se puede ver la tendencia generalizada de la evolución del valor de la función de *fitness*, la cual muestra un crecimiento muy acentuado en iteraciones concretas para el *fitness* máximo poblacional y una tendencia oscilante acotada para el *fitness* medio poblacional. El PSO se distingue de los otros dos algoritmos usados en que todos los individuos modifican su estado de una iteración a otra, por lo que el *fitness* medio de la población está en constante oscilación y no siempre en crecimiento. Sin embargo, el *fitness* máximo no muestra una tendencia creciente conforme avanza la ejecución, a pesar de que las subidas de valores se realicen de forma brusca.

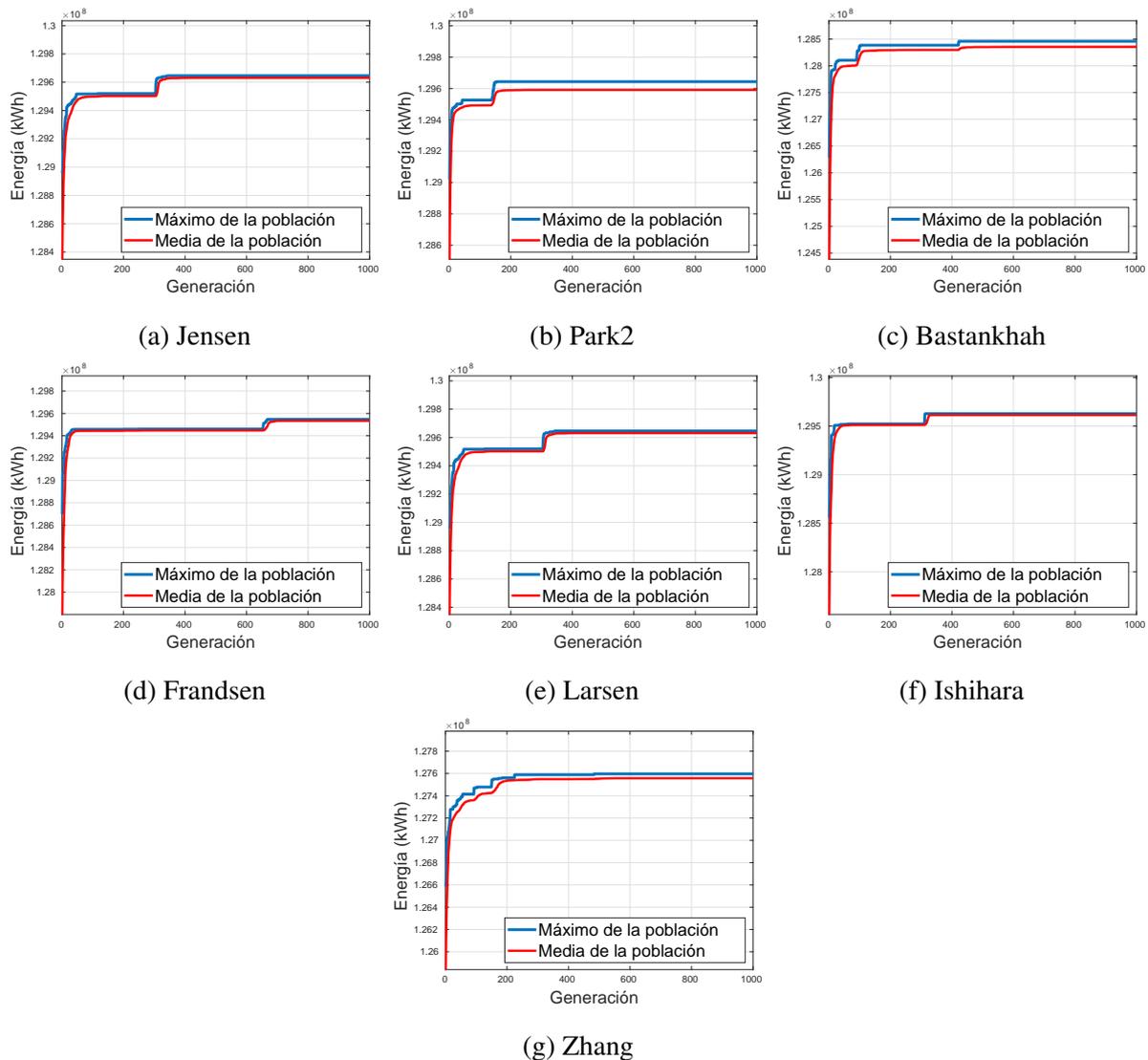


Figura 6.2: Resultados Algoritmo Genético Básico para terreno de 50×50 con viento real.

Por lo general, en las primeras versiones del algoritmo desarrolladas, una subida en el *fitness* máximo de la población generaba un cambio en los puntos de referencia del movimiento de las partículas, haciendo que, de igual forma que aumenta el valor máximo, los valores medios poblacionales también tendrán una subida generalizada bastante notoria, la cual se mantendrá, independientemente del margen oscilatorio. Sin embargo, en su forma básica, los valores de energía alcanzados eran más bajos que los obtenidos en la nueva versión, ya que el crecimiento de la función de *fitness* se estancaba rápidamente, por lo que se optó por introducir la metodología de permutación de partículas entre grupos explicada en la Sección 4.6 del Capítulo 4. Por la tendencia observada, se deduce que la probabilidad de permutación entre grupos de las partículas asignadas a una determinada solución es demasiado elevada, y provoca grandes niveles de aleatoriedad, centrándose menos en la explotación de las posiciones referenciales.

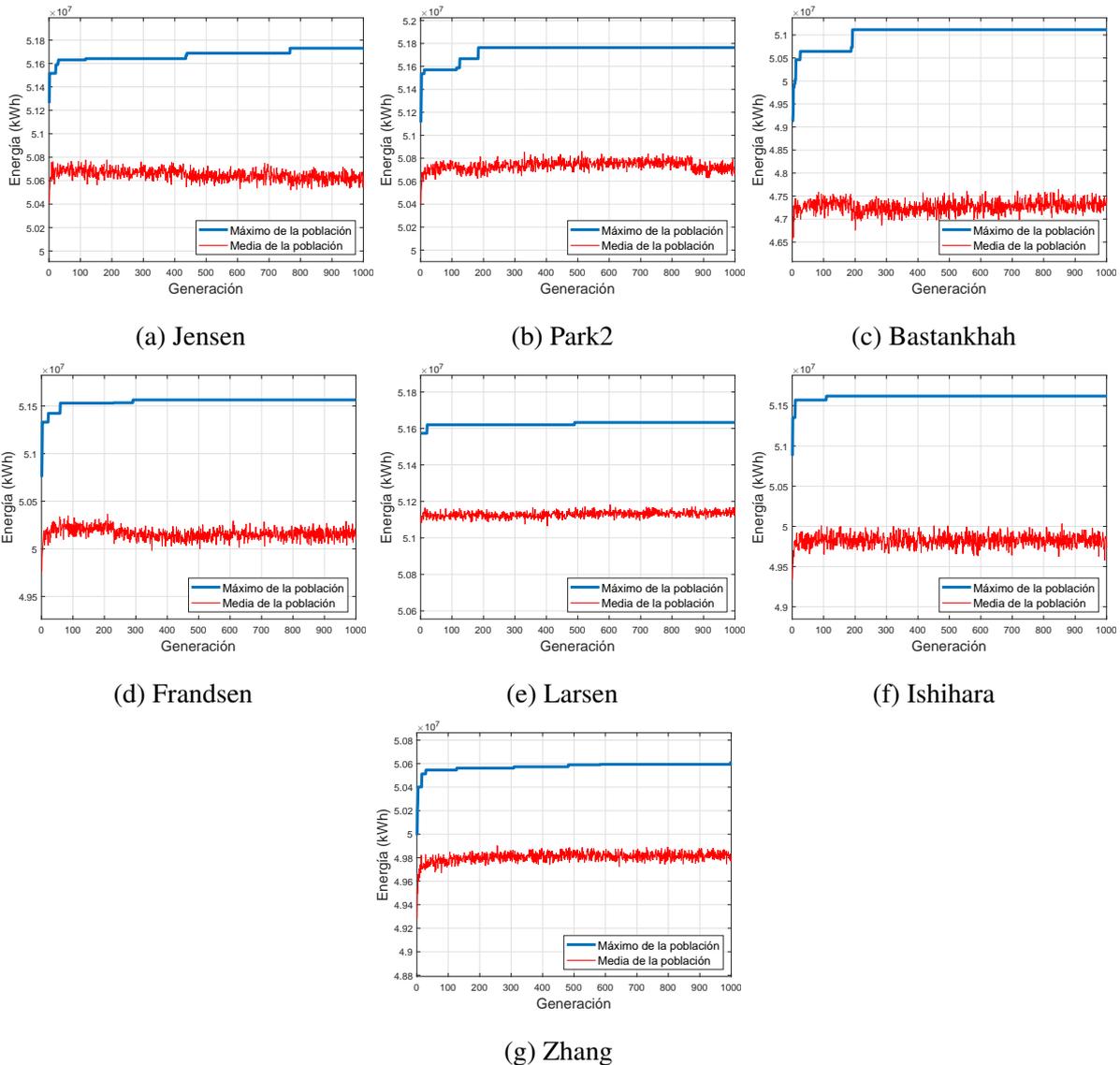


Figura 6.3: Resultados *Particle Swarm Optimization* para terreno de 20×20 con viento real.

En conclusión, se han establecido unos valores de hiperparámetros que fomentan la exploración de nuevas soluciones. El hecho de que no se alcancen valores de *fitness* superiores, demuestra que sería necesario incrementar la magnitud asociada a la inercia de la partícula para evitar la convergencia de partículas. Otro punto importante a analizar sería la probabilidad establecida de intercambio de partículas entre grupos, ya que es un proceso que permite escapar fácilmente de la convergencia de soluciones sin necesidad de ajustar otros hiperparámetros. La metodología usada por el PSO guarda una gran capacidad a la hora de optimizar el problema de colocación a través de matrices, pero es necesario un estudio detallado de los valores asignados a la inercia y aceleraciones para lograr explotar su potencial al máximo. Se pueden consultar las gráficas de la evolución de las soluciones vientos sintéticos en las Figuras E.3 y E.4 del

Apéndice E.

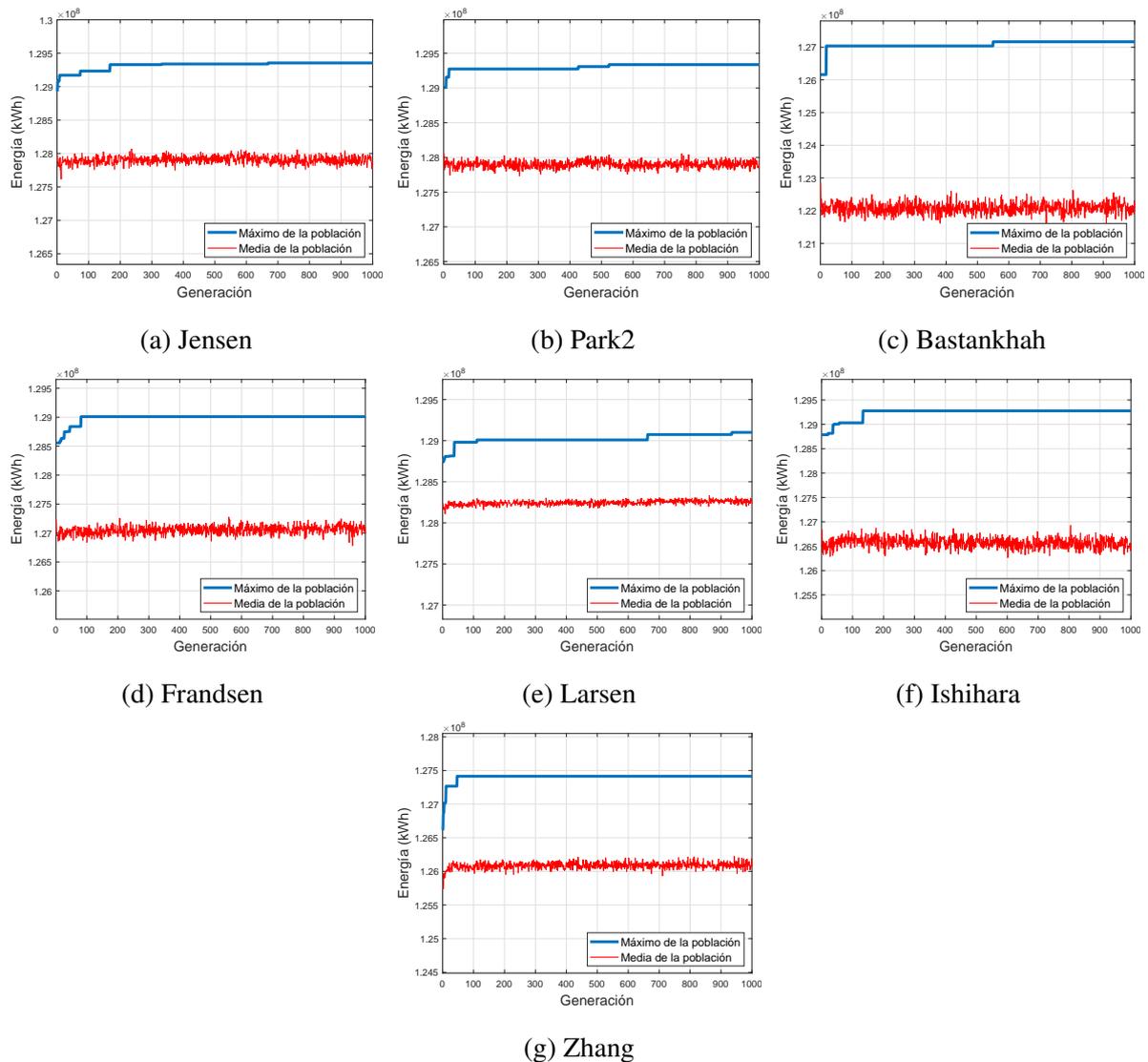


Figura 6.4: Resultados *Particle Swarm Optimization* para terreno de 50×50 con viento real.

En lo referente a la tendencia evolutiva de la función de *fitness* del algoritmo CRO-SL, representada en las Figuras 6.5 y 6.6, que muestran la variación de resultados para las proporciones de terreno de 20×20 y 50×50 respectivamente. También, se pueden consultar en las Figuras E.5 y E.6 del Apéndice E, las gráficas de crecimiento de la función *fitness* del algoritmo CRO-SL para el vector vientos sintético. Se observa que mantiene ciertas similitudes con las gráficas del BGA. Al igual que ocurría con éste, el CRO-SL, en la mayoría de los casos, sustituye a los individuos de la población si su función de *fitness* es mayor que el individuo con el que compete. Además, se muestra como un comportamiento de crecimiento de los valores medios y máximo bastante paralelo, y en función del modelo, tendencias tanto suaves como escalonadas.

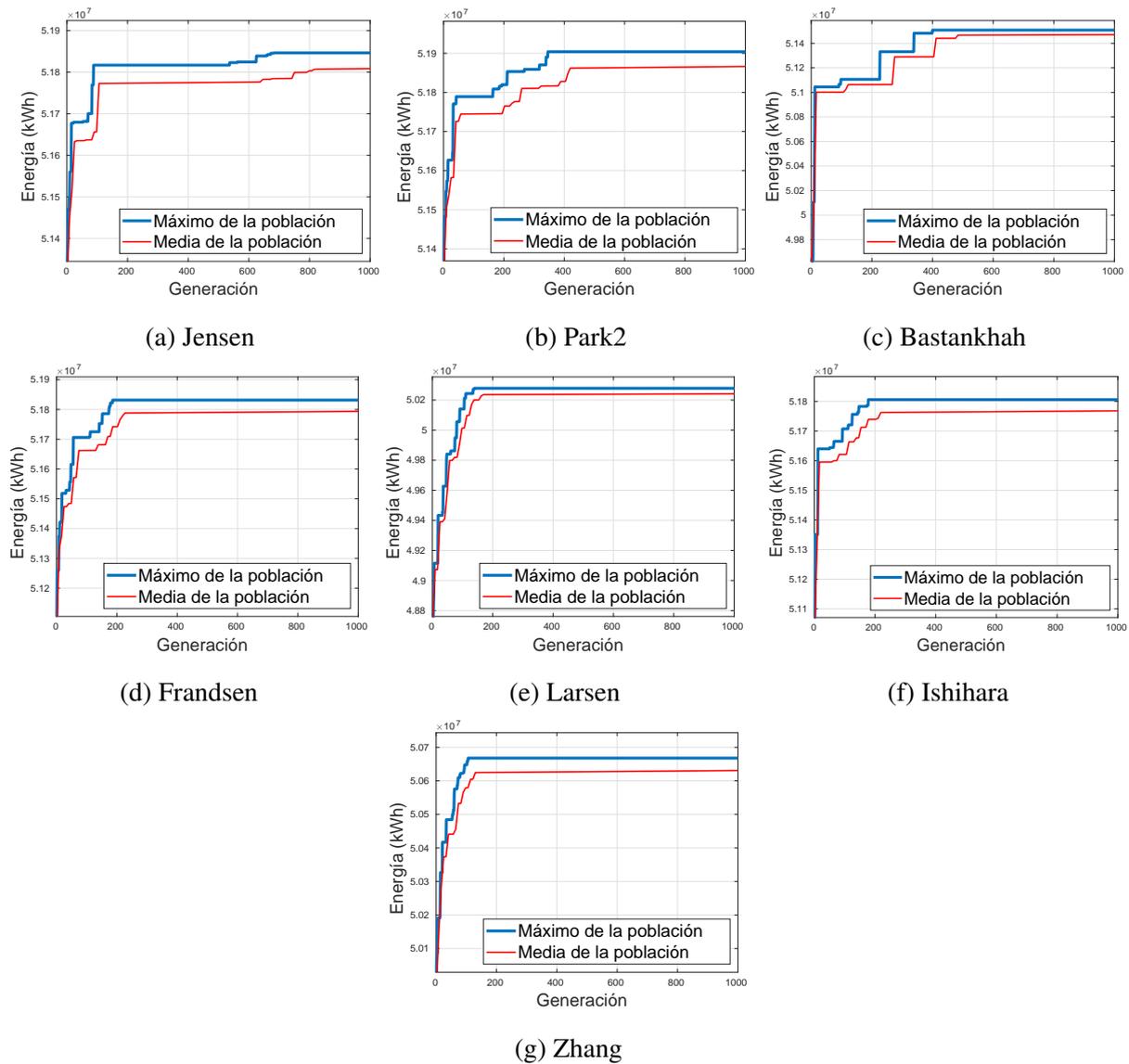


Figura 6.5: Resultados CRO-SL para terreno de 20×20 con viento real.

Adicionalmente, se aplica el mecanismo de defunción que no incorporaba el BGA, y que puede provocar, de forma poco frecuente y por aleatoriedad, que un individuo peor sustituya a otro que da resultados mejores. Independientemente de esto, se muestra un crecimiento continuo tanto de los valores medios y máximo de la población hasta llegar a la zona de convergencia y estacionalidad de las soluciones.

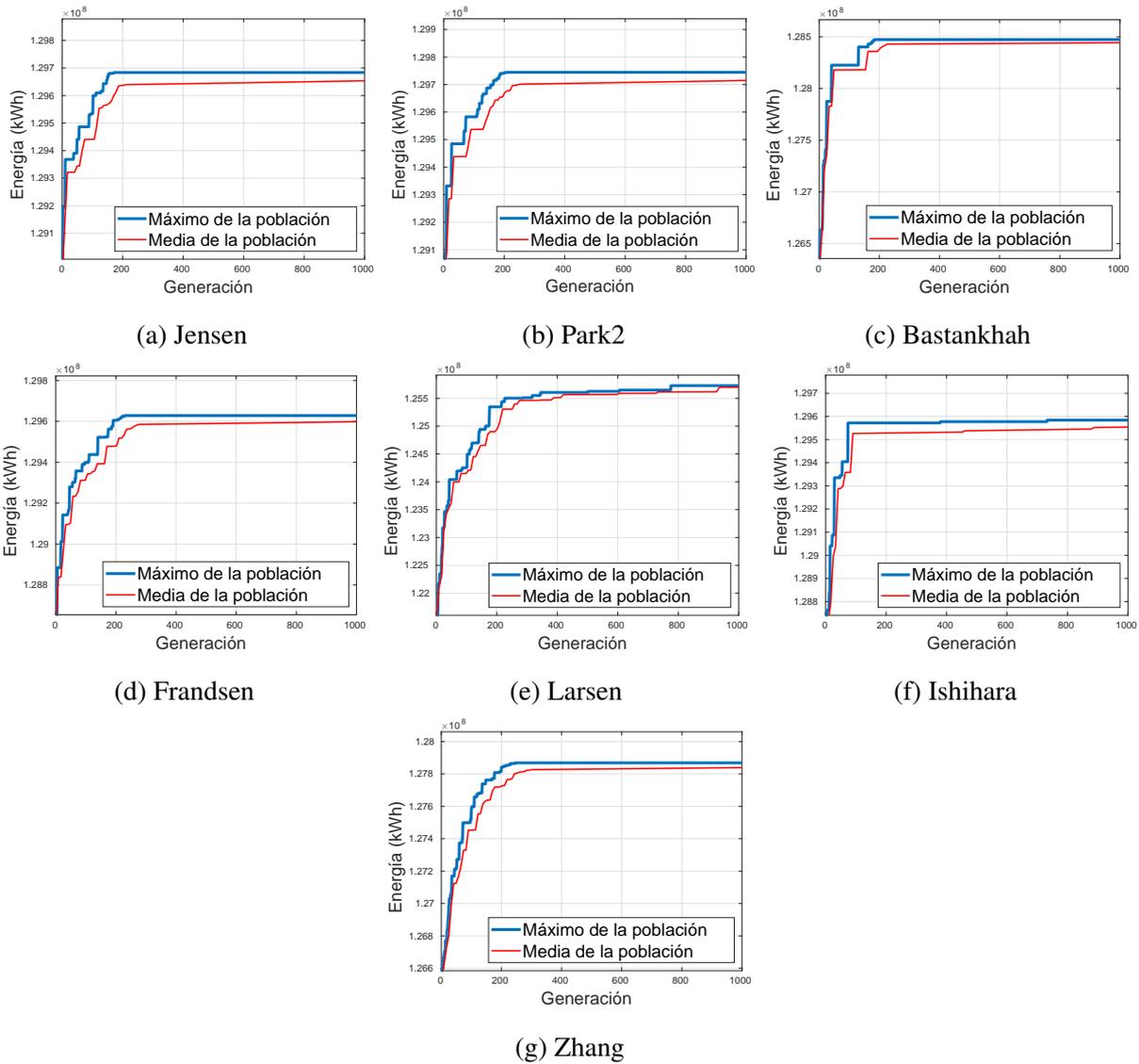


Figura 6.6: Resultados CRO-SL para terreno de 50×50 con viento real.

6.3. Mejores soluciones halladas para el problema

Tras la realización de los análisis cruzados, se ha obtenido la energía generada por cada uno de los modelos de estela sobre las 84 soluciones generadas tras el uso de los 3 algoritmos metaheurísticos encargados de optimizar la distribución de turbinas sobre terrenos de distintas proporciones. Los resultados han sido analizados separando las soluciones en función del tamaño de terreno y de la matriz de vientos usada, lo que nos deja con 4 soluciones finales, consideradas las óptimas, tras todo el proceso. Todas las soluciones para los experimentos que usan proporciones de terreno de 20×20 han sido conseguidas por el BGA, mientras que las mejores soluciones para las configuraciones de 50×50 provienen del algoritmo CRO-SL, lo que concuerda con las conclusiones extraídas en la Sección 6.1. Efectivamente, con este experi-

mento de soluciones cruzadas entre modelos, se confirma que BGA obtuvo mejores soluciones para porcentajes de ocupación turbinas-terreno más elevados (5 % de ocupación en terrenos de 20×20 con 20 turbinas), mientras que el CRO-SL consigue mejores posicionamientos de turbina en casos de porcentaje de ocupación inferiores (2 % de ocupación en terrenos de 50×50 con 50 turbinas).

De igual forma, se han recopilado las mejores soluciones para cada configuración de terreno y viento, de los tres algoritmos, BGA, PSO y CRO-SL. Con esto se busca determinar los modelos de estela de uso preferente a la hora de realizar las optimizaciones con ellos. En los Apéndices G y H se pueden observar todas las tablas de resultados completos de los experimentos cruzados de energía generada y rendimiento respectivamente. Las Tablas 6.7 y 6.8 muestran el algoritmo cuyas soluciones han mostrado un mayor rendimiento global para cada uno de los modelos de estela. Se observa una predominancia clara del algoritmo CRO-SL para las configuraciones de 50 turbinas en terrenos de 50×50 para ambos tipos de vientos, mientras que en configuraciones de 20 turbinas en terrenos de 20×20 , los resultados se encuentran más repartidos, consiguiendo en casi todos los casos el BGA mejores rendimientos en vientos sintéticos y el CRO-SL en vientos reales.

Tabla 6.7: Mejores rendimientos medios de modelo de estela para distribución de vientos sintéticos.

Modelo	20×20 con 20 turbinas		50×50 con 50 turbinas	
	Algoritmo	Rendimiento (%)	Algoritmo	Rendimiento (%)
Jensen	BGA	98.92	CRO-SL	99.12
Park2	BGA	99.43	CRO-SL	99.49
Bastankhah	BGA	98.48	CRO-SL	98.90
Frandsen	BGA	99.43	CRO-SL	99.37
Larsen	BGA	92.40	CRO-SL	90.78
Ishihara	BGA	99.44	CRO-SL	99.55
Zhang	CRO-SL	95.04	CRO-SL	96.79

Las soluciones con mejor media de rendimiento por cada uno de los 3 algoritmos se exponen en la Tabla 6.9, que recoge para cada configuración de terreno, turbinas y vientos, las 4 soluciones optimizadas de posicionamiento de turbinas con una media de rendimiento más alta.

Tabla 6.8: Mejores rendimientos medios de modelo de estela para distribución de vientos real.

Modelo	20×20 con 20 turbinas		50×50 con 50 turbinas	
	Algoritmo	Rendimiento (%)	Algoritmo	Rendimiento (%)
Jensen	BGA	98.67	CRO-SL	99.05
Park2	CRO-SL	99.41	CRO-SL	99.46
Bastankhah	CRO-SL	98.45	CRO-SL	98.79
Frandsen	CRO-SL	99.30	CRO-SL	99.34
Larsen	CRO-SL	93.00	CRO-SL	90.72
Ishihara	CRO-SL	99.45	CRO-SL	99.50
Zhang	CRO-SL	99.18	BGA	96.75

Analizando las mejores soluciones globales por configuración, que se marcan en **negrita**, los algoritmos que las han hallado verifican que se sigue conservando la tendencia de mejor funcionamiento del **BGA** en terrenos de mayor porcentaje de ocupación y del **CRO-SL** en terrenos de menor porcentaje de ocupación, aunque con magnitudes de rendimiento muy similares, y bastante superiores a los obtenidos por el PSO.

Otro hecho a destacar, es el buen funcionamiento del modelo de estela de Larsen a la hora de encontrar soluciones óptimas, ya que de las 12 mejores soluciones extraídas en función del algoritmo y configuración, el modelo de Larsen halló 5, lo que le deja con una participación en el conjunto de mejores resultados del 41.67%. El siguiente modelo que más se le acerca es el modelo de Frandsen, que obtuvo 2 de las 12 mejores soluciones (16.67% de participación), mientras que los otros 5 modelos encontraron una solución cada uno (8.33% de participación). Por lo que se puede concluir que la estela del modelo Larsen es la que tiene mayor nivel de afinidad global con el resto de los modelos analizados.

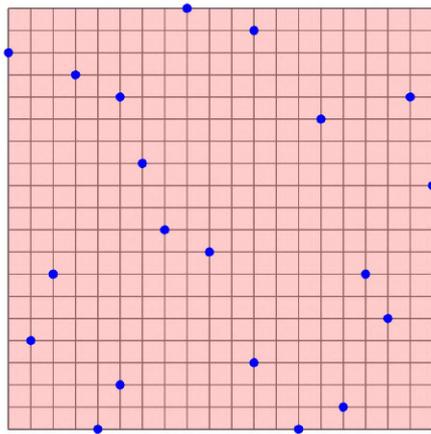
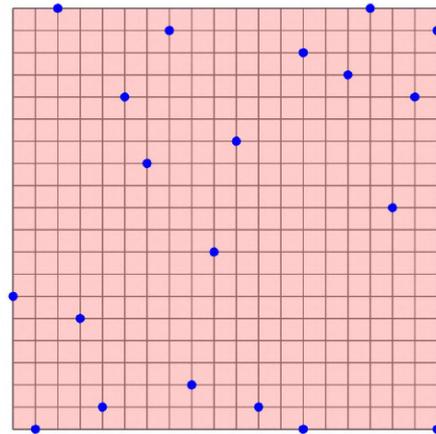
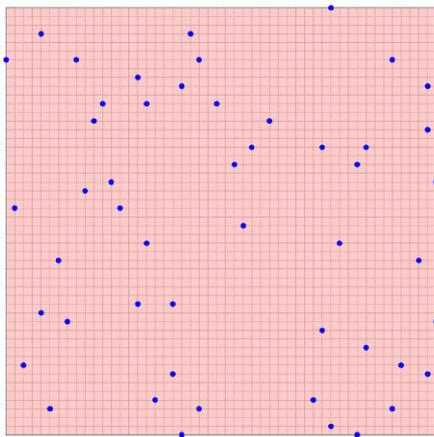
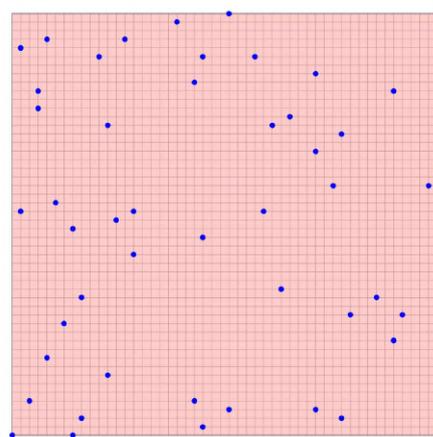
(a) BGA, Park2, 20×20 , sintético.(b) BGA, Jensen, 20×20 , real.(c) CRO-SL, Larsen, 50×50 , sintético.(d) CRO-SL, Larsen, 50×50 , real.

Figura 6.7: Mapa posicional de las soluciones cruzadas con mejor rendimiento medio global.

Tabla 6.9: Soluciones cruzadas con mejores rendimientos medios.

Algoritmo	Modelo	Terreno	Turbinas	Viento	Rendimiento
BGA	Park2	20×20	20	Sintético	97.94 %
BGA	Zhang	50×50	50	Sintético	97.71 %
BGA	Jensen	20×20	20	Real	97.97 %
BGA	Ishihara	50×50	50	Real	97.70 %
PSO	Larsen	20×20	20	Sintético	95.31 %
PSO	Larsen	50×50	50	Sintético	96.47 %
PSO	Frandsen	20×20	20	Real	95.14 %
PSO	Larsen	50×50	50	Real	95.89 %
CRO-SL	Bastankhah	20×20	20	Sintético	97.62 %
CRO-SL	Larsen	50×50	50	Sintético	97.93 %
CRO-SL	Frandsen	20×20	20	Real	97.81 %
CRO-SL	Larsen	50×50	50	Real	97.84 %

Capítulo 7

Conclusiones

En este capítulo se agrupa, a nivel general, todas las conclusiones extraídas tras la búsqueda de información y el desarrollo de los experimentos. En la Sección 7.1 se resumen todas las deducciones y conclusiones aprendidas de los resultados obtenidos y en la Sección 7.2 se orientan hacia el impacto que podrían generar a nivel económico, social y medioambiental. Posteriormente, en la Sección 7.3 se analiza el cumplimiento de las metas marcadas en la Sección 1.2 del Capítulo 1, y en las Secciones 7.4 y 7.5 se enumeran las lecciones aprendidas durante el trabajo y posibles líneas de investigación futuras que pueden surgir a raíz de este trabajo respectivamente.

7.1. Discusión final general

En la realización del trabajo sobre el diseño de algoritmos metaheurísticos para optimizar la colocación de turbinas eólicas, se han aprendido varias lecciones importantes. Una de ellas es que la elección adecuada del algoritmo de optimización es esencial para obtener una solución precisa y eficiente. Durante la investigación se han explorado varios algoritmos metaheurísticos, cada uno con sus propias fortalezas y debilidades, y se ha comprobado que la elección correcta puede tener un gran impacto en el resultado final.

Otro factor relevante identificado en el proceso de optimización es la calidad de los datos utilizados. En particular, la precisión de los datos de viento y topográficos es esencial para obtener una solución de calidad. Además, la función de *fitness* también es un componente clave del algoritmo de optimización y debe ser cuidadosamente diseñada para equilibrar la producción de energía con las restricciones ambientales o geográficas.

De igual forma, se ha enfatizado la importancia de la validación rigurosa de las soluciones generadas por cada uno de los 3 algoritmos. Esto puede implicar la comparación futura de las soluciones generadas con datos reales obtenidos de túneles de viento o metodologías CFD de mayor coste computacional para calcular la tasa de error de las soluciones. También es importante tener en cuenta las posibles implicaciones ambientales y sociales de la colocación de turbinas eólicas.

Otro aspecto crítico es la necesidad de incorporar la incertidumbre en el proceso de optimización. Es probable que los datos iniciales contengan cierta incertidumbre, y es importante tener en cuenta ese factor en la función objetivo y en la selección del algoritmo de optimización. Al no estar tratando con un problema de índole determinista, es necesario tener en cuenta que siempre se incorporará cierto rango de incertidumbre a la solución de nuestros problemas, que en la mayoría de los casos provocará que los resultados reales obtenidos sean peores que los simulados, sobre todo teniendo en cuenta todas las simplificaciones aerodinámicas realizadas. En nuestro caso no se ha incorporado dicha variable por falta de información, debido a que no teníamos datos de ninguna instalación de turbinas eólicas, relacionada con el punto donde se tomó el registro de datos utilizados en el vector vientos real, con el que poder estimar una tasa de error.

Realizando la comparativa entre los 3 algoritmos utilizados, se debe destacar el buen funcionamiento del algoritmo CRO-SL, siendo éste, el que ha generado las mejores soluciones en los procesos de optimización llevados a cabo. La incorporación de las capas de sustratos al algoritmo CRO, hace que se incremente la propiedad exploratoria de nuevas soluciones del algoritmo a la vez que se sigue enfatizando en su faceta explotativa para que las nuevas soluciones sean aptas a la hora de incorporarse a la población y que no desaparezcan. A pesar de su buen comportamiento a la hora de optimizar la colocación de turbinas, el tiempo necesario para hallar las soluciones es mucho más elevado que el requerido en el Algoritmo Genético Básico y PSO, por lo que es importante ponderar si es conveniente su uso o no según el problema que se busque solucionar y cuánta mejora se busque conseguir.

La alta velocidad de ejecución del Algoritmo Genético Básico programado, junto con el rendimiento obtenido a la hora de mejorar su población conforme avanzan las generaciones, hacen de éste, una vía bastante atractiva para realizar experimentos y estudios a gran escala, que requieran probar gran cantidad de configuraciones distintas.

Finalmente, se ha destacado que la optimización no es una solución única y permanente, y que la solución puede requerir de una adaptación y actualización en respuesta a cambios ambientales o evoluciones en el ámbito tecnológico en caso de querer llevarla a un entorno real. Es esencial seguir evaluando la solución y su impacto a largo plazo para asegurar que continúe siendo óptima en el tiempo, ya que pueden darse cambios sustanciales en los vientos de la zona o que surjan nuevas turbinas más eficientes a la hora de generar energía que las ya instaladas. En resumen, el diseño de algoritmos metaheurísticos para optimizar la colocación de turbinas eólicas es un proceso complejo que requiere la consideración cuidadosa de varios factores críticos para lograr una solución óptima y sostenible, tales como el modelo de estela que mejor simula el efecto del viento tras entrar en contacto con los distintos elementos, la geometría del terreno y el número de aerogeneradores que se desea instalar en la zona.

7.2. Impacto social, económico y medioambiental

El correcto posicionamiento de los aerogeneradores en los parques eólicos conlleva una serie de ventajas en los sectores económicos, medioambiental y social, que repercuten de manera directa en la vida de las personas que residen en un determinado territorio.

A nivel social, y en cierta parte también medioambiental, ver aumentos de rendimiento entre el 87-99 % dentro del umbral marcado por los valores recogidos en la Tablas 5.4 y 5.5 del Capítulo 5, hace que se requiera una cantidad menor de turbinas eólicas para abastecer la demanda eléctrica de una determinada zona, lo que provoca que se reduzca el impacto visual sobre el paisaje. Estos factores pueden hacer que se aumente el nivel de aceptación de los parques eólicos y ciertas poblaciones se muestren menos reacias a que se instalen infraestructuras cerca de su zona de residencia (véase como ejemplo una de las protestas ante proyectos macroeólicos de los últimos años en Galicia en [44]).

Atendiendo al aspecto económico, la producción de una cantidad superior de energía haciendo uso de un número menor de aerogeneradores, hace que se reduzcan notablemente los costes de transporte de componentes, instalación y mantenimiento de los distintos dispositivos (NREL realizó un modelo detallado de estimación coste que se puede consultar en [45]). Además, el hecho de necesitar una cantidad menor de turbinas eólicas para generar electricidad en una zona determinada, también permite destinar parte de los recursos sobrantes a diversificar la inversión en otros tipos de energías renovables como la solar fotovoltaica o la hidroeléctrica,

o fabricar nuevos parques eólicos en otros territorios.

En cuanto al beneficio medioambiental, a pesar de ser una energía con un régimen de funcionamiento muy variable, el hecho de aprovechar lo máximo posible los períodos de operación, genera una disminución de la dependencia de otras fuentes de energía más contaminantes como el carbón, favoreciendo así la calidad del aire. Los parques eólicos pueden alterar la fauna de la zona perjudicando gravemente a las aves migratorias que pasan por la zona, tal y como se recoge en [46], donde se analiza el impacto que tienen las turbinas eólicas sobre las aves que transitan el sur de España. Por este motivo, además de intentar minimizar el número de turbinas eólicas necesarias para la producción de energía a través del correcto posicionamiento, es importante hacer un estudio previo de la zona donde se vayan a instalar, con el fin de minimizar el impacto negativo al hábitat de las distintas especies que circulen o habiten el territorio.

Las energías renovables, especialmente la eólica, es esencial para lograr una transición hacia un futuro más sostenible. Ciertas instituciones como ONU y la UE establecieron una serie de objetivos, como la Coalición Global por la Neutralidad del Carbono [6] y el Horizonte Europa [7], para reducir las emisiones de gases de efecto invernadero. Los Objetivos de Desarrollo Sostenible (ODS), establecidos en 2015 por la ONU, son un conjunto de 17 objetivos globales adoptados como parte de la Agenda 2030 para el Desarrollo Sostenible. Su objetivo es abordar los desafíos más urgentes a los que se enfrenta nuestro planeta, incluyendo la pobreza, la protección del medio ambiente y la promoción de la igualdad y la justicia social. Cada uno de los ODS está compuesto por una serie de metas específicas que deben ser alcanzadas en un plazo determinado para lograr un futuro más justo y sostenible, que se pueden consultar en [47]. Tras el análisis de todos ellos, se ha recopilado en la Tabla 7.1 aquellos objetivos y metas a los que aporta un beneficio el estudio llevado a cabo en esta memoria. En concreto destacar el ODS 7, el cual se compenetra muy bien con la temática de este trabajo en prácticamente la totalidad de sus metas específicas.

Tabla 7.1: Metas específicas de los ODS que se adecuan a este estudio.

Objetivo	Meta específica	Descripción
ODS 7: Garantizar el acceso a una energía asequible, segura, sostenible y moderna.	ODS 7.1	De aquí a 2030, garantizar el acceso universal a servicios energéticos asequibles, fiables y modernos.
	ODS 7.2	De aquí a 2030, aumentar considerablemente la proporción de energía renovable en el conjunto de fuentes energéticas.
	ODS 7.3	De aquí a 2030, duplicar la tasa mundial de mejora de la eficiencia energética.
	ODS 7.b	De aquí a 2030, ampliar la infraestructura y mejorar la tecnología para prestar servicios energéticos modernos y sostenibles para todos en los países en desarrollo, en particular los países menos adelantados, los pequeños Estados insulares en desarrollo y los países en desarrollo sin litoral, en consonancia con sus respectivos programas de apoyo.
ODS 8: Promover el crecimiento económico inclusivo y sostenible, el empleo y el trabajo decente para todos.	ODS 8.4	Mejorar progresivamente, de aquí a 2030, la producción y el consumo eficientes de los recursos mundiales y procurar desvincular el crecimiento económico de la degradación del medio ambiente, conforme al Marco Decenal de Programas sobre Modalidades de Consumo y Producción Sostenibles, empezando por los países desarrollados.
ODS 12: Garantizar modalidades de consumo y producción sostenibles.	ODS 12.2	De aquí a 2030, lograr la gestión sostenible y el uso eficiente de los recursos naturales.

7.3. Consecución de objetivos

Se ha creado un repositorio de GitHub con la mayoría de los códigos usados en el proyecto, cuyo enlace de acceso se puede encontrar en el Apéndice C (OG1). En la Sección 3.2 del Capítulo 3 se describe de forma detallada el funcionamiento teórico de los distintos modelos de estela que se quieren usar en el estudio de funcionamiento de los algoritmos metaheurísticos escogidos. Posteriormente en la Sección 4.4 del Capítulo 4 se aportan las pautas seguidas para su implementación en forma de *scripts* a la función de *fitness*, permitiendo analizar las distribuciones de estela con un total de 7 modelos (OE1.2).

Las Secciones 4.5 y 4.6 del Capítulo 4 se desarrolla el proceso de elaboración de los *scripts* correspondientes a los algoritmos PSO y BGA, así como el funcionamiento interno de cada uno y los hiperparámetros que rigen su funcionamiento (OE2.1). En lo referente al CRO-SL, se recogen los pasos seguidos para la adaptación en la Sección 4.7 del Capítulo 4 (OE2.2), dado que el algoritmo únicamente fue adaptado incorporando nuestra función de *fitness* y modificando parte de la estructura, restricciones y sustratos para que pudiesen trabajar con el tipo de individuos matriciales que se manejan. En el Capítulo 6.1 se recogen los valores de energía generada y su rendimiento con respecto a la solución ideal. En la Sección 6.2 del Capítulo 6, se aportan las gráficas de monitorización de los valores de *fitness* medio y máximo obtenidos por la población a lo largo de las iteraciones de los 3 algoritmos utilizados (OE2.3). Gracias a estas tablas y gráficas se ha podido determinar la tendencia de sustitución de individuos y la velocidad de crecimiento por iteración de los distintos algoritmos, lo cual ha permitido comparar el comportamiento de cada uno de ellos para alcanzar una solución óptima (OG2).

Tras el análisis cruzado de resultados, los cuales se pueden ver en la Sección 6.3 del Capítulo 6 y en los Apéndices G y H (OE3.1), se ha conseguido determinar cuál es la mejor distribución para cada configuración de granja eólica establecidas, quedando el posicionamiento de turbinas tal y como se muestra en la Figura 6.7 del Capítulo 6 (OG3). También se han hallado las mejores soluciones generadas por cada uno de los algoritmos genéticos en función de la energía media generada por todos los modelos en el análisis anual de vientos sobre cada solución optimizada, determinando así, el modelo de estela recomendado para optimizar, a través de cada algoritmo, una configuración de parámetros de entorno (OE3.2). Las mejores distribuciones de turbinas para cada algoritmo de estela quedan recogidas en la Tabla 6.9 del Capítulo 6.

7.4. Lecciones aprendidas

A lo largo de la elaboración del TFG, se han ido adquiriendo y mejorando las siguientes destrezas:

1. Funcionamiento interno y más específico de los algoritmos metaheurísticos, los parámetros que son necesarios adaptar para los distintos problemas que se quieran abarcar, sus clasificaciones y estructura de código.
2. La importancia de la correcta asignación de valores a los distintos hiperparámetros de los algoritmos.
3. Nuevas funciones y metodologías de programación en MATLAB, así como el tratamiento de datos, librerías y pasos a seguir en la búsqueda de soluciones.
4. Los modelos de estela usados en el ámbito de las turbinas eólicas y su incidencia sobre las soluciones obtenidas en los procesos de optimización.
5. Redacción en LaTeX de memoria de carácter científico-técnico.
6. Búsqueda y análisis de literatura científica en bases de datos como Google Scholar o Scopus.

7.5. Trabajos futuros

El estudio de optimización de funcionamiento y posicionamiento de turbinas eólicas es un campo muy amplio y con muchas vertientes que pueden ser mejoradas. En este trabajo se han hecho una serie de asunciones simplificadoras que limitaban, en cierto aspecto, la aplicación real de los resultados, como pueden ser la geometría del terreno sobre el que se posiciona la granja (supuesto de geometría cuadrada) y la distribución de alturas de la zona (superficie plana). Como podemos observar, en la realidad estas condiciones son muy complicadas que se den, por lo que una línea de trabajo que se pretende estudiar en el futuro es la posibilidad de realizar las optimizaciones propuestas en terrenos de geometrías variables con determinadas zonas tabú, donde no podrán ser posicionadas las turbinas, y el uso de los modelos de estela tridimensionales que sean capaces de calcular la afectación de las distintas estelas generadas a través de un mapeado de desniveles dado.

El correcto ajuste de hiperparámetros puede ser otro campo a explorar. Como ya se ha mencionado, son un factor muy relevante en la calidad de las soluciones extraídas del proceso de optimización metaheurística, y una correcta selección puede hacer que aumente significativamente los valores de energía obtenidos o que se alcancen en un número menor de iteraciones, ahorrando tiempo de cálculo e incrementando la ganancia económica de la instalación de aerogeneradores. La recopilación de una serie de valores de referencia de hiperparámetros de diversos algoritmos metaheurísticos para el problema de colocación de turbinas sobre el terreno, en función del número de turbinas instaladas, tamaño de terreno y modelo de estela utilizado, podría ser de gran interés para futuros trabajos y proyectos.

Otro aspecto de interés sería la aportación de nuevos sustratos específicos y orientados a granjas eólicas, que mejoren y diversifiquen el funcionamiento del algoritmo CRO-SL actual. Ahora mismo, el algoritmo cuenta con una serie limitada de sustratos, por lo que, el desarrollo e implementación de nuevas capas de sustratos con los que se pudiese conseguir aumentar el rendimiento del mismo a la hora de obtener mejores soluciones para este campo, lo cual influiría muy positivamente en el estudio global de metodologías de distribución de turbinas eólicas sobre el terreno a través de la aplicación de algoritmos metaheurísticos.

En el futuro, también se podría llevar a cabo un estudio sobre el comportamiento y la tendencia que describen las turbinas eólicas a la hora de distribuirse sobre el terreno de manera óptima, lo cual podría ayudar a la elaboración de un listado de restricciones posicionales de carácter dependiente de las posiciones previas ya seleccionadas, lo que generaría un modelo más restrictivo conforme se vayan fijando posiciones. Esto permitiría operar a través de herramientas de optimización lineal como IBM ILOG CPLEX o lenguajes de planificación como PDDL (*Planning Domain Definition Language*) o STRIPS (*Stanford Research Institute Problem Solver*). Este tipo de técnicas han demostrado tener un gran potencial a la hora de conseguir optimizar problemas de diversa índole como se puede ver en la *International Planning Competition* (IPC) [48], competición realizada por el *International Conference on Planning and Scheduling* (ICAPS), donde cada se presenta a la comunidad investigadora, una serie de problemas a resolver para fomentar el desarrollo de nuevas técnicas de planificación automática con PDDL.

Bibliografía

- [1] D. Gibb, N. Ledanois, L. Ranalder, H. Yaqoob, H. E. Murdock, N. Achury, T. Andre, I. Benachir, A. Dhar, S. Gicquel, V. Hamirwasia, G. Mavili, P. Stalter, N. Wafa, Y. Yuan-Perrin, Y. Abd-El-Aziz, J. Chantre Raposo, J. Croft, A. Djahafi, V. Eke, J. Jones-Langley, T. Mayer, L. E. Williamson, L. Mastny, K. Trumbull, J. L. Sawin, F. Sverrisson, T. D. Couture, B. Epp, F. Joubert, K. Seyboth, J. Skeen, G. Wright, A. Brown, H. Abdelnabi, A. Elguindy, N. Fichaux, and R. Levin, “Renewables 2022 global status report + renewable energy data in perspective + press releases + regional fact sheets + country fact sheets,” tech. rep., France, 2022. ENERGY PLANNING, POLICY AND ECONOMY.
- [2] P. Fairley, “Algorithms help turbines share the wind,” Mar 2023.
- [3] X. Gao, Y. Li, F. Zhao, and H. Sun, “Comparisons of the accuracy of different wake models in wind farm layout optimization,” *Energy Exploration & Exploitation*, vol. 38, no. 5, pp. 1725–1741, 2020.
- [4] S. Hitz and J. Smith, “Estimating global impacts from climate change,” *Global Environmental Change*, vol. 14, pp. 201–218, 10 2004.
- [5] “Renewable energy – powering a safer future.” <https://www.un.org/en/climatechange/raising-ambition/renewable-energy>.
- [6] “Net zero coalition.” <https://www.un.org/es/climatechange/net-zero-coalition>.
- [7] “Nuevo programa marco de la ue.” <https://www.horizonteeuropa.es/que-es>.
- [8] “Why is it important to know the locations of wind turbines?.” <https://www.usgs.gov/faqs/why-it-important-know-locations-wind-turbines>.

- [9] S. Zergane, A. Smaïli, and C. Masson, “Optimization of wind turbine placement in a wind farm using a new pseudo-random number generation method,” *Renewable Energy*, vol. 125, 02 2018.
- [10] J. Pérez-Aracil, D. Casillas-Pérez, S. Jiménez-Fernández, L. Prieto-Godino, and S. Salcedo-Sanz, “A versatile multi-method ensemble for wind farm layout optimization,” *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 225, p. 104991, 2022.
- [11] N. Baker, A. Stanley, J. Thomas, A. Ning, and K. Dykes, “Best practices for wake model and optimization algorithm selection in wind farm layout optimization,” 01 2019.
- [12] W. Mengist, T. Soromessa, and G. Legese, “Method for conducting systematic literature review and meta-analysis for environmental science research,” *MethodsX*, vol. 7, p. 100777, 2020.
- [13] S. Rehman, S. Khan, and L. Alhems, “The effect of acceleration coefficients in particle swarm optimization algorithm with application to wind farm layout design,” *FME Transactions*, vol. 48, pp. 922–930, 01 2020.
- [14] S. M. Masoudi and M. Baneshi, “Layout optimization of a wind farm considering grids of various resolutions, wake effect, and realistic wind speed and wind direction data: A techno-economic assessment,” *Energy*, vol. 244, p. 123188, 2022.
- [15] S. V. Nagpal, M. V. Liu, and C. L. Anderson, “A comparison of deterministic refinement techniques for wind farm layout optimization,” *Renewable Energy*, vol. 168, pp. 581–592, 2021.
- [16] V. Mytilinou and A. J. Kolios, “A multi-objective optimisation approach applied to offshore wind farm location selection,” *Journal of Ocean Engineering and Marine Energy*, vol. 3, pp. 265–284, Aug 2017.
- [17] G. Rediske, H. Burin, P. Rigo, C. Rosa, L. Michels, and J. Siluk, “Wind power plant site selection: A systematic review,” *Renewable and Sustainable Energy Reviews*, vol. 148, p. 111293, 2021.
- [18] R. van Haaren and V. Fthenakis, “Gis-based wind farm site selection using spatial multi-criteria analysis (smca): Evaluating the case for new york state,” *Renewable and Sustainable Energy Reviews*, vol. 15, no. 7, pp. 3332–3340, 2011.

- [19] R. L. U. de Freitas Pinto and B. P. F. Gonçalves, “A revised theoretical analysis of aerodynamic optimization of horizontal-axis wind turbines based on bem theory,” *Renewable Energy*, vol. 105, pp. 625–636, 2017.
- [20] A. C. Benim, M. Diederich, and B. Pfeiffelmann, “Aerodynamic optimization of airfoil profiles for small horizontal axis wind turbines,” *Computation*, vol. 6, no. 2, 2018.
- [21] A. Chehouri, R. Younes, A. Ilinca, and J. Perron, “Review of performance optimization techniques applied to wind turbines,” *Applied Energy*, vol. 142, pp. 361–388, 2015.
- [22] W. Xudong, W. Z. Shen, W. J. Zhu, J. N. Sørensen, and C. Jin, “Shape optimization of wind turbine blades,” *Wind Energy*, vol. 12, no. 8, pp. 781–803, 2009.
- [23] X. Gao, H. Yang, and L. Lu, “Study on offshore wind power potential and wind farm optimization in hong kong,” *Applied Energy*, vol. 130, pp. 519–531, 2014.
- [24] S. Salcedo-Sanz, D. Gallo-Marazuela, A. Pastor-Sánchez, L. Carro-Calvo, A. Portilla-Figuera, and L. Prieto, “Offshore wind farm design with the coral reefs optimization algorithm,” *Renewable Energy*, vol. 63, pp. 109–115, 2014.
- [25] J. C. Bansal, P. Farswan, and A. K. Nagar, “Design of wind farm layout with non-uniform turbines using fitness difference based bbo,” *Engineering Applications of Artificial Intelligence*, vol. 71, pp. 45–59, 2018.
- [26] M. Song, K. Chen, Z. He, and X. Zhang, “Bionic optimization for micro-siting of wind farm on complex terrain,” *Renewable Energy*, vol. 50, pp. 551–557, 2013.
- [27] F. Daqaq, R. Ellaia, M. Ouassaid, H. M. Zawbaa, and S. Kamel, “Enhanced chaotic manta ray foraging algorithm for function optimization and optimal wind farm layout problem,” *IEEE Access*, vol. 10, pp. 78345–78369, 2022.
- [28] B. Tseng, C. Garcia, and J. Kwon, “Optimization of wind turbine placement layout on non-flat terrains,” *The International Journal of Industrial Engineering: Theory, Applications and Practice*, vol. 21, pp. 384–395, 01 2014.
- [29] K. Chen, M. Song, and X. Zhang, “A statistical method to merge wind cases for wind power assessment of wind farm,” *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 119, pp. 69–77, 2013.

- [30] H. John, "Holland. genetic algorithms," *Scientific american*, vol. 267, no. 1, pp. 44–50, 1992.
- [31] N. Jensen, "A note on wind generator interaction," *Risø National Laboratory*, 1983.
- [32] O. Rathmann, B. Hansen, K. Hansen, N. Mortensen, and J. Murcia Leon, *The Park2 Wake Model - Documentation and Validation*. DTU Wind Energy E, 2018.
- [33] S. Frandsen, R. Barthelmie, S. Pryor, O. Rathmann, S. Larsen, J. Højstrup, and M. Thøgersen, "Analytical modelling of wind speed deficit in large offshore wind farms," *Wind Energy*, vol. 9, no. 1-2, pp. 39–53, 2006.
- [34] G. C. Larsen, "Simple wake calculation procedure. [wind turbines]," tech. rep., Dec 1988.
- [35] J. Pierik, J. Dekker, H. Braam, B. Bulder, D. Winkelaar, G. Larsen, E. Morfiadakis, P. Chaviropoulos, A. Derrick, and J. Molly, "European wind turbine standards ii (ewts-ii)," tech. rep., Netherlands Energy Research Foundation ECN, Petten, Netherlands, Mar. 1999. <https://www.osti.gov/etdeweb/biblio/357560>.
- [36] Y. Cheng, M. Zhang, Z. Zhang, and J. Xu, "A new analytical model for wind turbine wakes based on monin-obukhov similarity theory," *Applied Energy*, vol. 239, pp. 96–106, 2019.
- [37] T. Ishihara and G.-W. Qian, "A new gaussian-based analytical wake model for wind turbines considering ambient turbulence intensities and thrust coefficient effects," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 177, pp. 275–292, 2018.
- [38] H. H. Frisinger, *Aristotle's Meteorologica*. Boston, MA: American Meteorological Society, 1983.
- [39] V. Pollio, H. Warren, and M. Morgan, *The Ten Books on Architecture*. Harvard University Press, 1914.
- [40] P. Lissaman, "Energy effectiveness of arbitrary arrays of wind turbines," *Journal of Energy*, vol. 3, no. 6, pp. 323–328, 1979.
- [41] O. H. Bossuy, "Modelling and validation of wind turbine wake superposition using wind farm data," tech. rep., The Technical University of Denmark. Delft University of Technology, Denmark, Aug. 2018.

<https://repository.tudelft.nl/islandora/object/uuid:8a9ff2f3-d4d4-470b-a5b8-11a08e31eb16/datastream/OBJ/download>.

- [42] D. Zhang, Y. Liang, C. Li, Y. Xiao, and G. Hu, “Applicability of wake models to predictions of turbine-induced velocity deficit and wind farm power generation,” *Energies*, vol. 15, p. 7431, 10 2022.
- [43] F. Avia, *Curva de potencia del aerogenerador (AWEC-60)*. Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas (Ciemat), 1992.
- [44] E. Press, “Ortigueira se manifiesta para pedir que no se “asedie” con “macro eólicos”,” feb 2023. https://www.niusdiario.es/espana/galicia/20230216/concentracion-ortigueira-coruna-pedir-no-asedie-macro-eolicos_18_08756816.html.
- [45] F. LJ, M. Hand, and L. AS, “Wind turbine design cost and scaling model,” 01 2006.
- [46] M. A. Farfán, J. M. Vargas, J. Duarte, and R. Real, “What is the impact of wind farms on birds? a case study in southern spain,” *Biodiversity and Conservation*, vol. 18, pp. 3743–3758, Dec 2009.
- [47] “Objetivos y metas de desarrollo sostenible.” <https://www.un.org/sustainabledevelopment/es/objetivos-de-desarrollo-sostenible/>.
- [48] “Icaps competitions - ipc.” <https://www.icaps-conference.org/competitions/>.
- [49] A. Kumar, “Estimating the flammable mass of a vapor cloud. by john l. woodward center for chemical process safety (ccps) american institute of chemical engineers, new york, ny (1999) 292 pages pub g-60, isbn 0-8169-0778-1,” *Environmental Progress - ENVIRON PROG*, vol. 19, 12 2000.

Apéndice A

Glosario de variables

- A_r : área del rotor.
- c_1 : aceleración personal de la partícula.
- c_2 : aceleración social de la partícula.
- c_T : coeficiente de empuje/tracción de la turbina eólica.
- D_r : diámetro del rotor.
- E_{gen} : energía anual generada por una distribución de turbinas para un modelo de estela.
- E_{max} : energía anual generada máxima por un modelo de estela determinado.
- E_{min} : energía anual generada mínima por un modelo de estela determinado.
- I_0 : intensidad de turbulencia ambiental estacionaria.
- I_a : intensidad de turbulencia ambiental añadida.
- I_w : intensidad de turbulencia ambiental en la estela.
- r : distancia radial de análisis medida desde el centro de la estela.
- U_∞ : corriente no perturbada aguas arriba.
- U_x : velocidad de la corriente en un punto concreto del plano.
- w : factor de inercia.
- x : distancia axial de análisis medida desde la turbina eólica generadora de estela.

- x_g^{Best} : mejor posición histórica del grupo que ha alcanzado una partícula entre toda la población.
- x_p^{Best} : mejor posición personal obtenida por una partícula en todo su historial propio de movimientos.
- z : altitud de análisis respecto al terreno.
- z_0 : rugosidad del terreno.
- z_e : altura del elemento rugoso sobre el terreno.
- z_h : altura del eje rotor de la turbina eólica.
- β : ángulo de resbalamiento de la pala.
- δ : anchura radial de la estela.
- σ : desviación típica.
- $\{S\}$: conjunto de soluciones optimizadas.
- $\{WM\}$: conjunto de modelos de estela.

Apéndice B

Glosario de acrónimos

- 1Px: *One Point Crossover.*
- 2Px: *Two Points Crossover.*
- ACO: *Ant Colony Optimization.*
- AEDLS: *Ambient-based Energy Deficit Linear Sum.*
- AVDLS: *Ambient-based Velocity Deficit Linear Sum.*
- AVDRSS: *Ambient-based Velocity Deficit Root Sum Square.*
- BBBO: *Blend Biogeography-based Optimization.*
- BBO: *Biogeography-based Optimization.*
- BCO: *Bee Colony Optimization.*
- BGA: *Basic Genetic Algorithm.*
- BO: *Bionic Optimization.*
- CIA: *Cluster Identification Algorithm.*
- CMRFA: *Chaotic Manta Ray Foraging Algorithm.*
- CRO: *Coral Reefs Optimization.*
- CRO-SL: *Coral Reefs Optimization with Substrate Layers.*
- DE: *Differential Evolution.*

- FD-BBO: *Fitness Difference Based BBO*.
- GM: *Gaussian Mutation*.
- HS: *Harmonic Search*.
- IA: Inteligencia Artificial.
- ICAPS: *International Conference on Planning and Scheduling*.
- IEA: *International Energy Agency*.
- IPC: *International Planning Competition*.
- MOGA: *Multi-Objective Genetic Algorithm*.
- MPx: *Multi Point Crossover*.
- NREL: *National Renewable Energy Laboratory*.
- ODS: Objetivos de Desarrollo Sostenible.
- OE: Objetivos Específicos.
- OG: Objetivos Generales.
- ONU: Organización de Naciones Unidas.
- PDDL: *Planning Domain Definition Language*.
- PRSO: *Pseudorandom Search Optimization*.
- PSO: *Particle Swarm Optimization*.
- REDLS: *Rotor-based Energy Deficit Linear Sum*.
- RVDLS: *Rotor-based Velocity Deficit Linear Sum*.
- RVDRSS: *Rotor-based Velocity Deficit Root Sum Square*.
- SLR: *Systematic Literature Review*.
- TFG: Trabajo Fin de Grado.
- UE: Unión Europea.

Apéndice C

Scripts utilizados para los experimentos

Se puede consultar los distintos *scripts* de MATLAB y archivos solución en formato ‘.mat’ en el repositorio de GitHub accesible a través la siguiente dirección URL:

`https://github.com/antoniojromero/tfg_antoniojromero`

En este repositorio están disponibles los *scripts* de creación propia: Algoritmo Genético Básico, algoritmo PSO, modelos de estela y el resto de archivos de la función de *fitness*. No se encuentra ahí el algoritmo CRO-SL debido a que se adaptó la estructura ya existente de *scripts* de creación no propia, y al carecer de consentimiento por parte del autor original, se ha decidido no incorporarlo al repositorio.

Apéndice D

Valores orientativos de rugosidad del terreno

En la Tabla D.1 se recopilan una serie de valores orientativos para la variable de rugosidad z en función de la tipología de terreno sobre la que se sitúen las turbinas eólicas. Esta variable es utilizada en modelos como el de Jensen descrito en la Sección 3.2.1 del Capítulo 3, para el cálculo de la pendiente del cono de la estela.

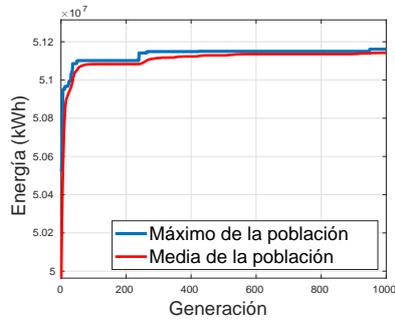
Tabla D.1: Valores de rugosidad para distintas tipologías de terreno. Información obtenida de [49].

$z_0(m)$	Clase general de terreno	Terreno específico
$1.0 \cdot 10^{-4}$	Extensión agua/desierto	Mar abierto calmado o suelo plano laminado o con capa de nieve.
$1.0 \cdot 10^{-3}$	Extensión agua/desierto	Áreas costeras.
$2.0 \cdot 10^{-3}$	Extensión agua/desierto	Suelo de cultivo nevado.
$5.0 \cdot 10^{-3}$	Suelo nivelado cubierto de vegetación	Césped (30 cm aprox.).
$1.0 \cdot 10^{-2}$	Suelo nivelado cubierto de vegetación	Pocos árboles (en invierno).
$2.5 \cdot 10^{-2}$	Pistas aeropuertos	Árboles aislados.
$3.0 \cdot 10^{-2}$	Tierras de cultivo	Casas rurales.
$5.0 \cdot 10^{-2}$	Tierras de cultivo	Pocos árboles (en verano).
$8.0 \cdot 10^{-2}$	Tierras de cultivo	Muchos setos.
0.2	Terreno rural	Muchos árboles, setos, pequeños edificios...
0.4	Muchos niveles boscosos	Límites de ciudades.
0.5	Terreno rural	Centro de pequeños pueblos.
1.0	Bosques	Centro de grandes pueblos, ciudades. Casos urbanos. Plantas de procesamiento.
1.5 - 3.0	Con desniveles y montañosos	Centro de ciudades con muchos edificios de gran altura.

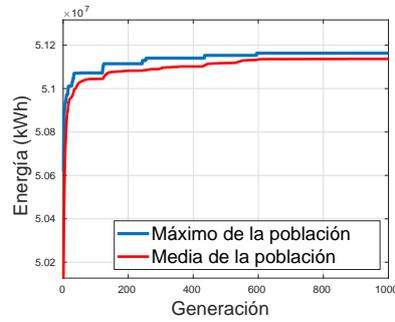
Apéndice E

Evolución de valores de la función de *fitness*

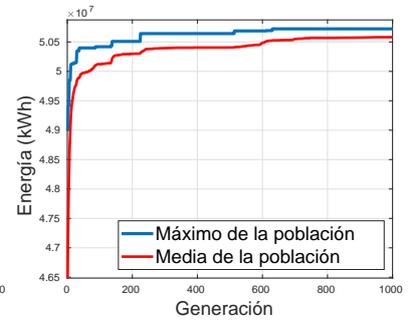
En este apéndice se muestran las gráficas correspondientes a la evolución de los valores de la función *fitness* del mejor individuo de la población y la media general de todos los individuos, conforme avanzan las iteraciones de los algoritmos BGA, PSO y CRO-SL. Únicamente se recogen, de forma complementaria, las configuraciones de vientos sintéticos en terrenos de proporciones 20×20 con 20 turbinas y 50×50 con 50 turbinas, no mostradas en el contenido principal de la memoria.



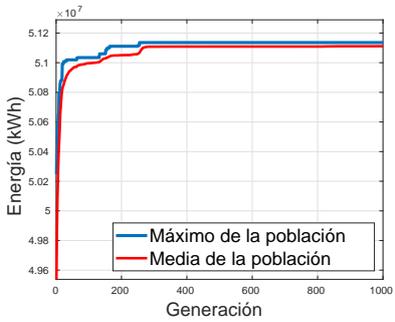
(a) Jensen



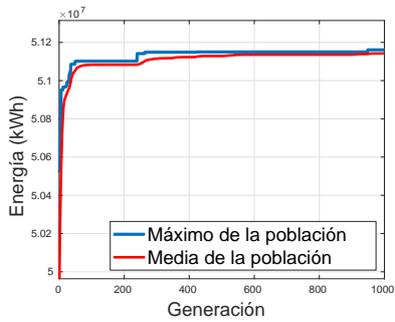
(b) Park2



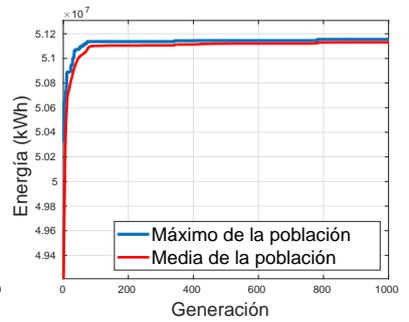
(c) Bastankhah



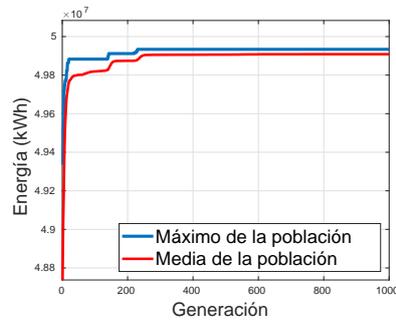
(d) Frandsen



(e) Larsen



(f) Ishihara



(g) Zhang

Figura E.1: Resultados Algoritmo Genético Básico para terreno de 20×20 con viento sintético.

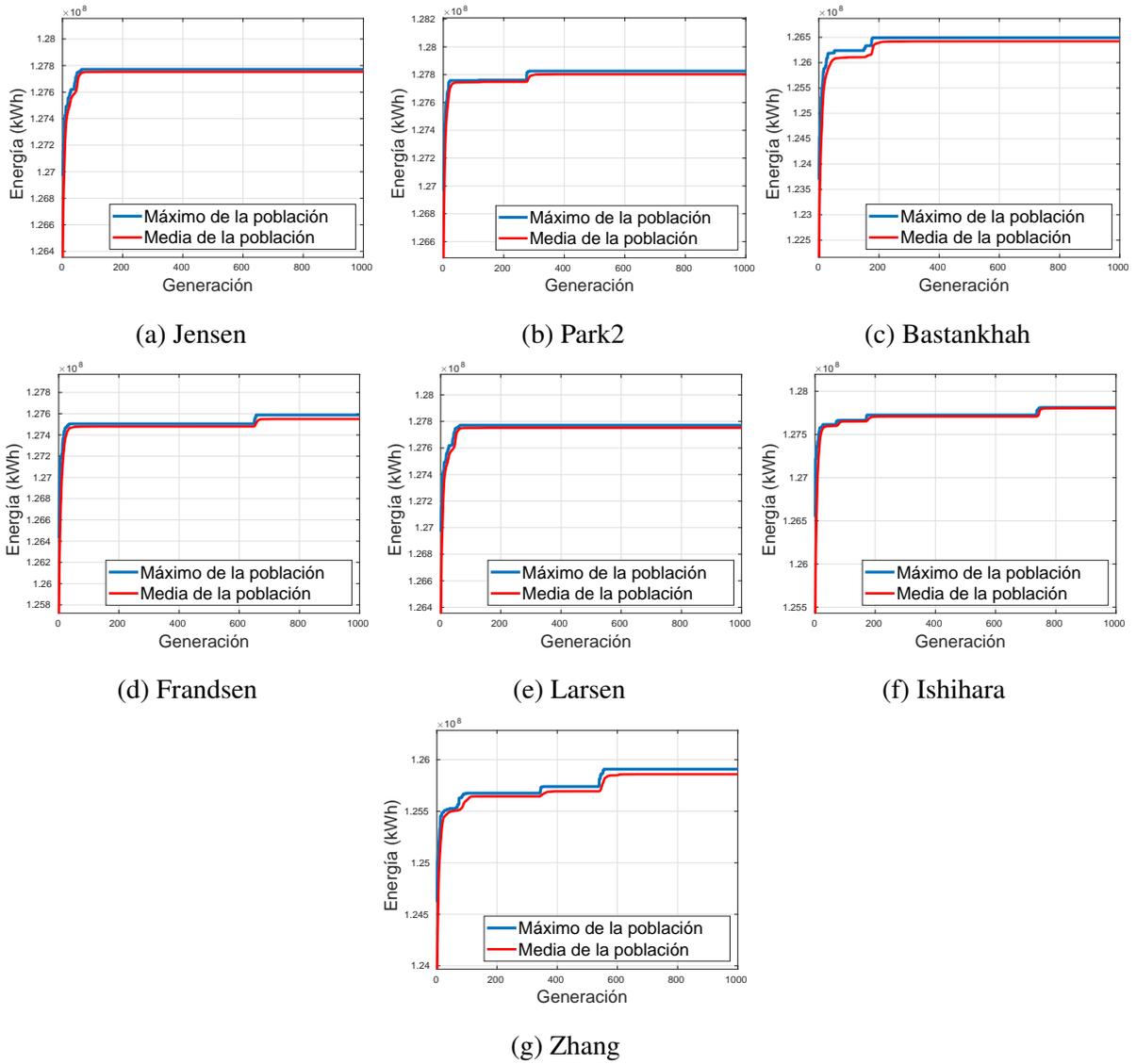
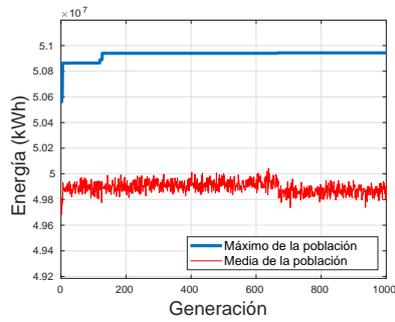
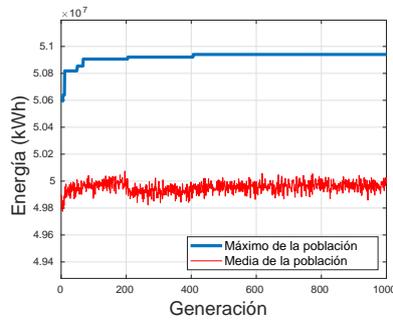


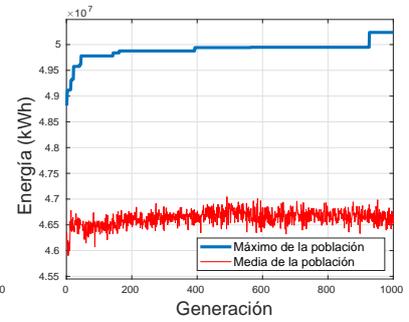
Figura E.2: Resultados Algoritmo Genético Básico para terreno de 50×50 con viento sintético.



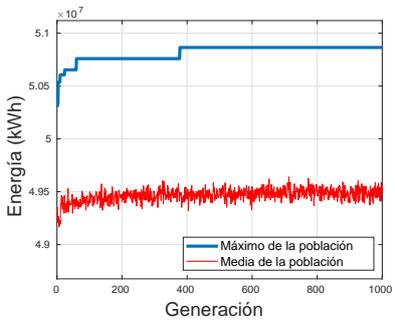
(a) Jensen



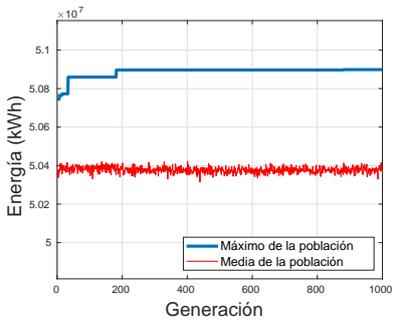
(b) Park2



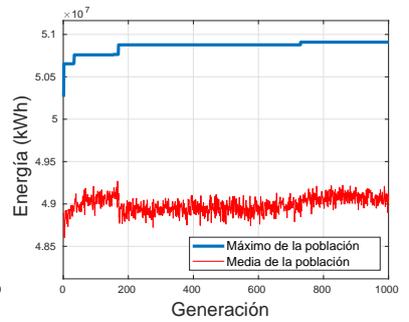
(c) Bastankhah



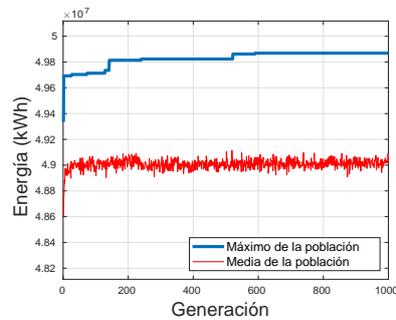
(d) Frandsen



(e) Larsen

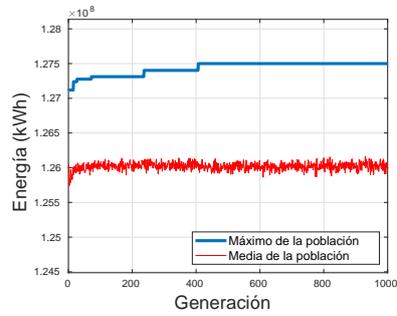


(f) Ishihara

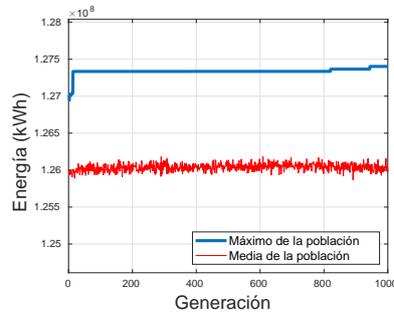


(g) Zhang

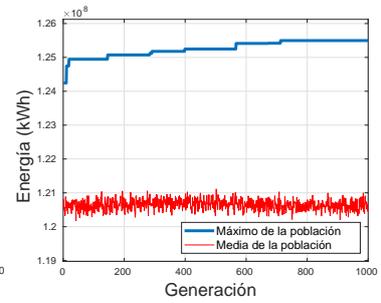
Figura E.3: Resultados *Particle Swarm Optimization* para terreno de 20×20 con viento sintético.



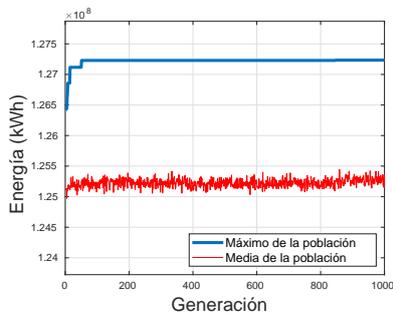
(a) Jensen



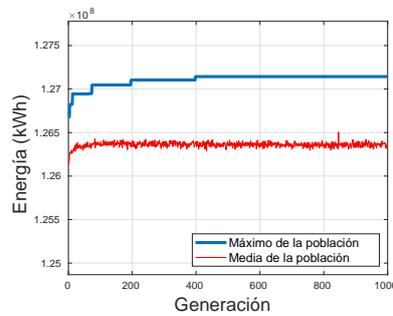
(b) Park2



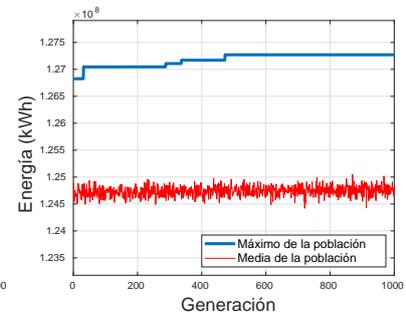
(c) Bastankhah



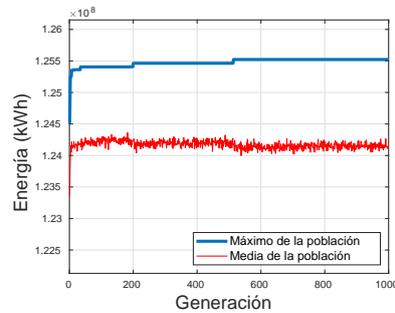
(d) Frandsen



(e) Larsen

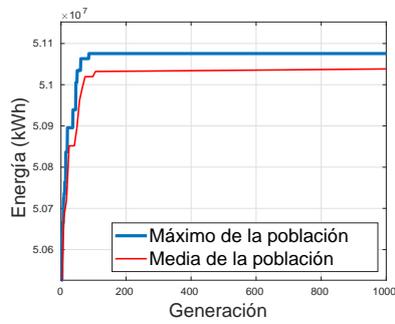


(f) Ishihara

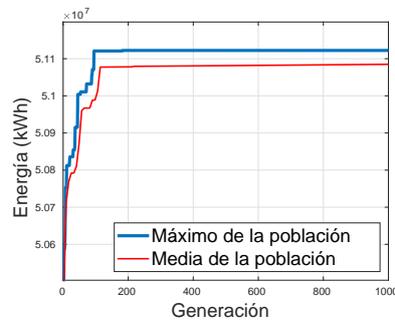


(g) Zhang

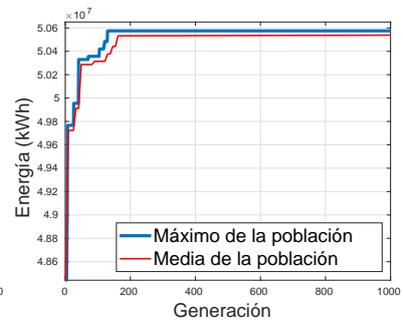
Figura E.4: Resultados *Particle Swarm Optimization* para terreno de 50×50 con viento sintético.



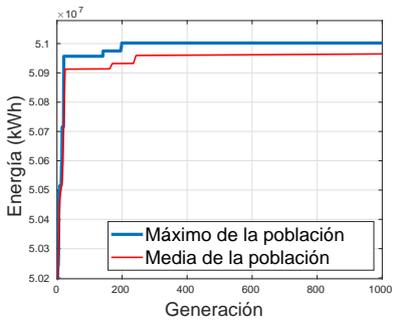
(a) Jensen



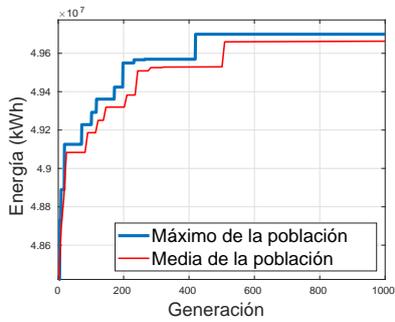
(b) Park2



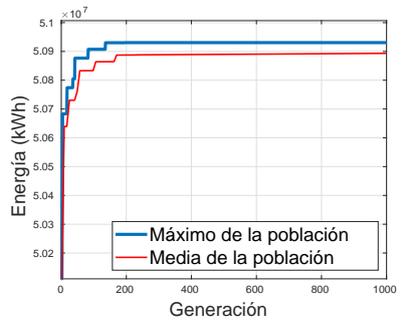
(c) Bastankhah



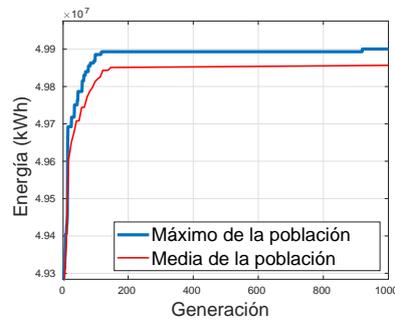
(d) Frandsen



(e) Larsen

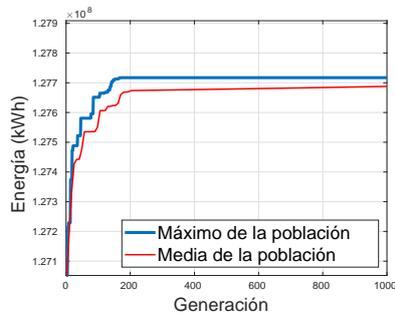


(f) Ishihara

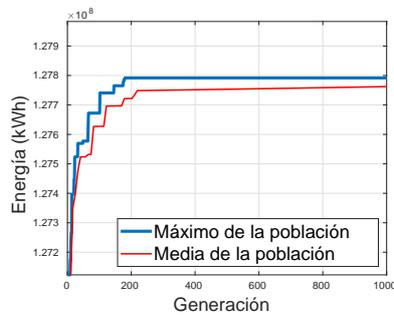


(g) Zhang

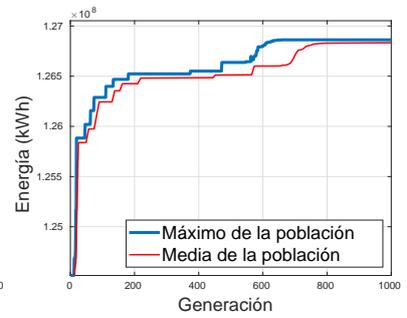
Figura E.5: Resultados CRO-SL para terreno de 20×20 con viento sintético.



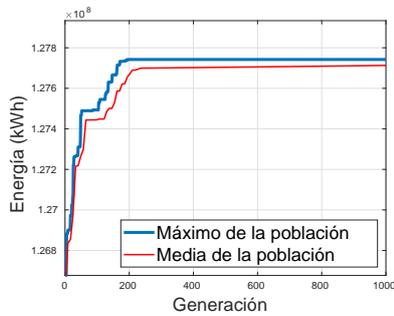
(a) Jensen



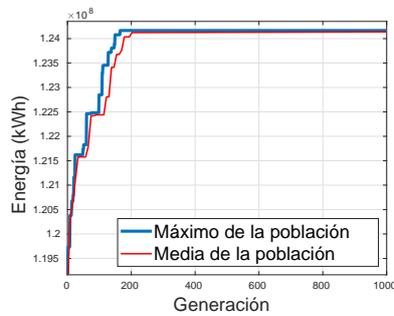
(b) Park2



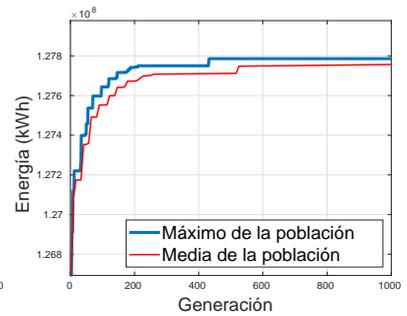
(c) Bastankhah



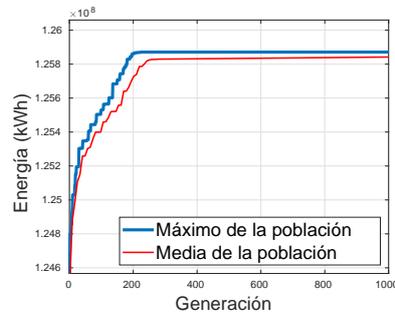
(d) Frandsen



(e) Larsen



(f) Ishihara



(g) Zhang

Figura E.6: Resultados CRO-SL para terreno de 50×50 con viento sintético.

Apéndice F

Soluciones de posicionamiento de turbinas

Este apéndice recopila las figuras que muestran el mapeo de turbinas sobre un *grid* de proporciones 20×20 y 50×50 con vientos sintéticos y reales, para cada uno de los modelos de estela y algoritmos utilizados.

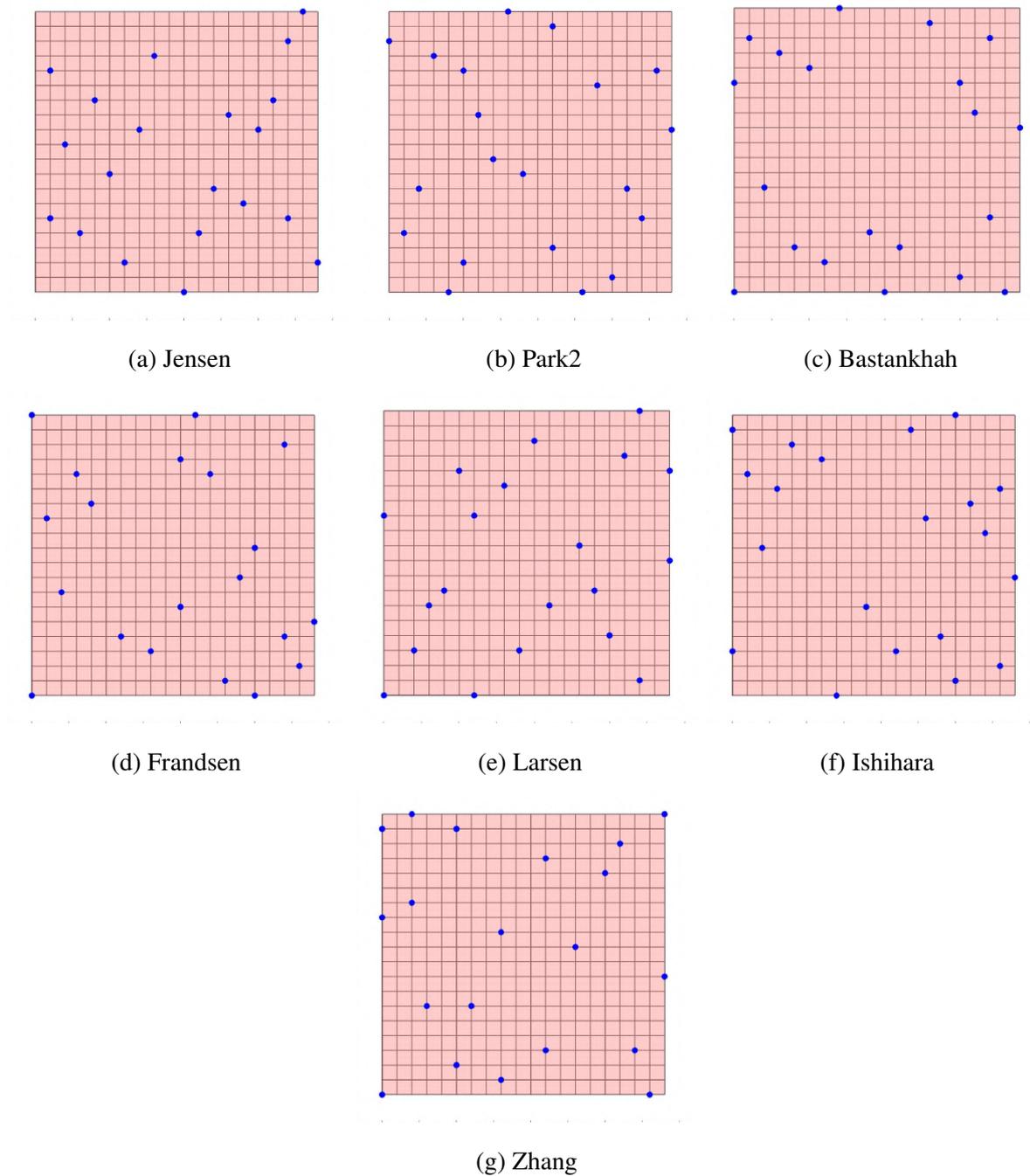
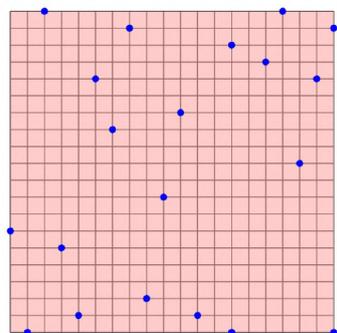
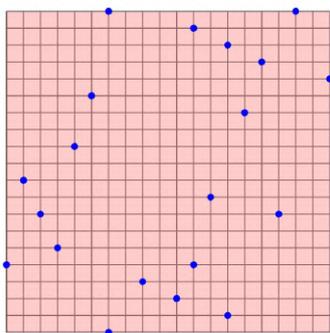


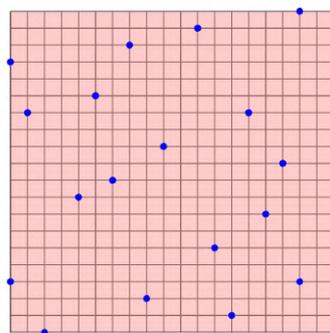
Figura F.1: Resultados Algoritmo Genético Básico para terreno de 20×20 con viento sintético.



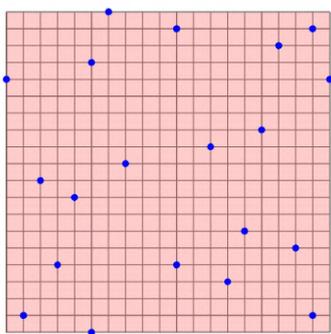
(a) Jensen



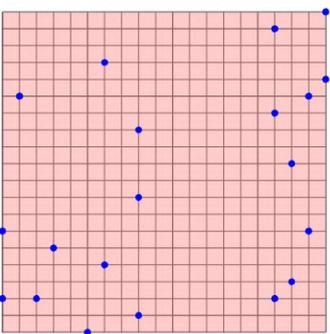
(b) Park2



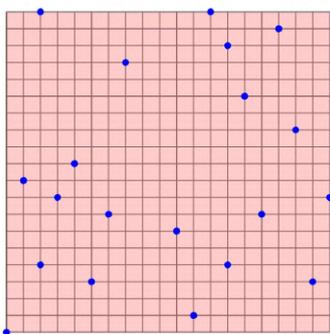
(c) Bastankhah



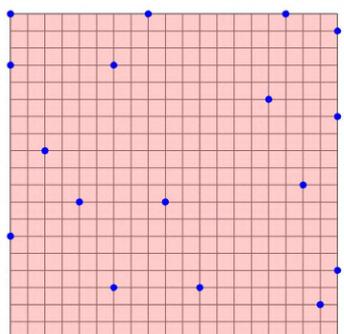
(d) Frandsen



(e) Larsen



(f) Ishihara



(g) Zhang

Figura F.2: Resultados Algoritmo Genético Básico para terreno de 20×20 con viento real.

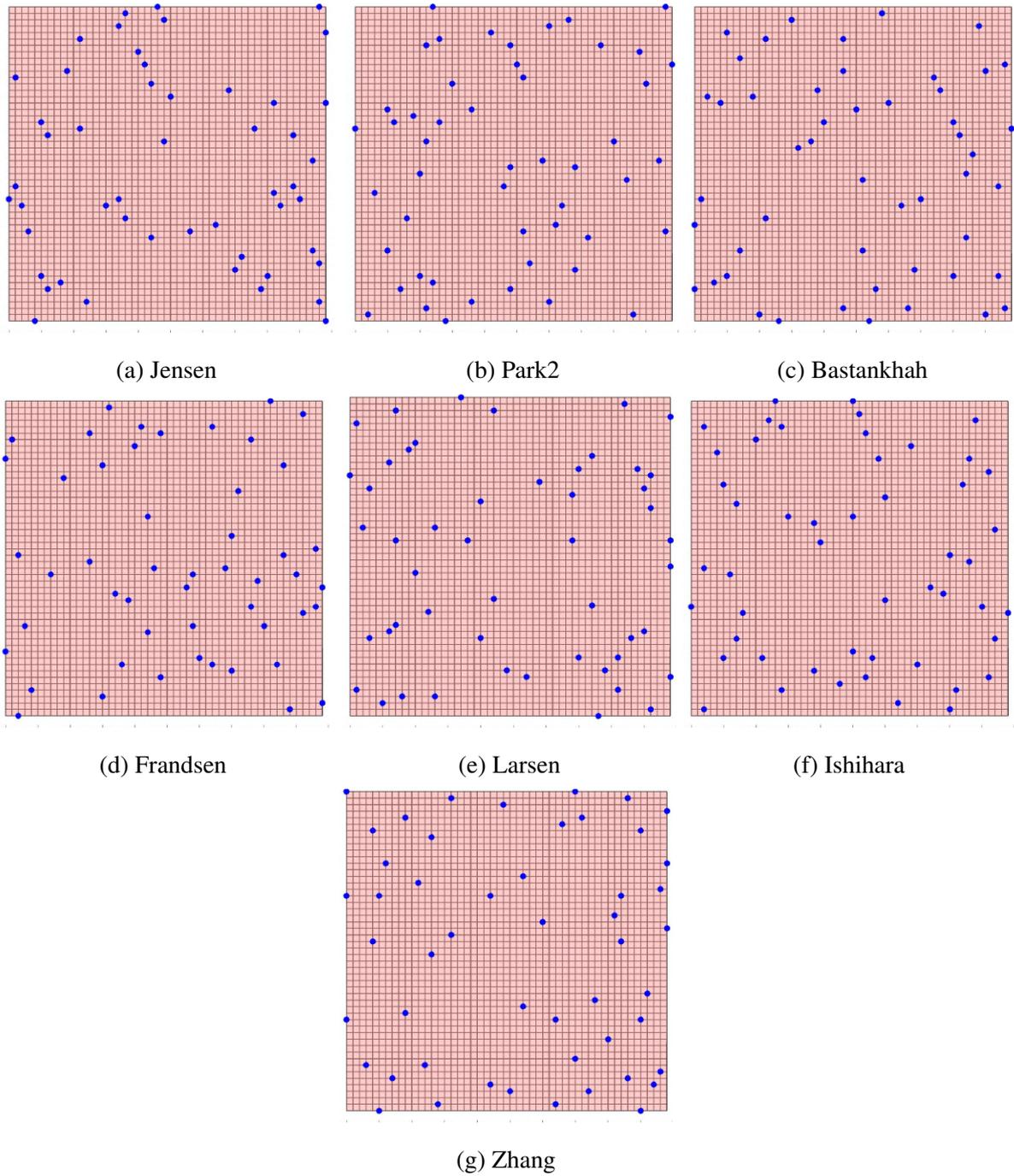


Figura F.3: Resultados Algoritmo Genético Básico para terreno de 50×50 con viento sintético.

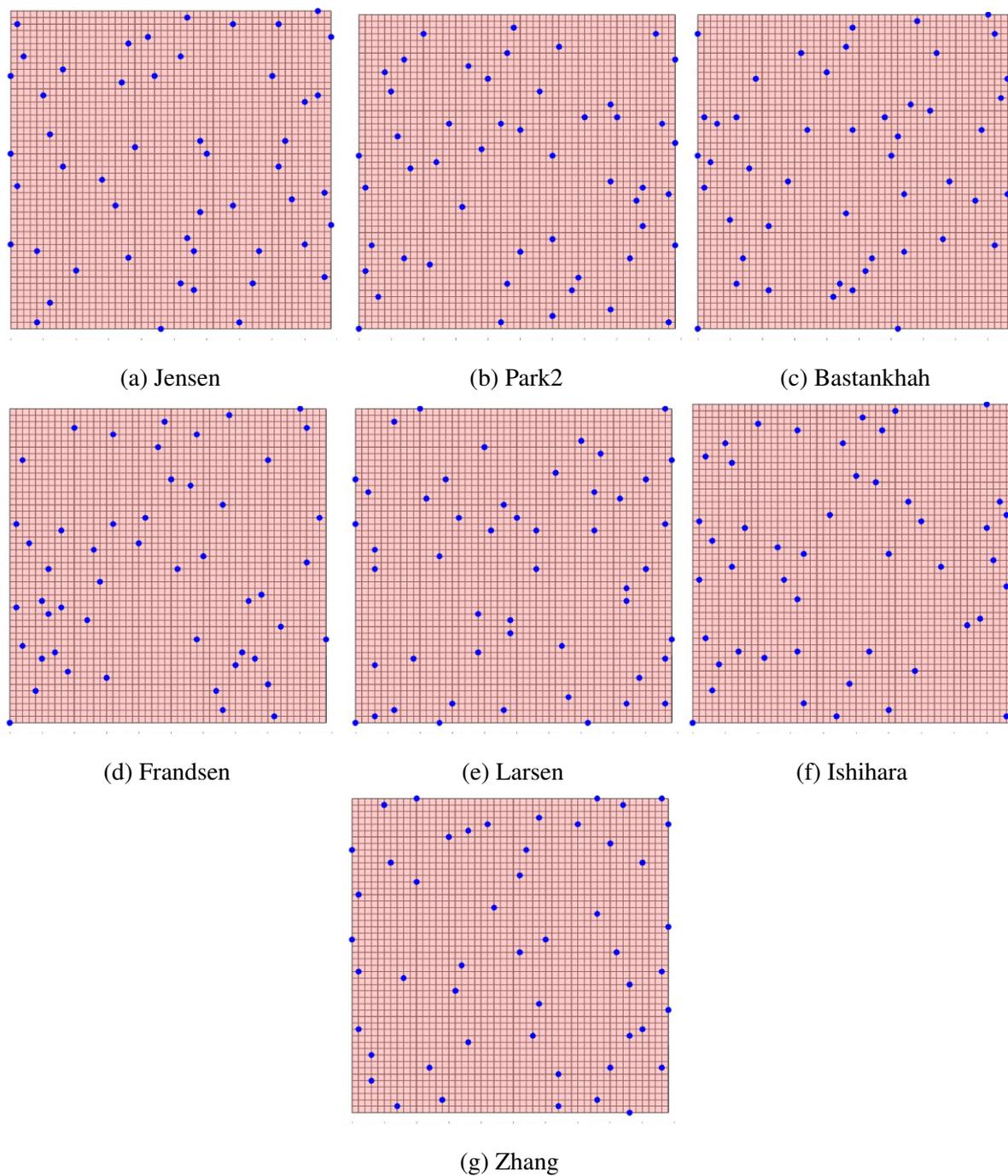
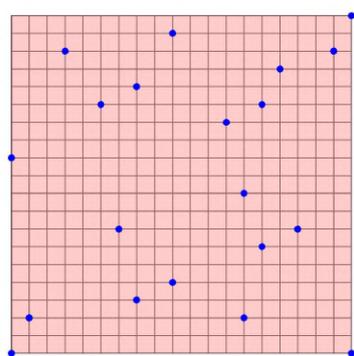
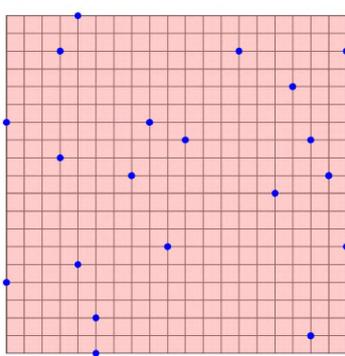


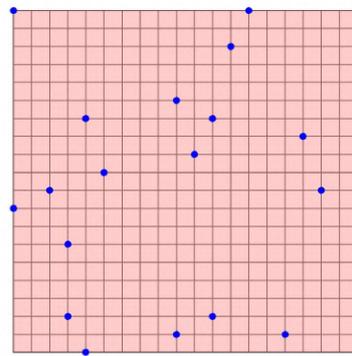
Figura F.4: Resultados Algoritmo Genético Básico para terreno de 50×50 con viento real.



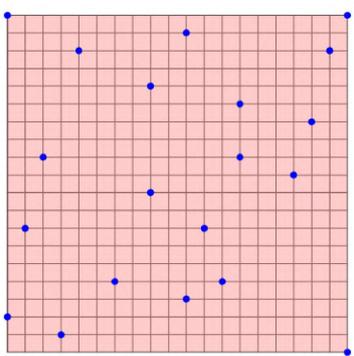
(a) Jensen



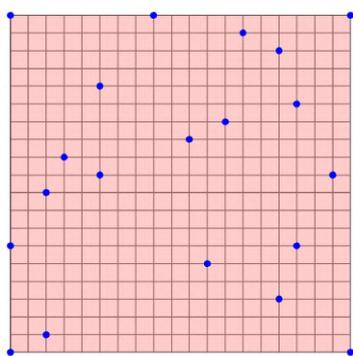
(b) Park2



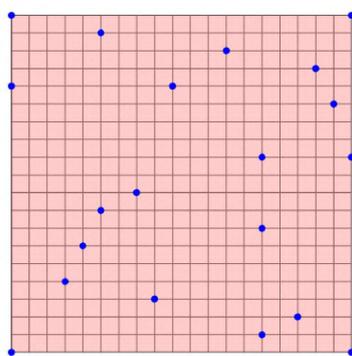
(c) Bastankhah



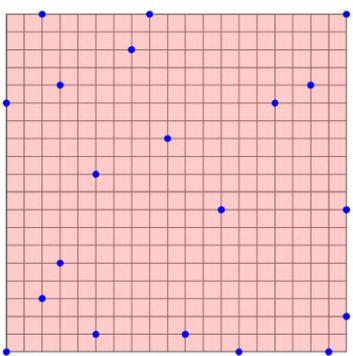
(d) Frandsen



(e) Larsen



(f) Ishihara



(g) Zhang

Figura F.5: Resultados Algoritmo PSO para terreno de 20×20 con viento sintético.

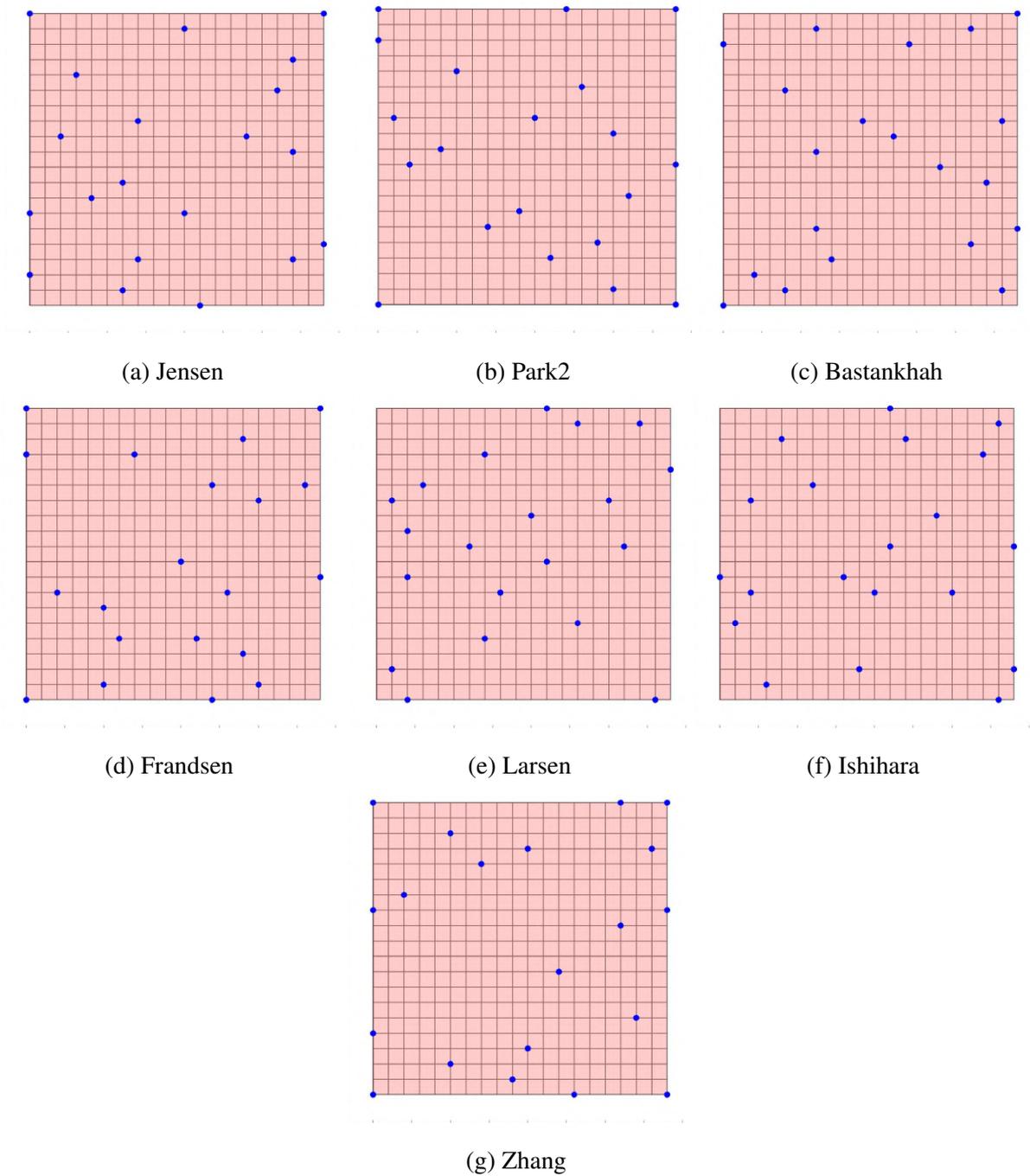


Figura F.6: Resultados Algoritmo PSO para terreno de 20×20 con viento real.

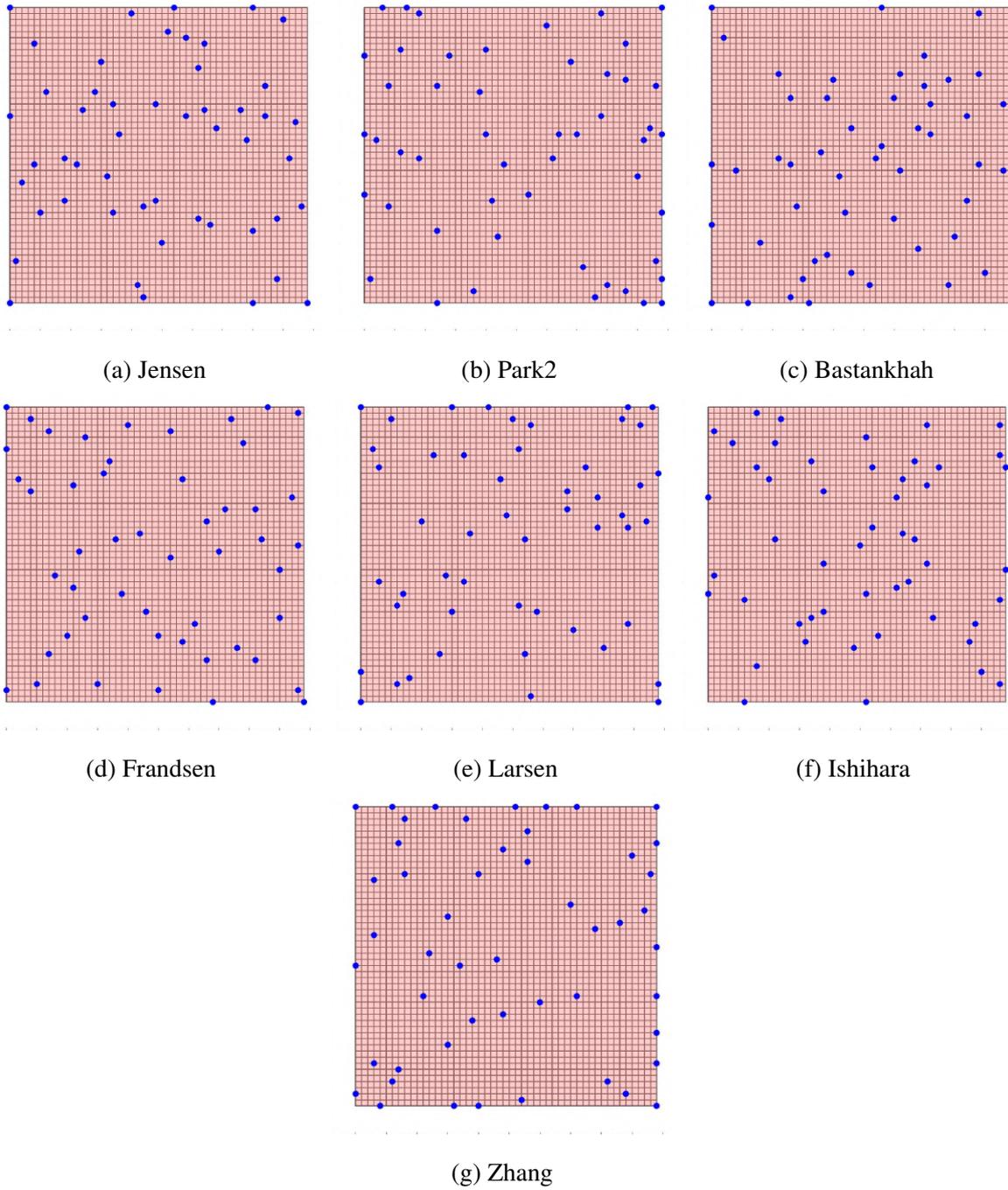
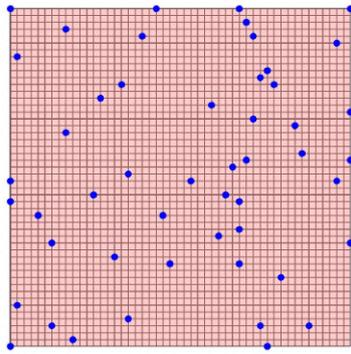
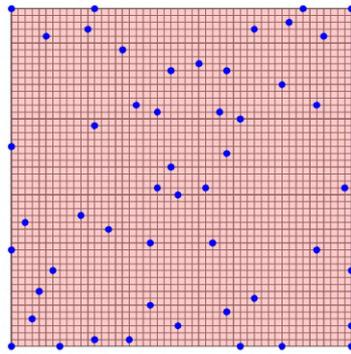


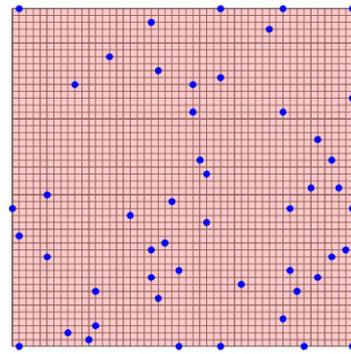
Figura F.7: Resultados Algoritmo PSO para terreno de 50×50 con viento sintético.



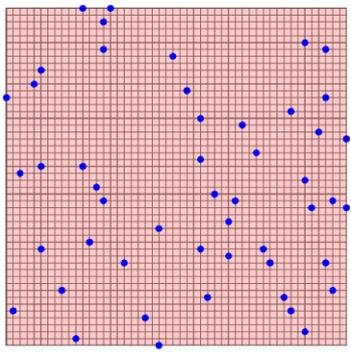
(a) Jensen



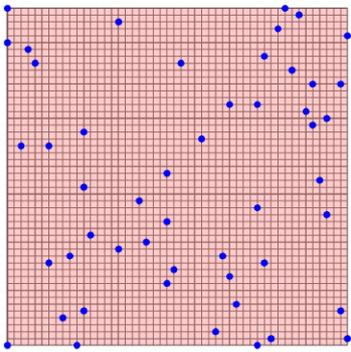
(b) Park2



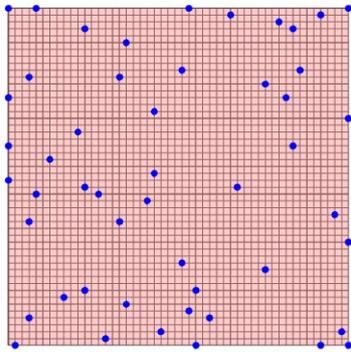
(c) Bastankhah



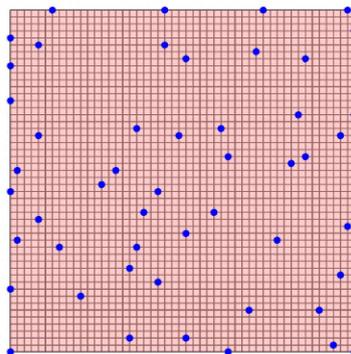
(d) Frandsen



(e) Larsen



(f) Ishihara



(g) Zhang

Figura F.8: Resultados Algoritmo PSO para terreno de 50×50 con viento real.

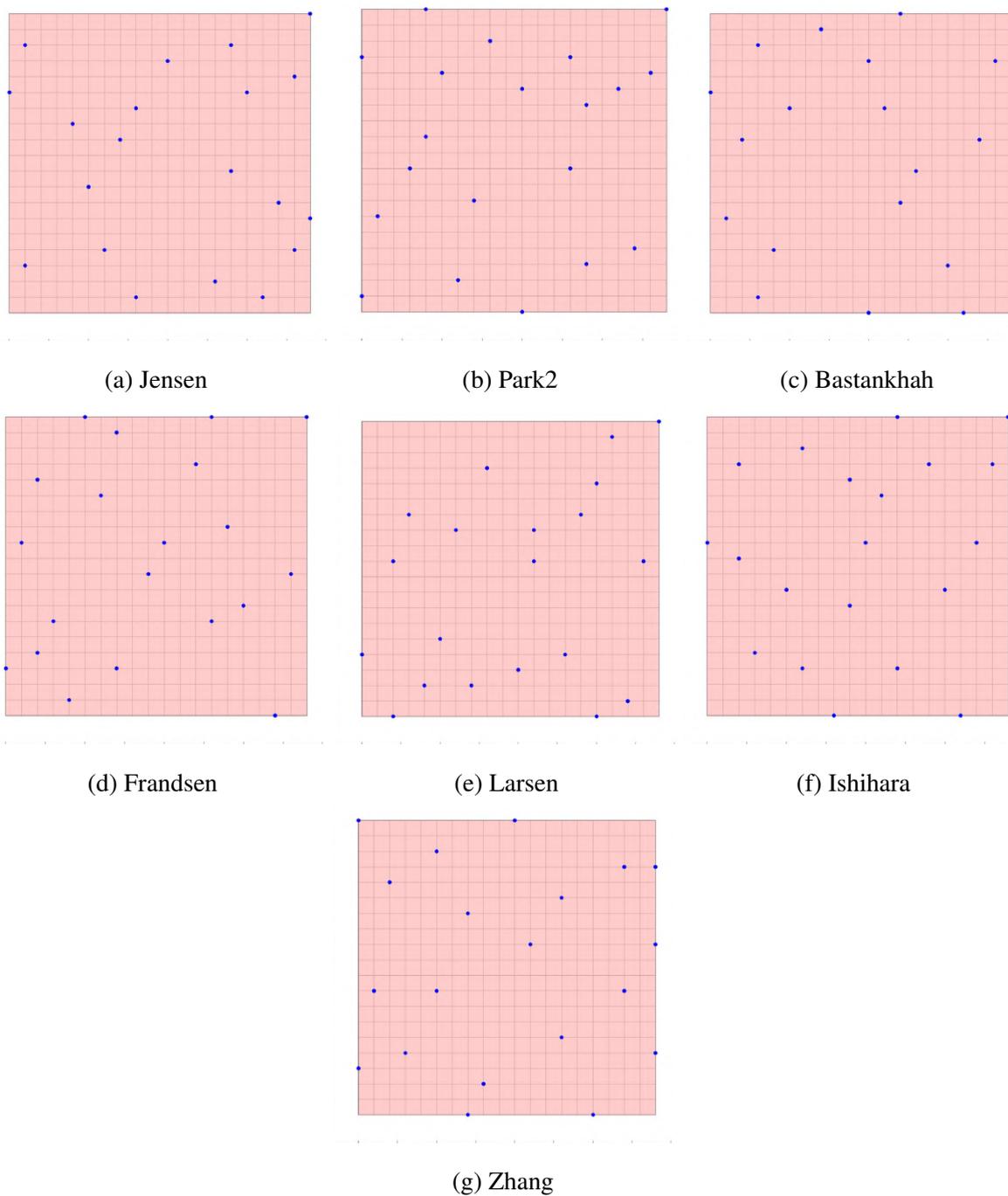


Figura F.9: Resultados Algoritmo CRO-SL para terreno de 20×20 con viento sintético.

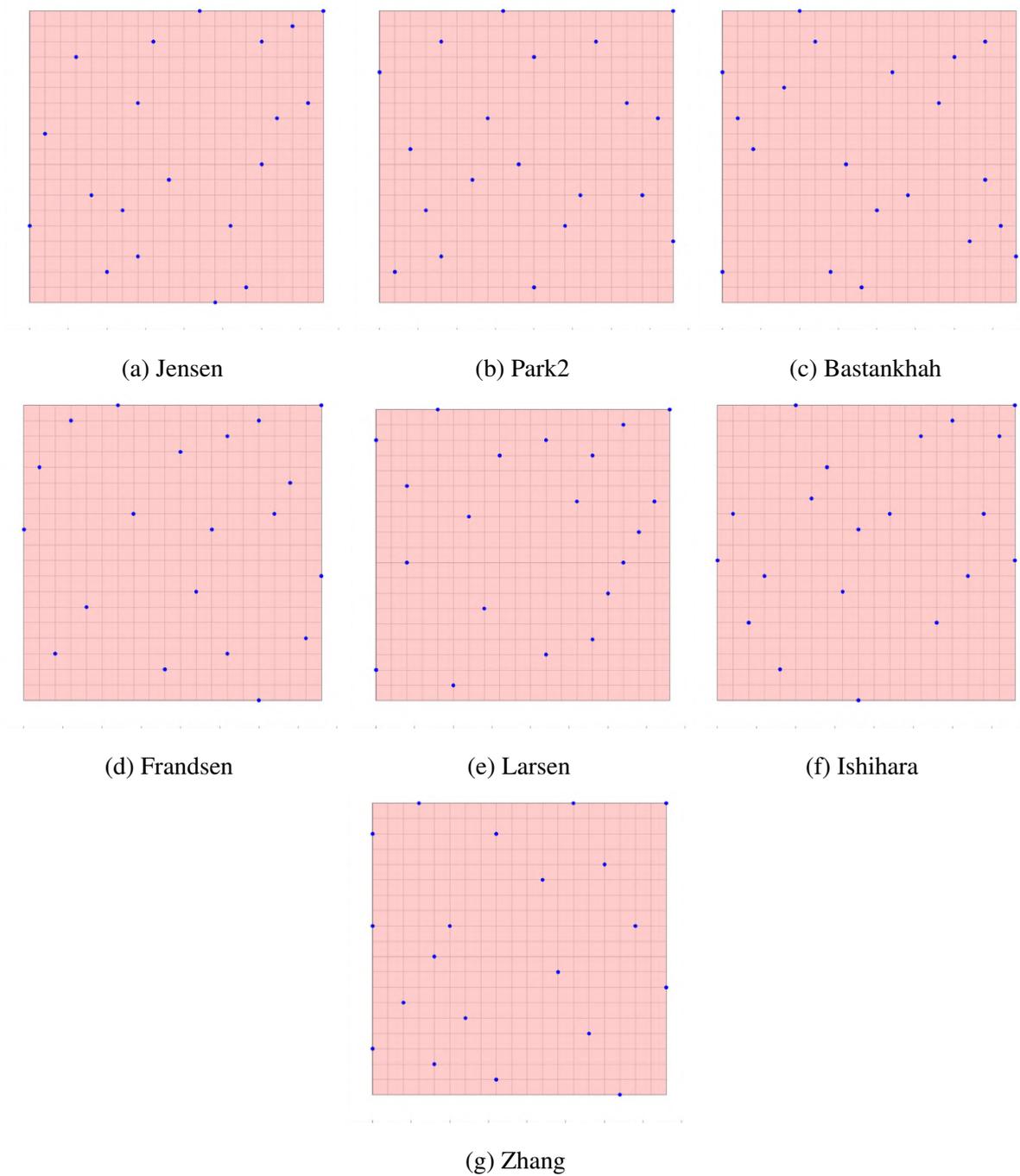


Figura F.10: Resultados Algoritmo CRO-SL para terreno de 20×20 con viento real.

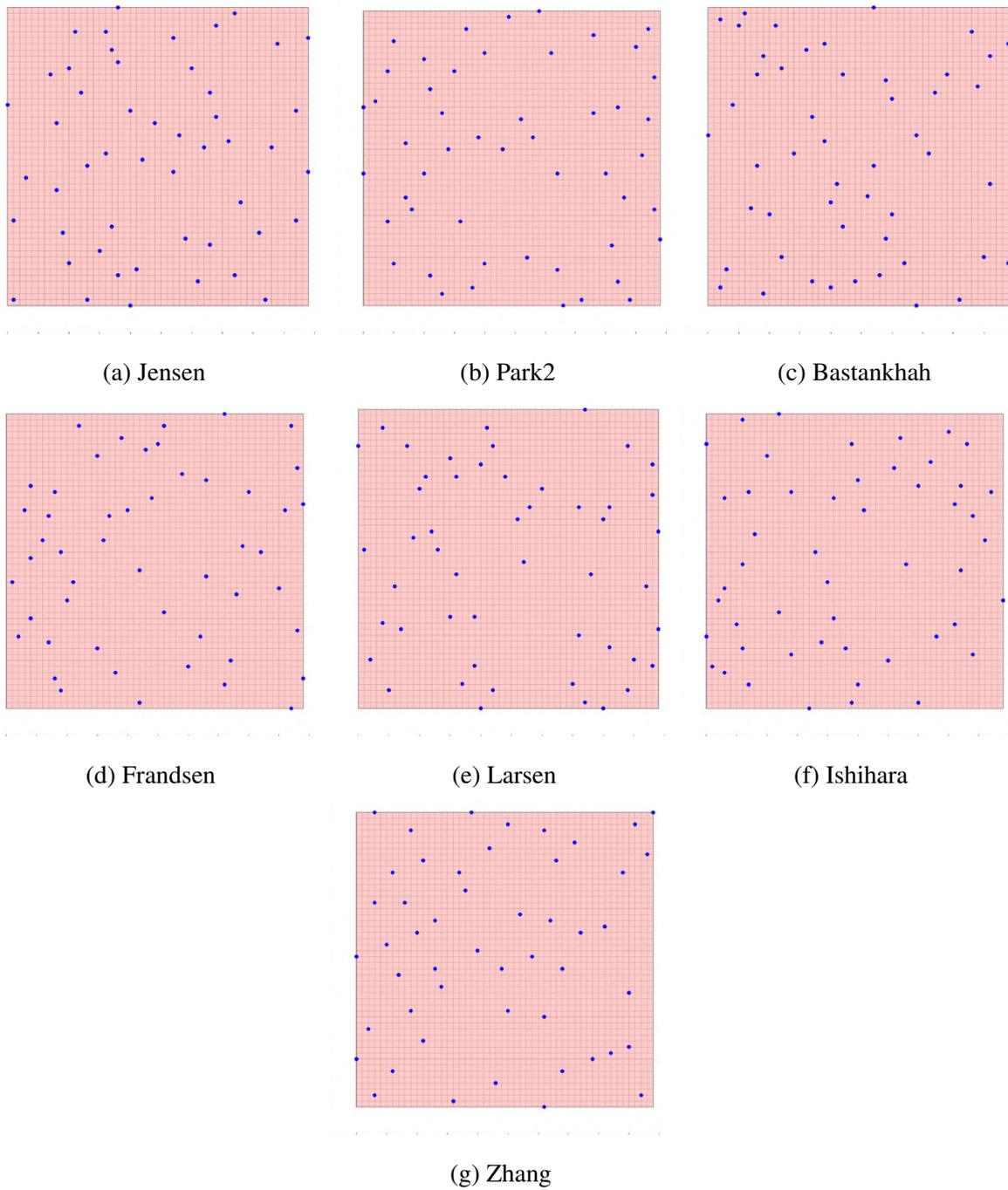


Figura F.11: Resultados Algoritmo CRO-SL para terreno de 50×50 con viento sintético.

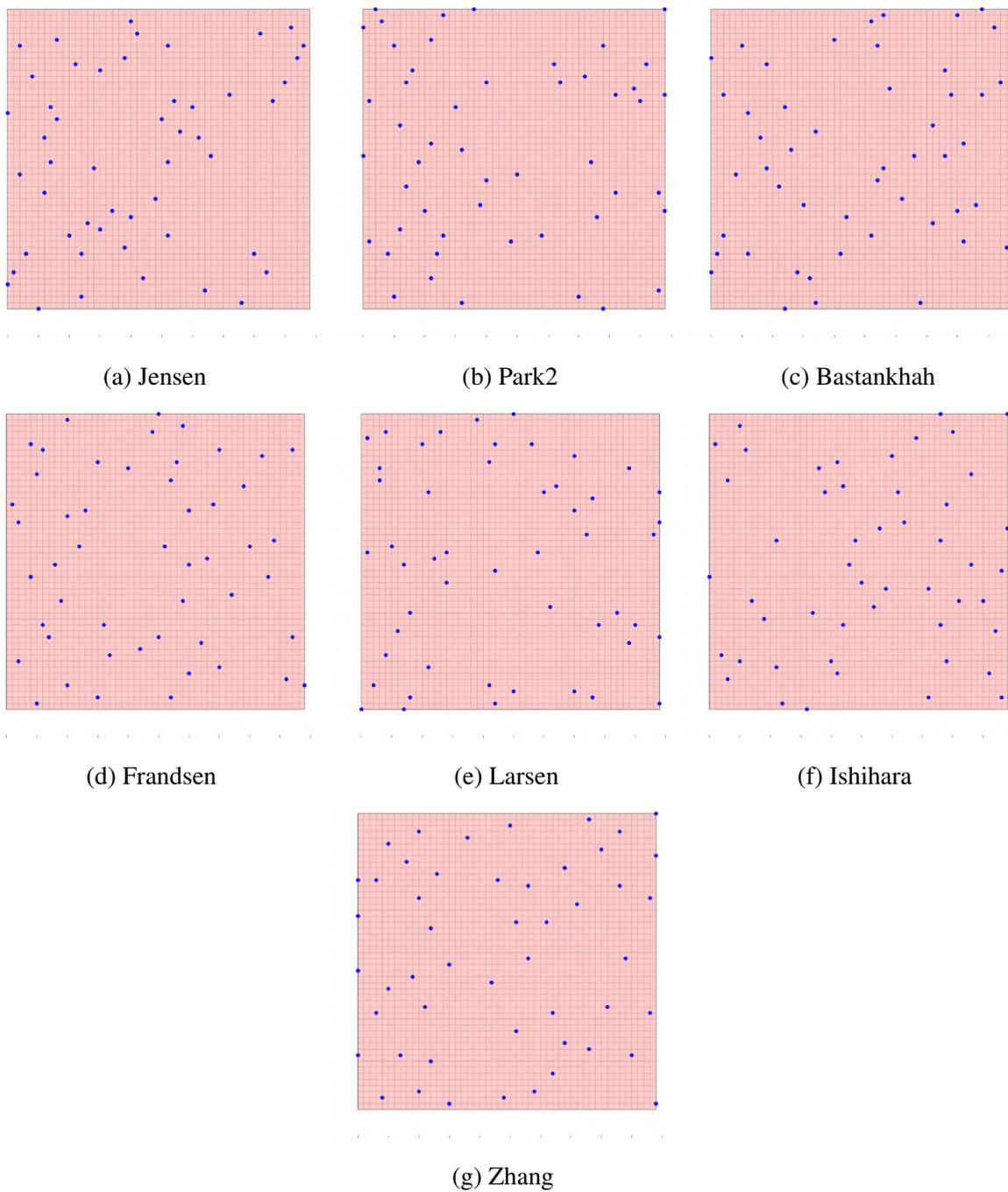


Figura F.12: Resultados Algoritmo CRO-SL para terreno de 50×50 con viento real.

Apéndice G

Resultados de *fitness* de las soluciones cruzadas

Tabla G.1: Soluciones cruzadas de *fitness* de BGA con configuración de 20×20 y viento sintético.

Soluciones / Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang
Jensen	$5.1161 \cdot 10^7$	$5.102 \cdot 10^7$	$5.1076 \cdot 10^7$	$5.1079 \cdot 10^7$	$5.0795 \cdot 10^7$	$5.1138 \cdot 10^7$	$5.088 \cdot 10^7$
Park2	$5.1174 \cdot 10^7$	$5.1163 \cdot 10^7$	$5.1121 \cdot 10^7$	$5.1179 \cdot 10^7$	$5.0872 \cdot 10^7$	$5.1147 \cdot 10^7$	$5.0853 \cdot 10^7$
Bastankhah	$5.089 \cdot 10^7$	$5.0936 \cdot 10^7$	$5.0827 \cdot 10^7$	$5.0888 \cdot 10^7$	$4.9982 \cdot 10^7$	$5.0977 \cdot 10^7$	$4.9911 \cdot 10^7$
Frandsen	$5.113 \cdot 10^7$	$5.1107 \cdot 10^7$	$5.1054 \cdot 10^7$	$5.1136 \cdot 10^7$	$5.0709 \cdot 10^7$	$5.1086 \cdot 10^7$	$5.0682 \cdot 10^7$
Larsen	$4.6604 \cdot 10^7$	$4.8719 \cdot 10^7$	$4.8219 \cdot 10^7$	$4.7246 \cdot 10^7$	$5.0213 \cdot 10^7$	$4.7794 \cdot 10^7$	$4.7876 \cdot 10^7$
Ishihara	$5.1154 \cdot 10^7$	$5.1121 \cdot 10^7$	$5.1099 \cdot 10^7$	$5.1131 \cdot 10^7$	$5.0685 \cdot 10^7$	$5.1155 \cdot 10^7$	$5.0767 \cdot 10^7$
Zhang	$4.9104 \cdot 10^7$	$4.9411 \cdot 10^7$	$4.9471 \cdot 10^7$	$4.9351 \cdot 10^7$	$4.9679 \cdot 10^7$	$4.9348 \cdot 10^7$	$4.9934 \cdot 10^7$

Tabla G.2: Soluciones cruzadas de *fitness* de BGA con configuración de 20×20 y viento real.

Soluciones / Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang
Jensen	$5.1896 \cdot 10^7$	$5.1869 \cdot 10^7$	$5.1873 \cdot 10^7$	$5.1846 \cdot 10^7$	$5.1209 \cdot 10^7$	$5.1831 \cdot 10^7$	$5.1513 \cdot 10^7$
Park2	$5.1893 \cdot 10^7$	$5.1941 \cdot 10^7$	$5.1911 \cdot 10^7$	$5.1937 \cdot 10^7$	$5.1256 \cdot 10^7$	$5.191 \cdot 10^7$	$5.1487 \cdot 10^7$
Bastankhah	$5.1605 \cdot 10^7$	$5.169 \cdot 10^7$	$5.168 \cdot 10^7$	$5.1705 \cdot 10^7$	$4.9691 \cdot 10^7$	$5.1702 \cdot 10^7$	$5.0531 \cdot 10^7$
Frandsen	$5.1818 \cdot 10^7$	$5.1898 \cdot 10^7$	$5.1848 \cdot 10^7$	$5.1893 \cdot 10^7$	$5.0941 \cdot 10^7$	$5.1846 \cdot 10^7$	$5.1269 \cdot 10^7$
Larsen	$4.9136 \cdot 10^7$	$4.8333 \cdot 10^7$	$4.8083 \cdot 10^7$	$4.8654 \cdot 10^7$	$4.8653 \cdot 10^7$	$4.8398 \cdot 10^7$	$4.8793 \cdot 10^7$
Ishihara	$5.1879 \cdot 10^7$	$5.1908 \cdot 10^7$	$5.1893 \cdot 10^7$	$5.1895 \cdot 10^7$	$5.0802 \cdot 10^7$	$5.189 \cdot 10^7$	$5.1327 \cdot 10^7$
Zhang	$5.0437 \cdot 10^7$	$5.0131 \cdot 10^7$	$5.0398 \cdot 10^7$	$5.0408 \cdot 10^7$	$4.9991 \cdot 10^7$	$5.022 \cdot 10^7$	$5.0774 \cdot 10^7$

XXXII APÉNDICE G. RESULTADOS DE FITNESS DE LAS SOLUCIONES CRUZADAS

Tabla G.3: Soluciones cruzadas de *fitness* de BGA con configuración de 50×50 y viento sintético.

Soluciones / Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang
Jensen	1.2775·10 ⁸	1.2776·10 ⁸	1.276·10 ⁸	1.2761·10 ⁸	1.2715·10 ⁸	1.2775·10 ⁸	1.2765·10 ⁸
Park2	1.2772·10 ⁸	1.2778·10 ⁸	1.2766·10 ⁸	1.2771·10 ⁸	1.2718·10 ⁸	1.2774·10 ⁸	1.2759·10 ⁸
Bastankhah	1.2706·10 ⁸	1.2725·10 ⁸	1.2689·10 ⁸	1.2698·10 ⁸	1.2536·10 ⁸	1.2723·10 ⁸	1.2673·10 ⁸
Frandsen	1.2752·10 ⁸	1.276·10 ⁸	1.2744·10 ⁸	1.275·10 ⁸	1.2676·10 ⁸	1.2754·10 ⁸	1.2732·10 ⁸
Larsen	1.1699·10 ⁸	1.1617·10 ⁸	1.1644·10 ⁸	1.1509·10 ⁸	1.1956·10 ⁸	1.1624·10 ⁸	1.1709·10 ⁸
Ishihara	1.277·10 ⁸	1.2777·10 ⁸	1.2762·10 ⁸	1.2765·10 ⁸	1.2681·10 ⁸	1.2775·10 ⁸	1.2756·10 ⁸
Zhang	1.2412·10 ⁸	1.2461·10 ⁸	1.249·10 ⁸	1.2469·10 ⁸	1.2488·10 ⁸	1.2479·10 ⁸	1.2558·10 ⁸

Tabla G.4: Soluciones cruzadas de *fitness* de BGA con configuración de 50×50 y viento real.

Soluciones / Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang
Jensen	1.2963·10 ⁸	1.2949·10 ⁸	1.2948·10 ⁸	1.2956·10 ⁸	1.287·10 ⁸	1.296·10 ⁸	1.2945·10 ⁸
Park2	1.2957·10 ⁸	1.2959·10 ⁸	1.296·10 ⁸	1.2966·10 ⁸	1.2865·10 ⁸	1.2959·10 ⁸	1.2936·10 ⁸
Bastankhah	1.2889·10 ⁸	1.2882·10 ⁸	1.2891·10 ⁸	1.2896·10 ⁸	1.2594·10 ⁸	1.2895·10 ⁸	1.2819·10 ⁸
Frandsen	1.2934·10 ⁸	1.2938·10 ⁸	1.2938·10 ⁸	1.2947·10 ⁸	1.2808·10 ⁸	1.2938·10 ⁸	1.2905·10 ⁸
Larsen	1.1795·10 ⁸	1.1791·10 ⁸	1.1842·10 ⁸	1.1828·10 ⁸	1.219·10 ⁸	1.1862·10 ⁸	1.1745·10 ⁸
Ishihara	1.2957·10 ⁸	1.2953·10 ⁸	1.2953·10 ⁸	1.296·10 ⁸	1.2817·10 ⁸	1.2958·10 ⁸	1.2932·10 ⁸
Zhang	1.2691·10 ⁸	1.2681·10 ⁸	1.2661·10 ⁸	1.2633·10 ⁸	1.2672·10 ⁸	1.2679·10 ⁸	1.2731·10 ⁸

Tabla G.5: Soluciones cruzadas de *fitness* de PSO con configuración de 20×20 y viento sintético.

Soluciones / Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang
Jensen	4.9921·10 ⁷	4.9861·10 ⁷	4.9939·10 ⁷	4.9734·10 ⁷	5.0444·10 ⁷	5.0193·10 ⁷	4.9993·10 ⁷
Park2	5.0018·10 ⁷	4.9985·10 ⁷	5.0023·10 ⁷	4.9898·10 ⁷	5.0415·10 ⁷	5.0203·10 ⁷	5.0137·10 ⁷
Bastankhah	4.7143·10 ⁷	4.7496·10 ⁷	4.7489·10 ⁷	4.7425·10 ⁷	4.8963·10 ⁷	4.8017·10 ⁷	4.7848·10 ⁷
Frandsen	4.9528·10 ⁷	4.9482·10 ⁷	4.9526·10 ⁷	4.9333·10 ⁷	5.007·10 ⁷	4.9783·10 ⁷	4.967·10 ⁷
Larsen	4.6817·10 ⁷	4.7949·10 ⁷	4.6828·10 ⁷	4.7576·10 ⁷	4.7022·10 ⁷	4.614·10 ⁷	4.7293·10 ⁷
Ishihara	4.9062·10 ⁷	4.8984·10 ⁷	4.9114·10 ⁷	4.8689·10 ⁷	4.9977·10 ⁷	4.9593·10 ⁷	4.9133·10 ⁷
Zhang	4.8688·10 ⁷	4.9052·10 ⁷	4.907·10 ⁷	4.9022·10 ⁷	4.9161·10 ⁷	4.8585·10 ⁷	4.874·10 ⁷

Tabla G.6: Soluciones cruzadas de *fitness* de PSO con configuración de 20×20 y viento real.

Soluciones / Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang
Jensen	$5.0032 \cdot 10^7$	$5.0728 \cdot 10^7$	$5.0978 \cdot 10^7$	$5.1111 \cdot 10^7$	$5.1062 \cdot 10^7$	$5.026 \cdot 10^7$	$5.1009 \cdot 10^7$
Park2	$5.0179 \cdot 10^7$	$5.0776 \cdot 10^7$	$5.1088 \cdot 10^7$	$5.1081 \cdot 10^7$	$5.1052 \cdot 10^7$	$5.0305 \cdot 10^7$	$5.094 \cdot 10^7$
Bastankhah	$4.6324 \cdot 10^7$	$4.7804 \cdot 10^7$	$4.8991 \cdot 10^7$	$4.9046 \cdot 10^7$	$4.8921 \cdot 10^7$	$4.6928 \cdot 10^7$	$4.8867 \cdot 10^7$
Frandsen	$4.9437 \cdot 10^7$	$5.0293 \cdot 10^7$	$5.0709 \cdot 10^7$	$5.0714 \cdot 10^7$	$5.0669 \cdot 10^7$	$4.9614 \cdot 10^7$	$5.0512 \cdot 10^7$
Larsen	$4.7578 \cdot 10^7$	$4.8124 \cdot 10^7$	$4.749 \cdot 10^7$	$4.8069 \cdot 10^7$	$4.7555 \cdot 10^7$	$4.7027 \cdot 10^7$	$4.8214 \cdot 10^7$
Ishihara	$4.863 \cdot 10^7$	$5.0076 \cdot 10^7$	$5.0427 \cdot 10^7$	$5.0589 \cdot 10^7$	$5.0537 \cdot 10^7$	$4.9015 \cdot 10^7$	$5.0402 \cdot 10^7$
Zhang	$4.9368 \cdot 10^7$	$4.9615 \cdot 10^7$	$4.9587 \cdot 10^7$	$5.0105 \cdot 10^7$	$5.0149 \cdot 10^7$	$4.9587 \cdot 10^7$	$4.9941 \cdot 10^7$

Tabla G.7: Soluciones cruzadas de *fitness* de PSO con configuración de 50×50 y viento sintético.

Soluciones / Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang
Jensen	$1.2666 \cdot 10^8$	$1.2587 \cdot 10^8$	$1.2606 \cdot 10^8$	$1.2581 \cdot 10^8$	$1.267 \cdot 10^8$	$1.259 \cdot 10^8$	$1.2609 \cdot 10^8$
Park2	$1.2641 \cdot 10^8$	$1.2593 \cdot 10^8$	$1.2618 \cdot 10^8$	$1.2598 \cdot 10^8$	$1.2661 \cdot 10^8$	$1.2591 \cdot 10^8$	$1.2602 \cdot 10^8$
Bastankhah	$1.2373 \cdot 10^8$	$1.2221 \cdot 10^8$	$1.2223 \cdot 10^8$	$1.2161 \cdot 10^8$	$1.2403 \cdot 10^8$	$1.2164 \cdot 10^8$	$1.2211 \cdot 10^8$
Frandsen	$1.2571 \cdot 10^8$	$1.2502 \cdot 10^8$	$1.2537 \cdot 10^8$	$1.2509 \cdot 10^8$	$1.2599 \cdot 10^8$	$1.2499 \cdot 10^8$	$1.2515 \cdot 10^8$
Larsen	$1.136 \cdot 10^8$	$1.151 \cdot 10^8$	$1.1629 \cdot 10^8$	$1.1557 \cdot 10^8$	$1.1665 \cdot 10^8$	$1.1505 \cdot 10^8$	$1.1684 \cdot 10^8$
Ishihara	$1.2581 \cdot 10^8$	$1.2451 \cdot 10^8$	$1.2493 \cdot 10^8$	$1.2445 \cdot 10^8$	$1.26 \cdot 10^8$	$1.2452 \cdot 10^8$	$1.2498 \cdot 10^8$
Zhang	$1.2437 \cdot 10^8$	$1.2345 \cdot 10^8$	$1.2413 \cdot 10^8$	$1.2376 \cdot 10^8$	$1.2444 \cdot 10^8$	$1.2365 \cdot 10^8$	$1.2404 \cdot 10^8$

Tabla G.8: Soluciones cruzadas de *fitness* de PSO con configuración de 50×50 y viento real.

Soluciones / Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang
Jensen	$1.2814 \cdot 10^8$	$1.2725 \cdot 10^8$	$1.2749 \cdot 10^8$	$1.2802 \cdot 10^8$	$1.282 \cdot 10^8$	$1.2789 \cdot 10^8$	$1.2773 \cdot 10^8$
Park2	$1.2803 \cdot 10^8$	$1.2731 \cdot 10^8$	$1.2737 \cdot 10^8$	$1.2801 \cdot 10^8$	$1.2819 \cdot 10^8$	$1.2792 \cdot 10^8$	$1.2771 \cdot 10^8$
Bastankhah	$1.2433 \cdot 10^8$	$1.2157 \cdot 10^8$	$1.2333 \cdot 10^8$	$1.2424 \cdot 10^8$	$1.2508 \cdot 10^8$	$1.24 \cdot 10^8$	$1.2344 \cdot 10^8$
Frandsen	$1.272 \cdot 10^8$	$1.262 \cdot 10^8$	$1.2631 \cdot 10^8$	$1.2716 \cdot 10^8$	$1.2743 \cdot 10^8$	$1.2703 \cdot 10^8$	$1.2673 \cdot 10^8$
Larsen	$1.1597 \cdot 10^8$	$1.1628 \cdot 10^8$	$1.1682 \cdot 10^8$	$1.1764 \cdot 10^8$	$1.1701 \cdot 10^8$	$1.1703 \cdot 10^8$	$1.1741 \cdot 10^8$
Ishihara	$1.2698 \cdot 10^8$	$1.2543 \cdot 10^8$	$1.2574 \cdot 10^8$	$1.2682 \cdot 10^8$	$1.2708 \cdot 10^8$	$1.2653 \cdot 10^8$	$1.2613 \cdot 10^8$
Zhang	$1.2557 \cdot 10^8$	$1.248 \cdot 10^8$	$1.2567 \cdot 10^8$	$1.2588 \cdot 10^8$	$1.2599 \cdot 10^8$	$1.2557 \cdot 10^8$	$1.2592 \cdot 10^8$

XXXIV APÉNDICE G. RESULTADOS DE FITNESS DE LAS SOLUCIONES CRUZADAS

Tabla G.9: Soluciones cruzadas de *fitness* de CRO-SL con configuración de 20×20 y viento sintético.

Soluciones / Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang
Jensen	$5.1076 \cdot 10^7$	$5.1037 \cdot 10^7$	$5.1011 \cdot 10^7$	$5.1003 \cdot 10^7$	$5.0756 \cdot 10^7$	$5.0993 \cdot 10^7$	$5.0751 \cdot 10^7$
Park2	$5.1085 \cdot 10^7$	$5.1122 \cdot 10^7$	$5.1108 \cdot 10^7$	$5.1083 \cdot 10^7$	$5.0778 \cdot 10^7$	$5.0988 \cdot 10^7$	$5.079 \cdot 10^7$
Bastankhah	$5.0742 \cdot 10^7$	$5.0719 \cdot 10^7$	$5.0666 \cdot 10^7$	$5.0703 \cdot 10^7$	$4.983 \cdot 10^7$	$5.0327 \cdot 10^7$	$4.979 \cdot 10^7$
Frandsen	$5.1002 \cdot 10^7$	$5.1057 \cdot 10^7$	$5.1033 \cdot 10^7$	$5.1002 \cdot 10^7$	$5.0582 \cdot 10^7$	$5.0869 \cdot 10^7$	$5.0597 \cdot 10^7$
Larsen	$4.7484 \cdot 10^7$	$4.7538 \cdot 10^7$	$4.8012 \cdot 10^7$	$4.7463 \cdot 10^7$	$4.9699 \cdot 10^7$	$4.7117 \cdot 10^7$	$4.8537 \cdot 10^7$
Ishihara	$5.108 \cdot 10^7$	$5.1069 \cdot 10^7$	$5.1037 \cdot 10^7$	$5.1031 \cdot 10^7$	$5.0574 \cdot 10^7$	$5.0931 \cdot 10^7$	$5.0623 \cdot 10^7$
Zhang	$4.9474 \cdot 10^7$	$4.9516 \cdot 10^7$	$4.9621 \cdot 10^7$	$4.9508 \cdot 10^7$	$4.9464 \cdot 10^7$	$4.9277 \cdot 10^7$	$4.99 \cdot 10^7$

Tabla G.10: Soluciones cruzadas de *fitness* de CRO-SL con configuración de 20×20 y viento real.

Soluciones / Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang
Jensen	$5.1846 \cdot 10^7$	$5.1798 \cdot 10^7$	$5.1848 \cdot 10^7$	$5.1847 \cdot 10^7$	$5.1669 \cdot 10^7$	$5.1811 \cdot 10^7$	$5.1641 \cdot 10^7$
Park2	$5.1829 \cdot 10^7$	$5.1904 \cdot 10^7$	$5.1881 \cdot 10^7$	$5.1894 \cdot 10^7$	$5.1724 \cdot 10^7$	$5.1847 \cdot 10^7$	$5.1677 \cdot 10^7$
Bastankhah	$5.1458 \cdot 10^7$	$5.1571 \cdot 10^7$	$5.1603 \cdot 10^7$	$5.1604 \cdot 10^7$	$5.0951 \cdot 10^7$	$5.143 \cdot 10^7$	$5.097 \cdot 10^7$
Frandsen	$5.174 \cdot 10^7$	$5.1847 \cdot 10^7$	$5.1805 \cdot 10^7$	$5.1831 \cdot 10^7$	$5.1598 \cdot 10^7$	$5.1768 \cdot 10^7$	$5.1515 \cdot 10^7$
Larsen	$4.825 \cdot 10^7$	$4.8844 \cdot 10^7$	$4.8817 \cdot 10^7$	$4.8771 \cdot 10^7$	$5.0278 \cdot 10^7$	$4.8877 \cdot 10^7$	$4.9622 \cdot 10^7$
Ishihara	$5.1822 \cdot 10^7$	$5.1849 \cdot 10^7$	$5.1863 \cdot 10^7$	$5.1875 \cdot 10^7$	$5.1631 \cdot 10^7$	$5.1806 \cdot 10^7$	$5.1593 \cdot 10^7$
Zhang	$5.0205 \cdot 10^7$	$5.0383 \cdot 10^7$	$5.0102 \cdot 10^7$	$5.0411 \cdot 10^7$	$5.0299 \cdot 10^7$	$5.0245 \cdot 10^7$	$5.0668 \cdot 10^7$

Tabla G.11: Soluciones cruzadas de *fitness* de CRO-SL con configuración de 50×50 y viento sintético.

Soluciones / Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang
Jensen	$1.2771 \cdot 10^8$	$1.2769 \cdot 10^8$	$1.2774 \cdot 10^8$	$1.277 \cdot 10^8$	$1.2747 \cdot 10^8$	$1.2768 \cdot 10^8$	$1.2761 \cdot 10^8$
Park2	$1.2767 \cdot 10^8$	$1.2772 \cdot 10^8$	$1.2779 \cdot 10^8$	$1.2782 \cdot 10^8$	$1.2743 \cdot 10^8$	$1.2776 \cdot 10^8$	$1.2754 \cdot 10^8$
Bastankhah	$1.2703 \cdot 10^8$	$1.27 \cdot 10^8$	$1.2723 \cdot 10^8$	$1.2731 \cdot 10^8$	$1.2633 \cdot 10^8$	$1.2721 \cdot 10^8$	$1.2676 \cdot 10^8$
Frandsen	$1.2745 \cdot 10^8$	$1.2752 \cdot 10^8$	$1.2761 \cdot 10^8$	$1.2766 \cdot 10^8$	$1.2711 \cdot 10^8$	$1.2757 \cdot 10^8$	$1.2725 \cdot 10^8$
Larsen	$1.1675 \cdot 10^8$	$1.1723 \cdot 10^8$	$1.1556 \cdot 10^8$	$1.1626 \cdot 10^8$	$1.2127 \cdot 10^8$	$1.1653 \cdot 10^8$	$1.1712 \cdot 10^8$
Ishihara	$1.2768 \cdot 10^8$	$1.2768 \cdot 10^8$	$1.2776 \cdot 10^8$	$1.2776 \cdot 10^8$	$1.2737 \cdot 10^8$	$1.2773 \cdot 10^8$	$1.2755 \cdot 10^8$
Zhang	$1.2493 \cdot 10^8$	$1.2501 \cdot 10^8$	$1.247 \cdot 10^8$	$1.2479 \cdot 10^8$	$1.2472 \cdot 10^8$	$1.2491 \cdot 10^8$	$1.2561 \cdot 10^8$

Tabla G.12: Soluciones cruzadas de *fitness* de CRO-SL con configuración de 50×50 y viento real.

Soluciones Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang
Jensen	$1.2967 \cdot 10^8$	$1.2956 \cdot 10^8$	$1.2953 \cdot 10^8$	$1.2962 \cdot 10^8$	$1.2935 \cdot 10^8$	$1.2954 \cdot 10^8$	$1.2929 \cdot 10^8$
Park2	$1.2962 \cdot 10^8$	$1.2966 \cdot 10^8$	$1.2958 \cdot 10^8$	$1.2971 \cdot 10^8$	$1.2933 \cdot 10^8$	$1.2958 \cdot 10^8$	$1.2928 \cdot 10^8$
Bastankhah	$1.2907 \cdot 10^8$	$1.2904 \cdot 10^8$	$1.2889 \cdot 10^8$	$1.2919 \cdot 10^8$	$1.2804 \cdot 10^8$	$1.2879 \cdot 10^8$	$1.2821 \cdot 10^8$
Frandsen	$1.2941 \cdot 10^8$	$1.2947 \cdot 10^8$	$1.2937 \cdot 10^8$	$1.2955 \cdot 10^8$	$1.2902 \cdot 10^8$	$1.2936 \cdot 10^8$	$1.2893 \cdot 10^8$
Larsen	$1.1681 \cdot 10^8$	$1.1830 \cdot 10^8$	$1.2004 \cdot 10^8$	$1.1682 \cdot 10^8$	$1.2225 \cdot 10^8$	$1.1799 \cdot 10^8$	$1.2028 \cdot 10^8$
Ishihara	$1.2963 \cdot 10^8$	$1.296 \cdot 10^8$	$1.2954 \cdot 10^8$	$1.2967 \cdot 10^8$	$1.2916 \cdot 10^8$	$1.2954 \cdot 10^8$	$1.2916 \cdot 10^8$
Zhang	$1.2613 \cdot 10^8$	$1.2661 \cdot 10^8$	$1.2649 \cdot 10^8$	$1.2662 \cdot 10^8$	$1.2681 \cdot 10^8$	$1.2673 \cdot 10^8$	$1.2759 \cdot 10^8$

XXXVI APÉNDICE G. RESULTADOS DE FITNESS DE LAS SOLUCIONES CRUZADAS

Apéndice H

Resultados de rendimiento de las soluciones cruzadas

Tabla H.1: Rendimiento cruzado de BGA con configuración de 20×20 y viento sintético.

Modelos \ Soluciones	Soluciones							Rendimiento por modelo
	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang	
Jensen	99.49 %	98.91 %	99.14 %	99.15 %	97.98 %	99.4 %	98.33 %	98.92 %
Park2	99.71 %	99.68 %	99.56 %	99.72 %	98.89 %	99.63 %	98.84 %	99.43 %
Bastankhah	99.09 %	99.19 %	98.94 %	99.08 %	96.97 %	99.29 %	96.81 %	98.48 %
Frandsen	99.66 %	99.60 %	99.48 %	99.67 %	98.71 %	99.56 %	98.65 %	99.33 %
Larsen	88.85 %	93.89 %	92.70 %	90.38 %	97.45 %	91.69 %	91.88 %	92.40 %
Ishihara	99.73 %	99.66 %	99.62 %	99.68 %	98.75 %	99.73 %	98.92 %	99.44 %
Zhang	93.81 %	94.68 %	94.85 %	94.51 %	95.44 %	94.50 %	96.16 %	94.85 %
Rendimiento por solución	97.19 %	97.94 %	97.76 %	97.46 %	97.74 %	97.68 %	97.08 %	%

Tabla H.2: Rendimiento cruzado de BGA con configuración de 20×20 y viento real.

Modelos \ Soluciones	Soluciones							Rendimiento por modelo
	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang	
Jensen	99.40 %	99.29 %	99.30 %	99.19 %	96.57 %	99.13 %	97.82 %	98.67 %
Park2	99.60 %	99.73 %	99.65 %	99.72 %	97.89 %	99.64 %	98.51 %	99.25 %
Bastankhah	98.99 %	99.19 %	99.17 %	99.22 %	94.59 %	99.22 %	96.52 %	98.13 %
Frandsen	99.50 %	99.68 %	99.57 %	99.67 %	97.56 %	99.56 %	98.28 %	99.12 %
Larsen	93.16 %	91.27 %	90.68 %	92.03 %	92.02 %	91.42 %	92.35 %	91.85 %
Ishihara	99.66 %	99.72 %	99.69 %	99.70 %	97.45 %	99.69 %	98.53 %	99.21 %
Zhang	95.48 %	94.62 %	95.37 %	95.40 %	94.23 %	94.87 %	96.43 %	95.20 %
Rendimiento por solución	97.97 %	97.64 %	97.63 %	97.85 %	95.76 %	97.65 %	96.92 %	%

Tabla H.3: Rendimiento cruzado de BGA con configuración de 50×50 y viento sintético.

Soluciones Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang	Rendimiento por modelo
Jensen	99.27 %	99.28 %	99.03 %	99.04 %	98.32 %	99.26 %	99.11 %	99.05 %
Park2	99.53 %	99.59 %	99.48 %	99.52 %	99.01 %	99.55 %	99.41 %	99.44 %
Bastankhah	98.96 %	99.14 %	98.81 %	98.90 %	97.44 %	99.13 %	98.67 %	98.72 %
Frandsen	99.43 %	99.50 %	99.36 %	99.41 %	98.80 %	99.44 %	99.27 %	99.31 %
Larsen	90.56 %	89.87 %	90.10 %	88.97 %	92.73 %	89.93 %	90.65 %	90.40 %
Ishihara	99.59 %	99.65 %	99.53 %	99.55 %	98.88 %	99.63 %	99.48 %	99.47 %
Zhang	95.96 %	96.45 %	96.74 %	96.53 %	96.71 %	96.62 %	97.40 %	96.63 %
Rendimiento por solución	97.62 %	97.64 %	97.58 %	97.42 %	97.41 %	97.65 %	97.71 %	%

Tabla H.4: Rendimiento cruzado de BGA con configuración de 50×50 y viento real.

Soluciones Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang	Rendimiento por modelo
Jensen	99.24 %	99.02 %	99.00 %	99.13 %	97.76 %	99.19 %	98.95 %	98.90 %
Park2	99.49 %	99.51 %	99.52 %	99.58 %	98.63 %	99.51 %	99.29 %	99.36 %
Bastankhah	98.92 %	98.86 %	98.93 %	98.98 %	96.30 %	98.97 %	98.30 %	98.47 %
Frandsen	99.37 %	99.40 %	99.41 %	99.48 %	98.35 %	99.40 %	99.14 %	99.22 %
Larsen	89.92 %	89.88 %	90.30 %	90.19 %	93.19 %	90.47 %	89.50 %	90.49 %
Ishihara	99.57 %	99.54 %	99.54 %	99.60 %	98.47 %	99.58 %	99.38 %	99.38 %
Zhang	96.88 %	96.78 %	96.59 %	96.31 %	96.69 %	96.75 %	97.27 %	96.75 %
Rendimiento por solución	97.63 %	97.57 %	97.61 %	97.61 %	97.05 %	97.70 %	97.40 %	%

Tabla H.5: Rendimiento cruzado de PSO con configuración de 20×20 y viento sintético.

Soluciones Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang	Rendimiento por modelo
Jensen	94.36 %	94.10 %	94.43 %	93.58 %	96.52 %	95.48 %	94.65 %	94.73 %
Park2	96.58 %	96.49 %	96.59 %	96.25 %	97.65 %	97.08 %	96.90 %	96.79 %
Bastankhah	90.37 %	91.19 %	91.17 %	91.02 %	94.60 %	92.40 %	92.01 %	91.82 %
Frandsen	96.06 %	95.95 %	96.05 %	95.62 %	97.28 %	96.63 %	96.38 %	96.28 %
Larsen	89.36 %	92.05 %	89.38 %	91.17 %	89.84 %	87.74 %	90.49 %	90.01 %
Ishihara	95.37 %	95.21 %	95.48 %	94.59 %	97.28 %	96.48 %	95.52 %	95.70 %
Zhang	92.62 %	93.66 %	93.71 %	93.57 %	93.97 %	92.33 %	92.77 %	93.23 %
Rendimiento por solución	93.53 %	94.09 %	93.83 %	93.69 %	95.31 %	94.02 %	94.10 %	%

Tabla H.6: Rendimiento cruzado de PSO con configuración de 20×20 y viento real.

Soluciones Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang	Rendimiento por modelo
Jensen	91.73 %	94.59 %	95.62 %	96.17 %	95.96 %	92.67 %	95.75 %	94.64 %
Park2	95.01 %	96.61 %	97.44 %	97.42 %	97.34 %	95.35 %	97.05 %	96.60 %
Bastankhah	86.85 %	90.25 %	92.98 %	93.11 %	92.82 %	88.24 %	92.70 %	90.99 %
Frandsen	94.22 %	96.12 %	97.04 %	97.05 %	96.95 %	94.62 %	96.61 %	96.09 %
Larsen	89.50 %	90.78 %	89.29 %	90.65 %	89.44 %	88.20 %	90.99 %	89.84 %
Ishihara	92.98 %	95.96 %	96.68 %	97.01 %	96.90 %	93.78 %	96.63 %	95.71 %
Zhang	92.47 %	93.17 %	93.09 %	94.55 %	94.67 %	93.09 %	94.08 %	93.59 %
Rendimiento por solución	91.82 %	93.92 %	94.59 %	95.14 %	94.87 %	92.28 %	94.83 %	%

Tabla H.7: Rendimiento cruzado de PSO con configuración de 50×50 y viento sintético.

Soluciones Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang	Rendimiento por modelo
Jensen	97.53 %	96.26 %	96.57 %	96.17 %	97.59 %	96.32 %	96.62 %	96.72 %
Park2	98.28 %	97.82 %	98.06 %	97.87 %	98.48 %	97.8 %	97.91 %	98.03 %
Bastankhah	95.98 %	94.62 %	94.64 %	94.08 %	96.25 %	94.11 %	94.53 %	94.89 %
Frandsen	97.93 %	97.36 %	97.65 %	97.42 %	98.17 %	97.34 %	97.47 %	97.62 %
Larsen	87.71 %	88.97 %	89.97 %	89.37 %	90.28 %	88.93 %	90.43 %	89.38 %
Ishihara	98.08 %	97.03 %	97.37 %	96.99 %	98.23 %	97.05 %	97.41 %	97.45 %
Zhang	96.21 %	95.30 %	95.97 %	95.61 %	96.28 %	95.50 %	95.89 %	95.82 %
Rendimiento por solución	95.96 %	95.34 %	95.75 %	95.36 %	96.47 %	95.29 %	95.75 %	%

Tabla H.8: Rendimiento cruzado de PSO con configuración de 50×50 y viento real.

Soluciones Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang	Rendimiento por modelo
Jensen	96.88 %	95.48 %	95.85 %	96.69 %	96.97 %	96.49 %	96.23 %	96.37 %
Park2	98.04 %	97.36 %	97.41 %	98.02 %	98.19 %	97.93 %	97.73 %	97.81 %
Bastankhah	94.88 %	92.43 %	93.99 %	94.80 %	95.54 %	94.59 %	94.09 %	94.33 %
Frandsen	97.63 %	96.81 %	96.90 %	97.60 %	97.82 %	97.49 %	97.25 %	97.36 %
Larsen	88.28 %	88.53 %	88.97 %	89.66 %	89.13 %	89.15 %	89.47 %	89.03 %
Ishihara	97.53 %	96.31 %	96.55 %	97.4 %	97.61 %	97.17 %	96.86 %	97.06 %
Zhang	95.57 %	94.81 %	95.67 %	95.87 %	95.98 %	95.57 %	95.91 %	95.63 %
Rendimiento por solución	95.54 %	94.53 %	95.05 %	95.72 %	95.89 %	95.48 %	95.36 %	%

XLAPÉNDICE H. RESULTADOS DE RENDIMIENTO DE LAS SOLUCIONES CRUZADAS

Tabla H.9: Rendimiento cruzado de CRO-SL con configuración de 20×20 y viento sintético.

Soluciones Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang	Rendimiento por modelo
Jensen	99.14 %	98.98 %	98.87 %	98.84 %	97.81 %	98.8 %	97.80 %	98.61 %
Park2	99.46 %	99.57 %	99.53 %	99.46 %	98.63 %	99.2 %	98.67 %	99.22 %
Bastankhah	98.74 %	98.69 %	98.56 %	98.65 %	96.62 %	97.77 %	96.53 %	97.94 %
Frandsen	99.37 %	99.49 %	99.44 %	99.37 %	98.43 %	99.07 %	98.46 %	99.09 %
Larsen	90.94 %	91.07 %	92.2 %	90.9 %	96.23 %	90.07 %	93.46 %	92.12 %
Ishihara	99.58 %	99.55 %	99.49 %	99.47 %	98.52 %	99.26 %	98.62 %	99.21 %
Zhang	94.86 %	94.98 %	95.27 %	94.95 %	94.83 %	94.3 %	96.07 %	95.04 %
Rendimiento por solución	97.44 %	97.48 %	97.62 %	97.38 %	97.3 %	96.93 %	97.09 %	%

Tabla H.10: Rendimiento cruzado de CRO-SL con configuración de 20×20 y viento real.

Soluciones Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang	Rendimiento por modelo
Jensen	99.19 %	98.99 %	99.20 %	99.19 %	98.46 %	99.04 %	98.35 %	98.92 %
Park2	99.43 %	99.63 %	99.57 %	99.60 %	99.15 %	99.48 %	99.02 %	99.41 %
Bastankhah	98.65 %	98.91 %	98.99 %	98.99 %	97.49 %	98.59 %	97.53 %	98.45 %
Frandsen	99.33 %	99.56 %	99.47 %	99.53 %	99.01 %	99.39 %	98.83 %	99.30 %
Larsen	91.08 %	92.47 %	92.41 %	92.30 %	95.85 %	92.55 %	94.30 %	93.00 %
Ishihara	99.54 %	99.60 %	99.63 %	99.66 %	99.15 %	99.51 %	99.08 %	99.45 %
Zhang	94.83 %	95.33 %	94.54 %	95.41 %	95.09 %	94.94 %	96.13 %	95.18 %
Rendimiento por solución	97.44 %	97.79 %	97.69 %	97.81 %	97.74 %	97.64 %	97.61 %	%

Tabla H.11: Rendimiento cruzado de CRO-SL con configuración de 50×50 y viento sintético.

Soluciones Modelos	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang	Rendimiento por modelo
Jensen	99.20 %	99.17 %	99.24 %	99.19 %	98.83 %	99.16 %	99.05 %	99.12 %
Park2	99.49 %	99.54 %	99.60 %	99.63 %	99.25 %	99.57 %	99.36 %	99.49 %
Bastankhah	98.94 %	98.92 %	99.12 %	99.20 %	98.32 %	99.10 %	98.70 %	98.90 %
Frandsen	99.37 %	99.43 %	99.50 %	99.55 %	99.09 %	99.47 %	99.21 %	99.37 %
Larsen	90.36 %	90.76 %	89.36 %	89.94 %	94.16 %	90.17 %	90.67 %	90.78 %
Ishihara	99.58 %	99.58 %	99.64 %	99.64 %	99.32 %	99.62 %	99.47 %	99.55 %
Zhang	96.76 %	96.85 %	96.54 %	96.62 %	96.56 %	96.74 %	97.43 %	96.79 %
Rendimiento por solución	97.67 %	97.75 %	97.57 %	97.68 %	97.93 %	97.69 %	97.70 %	%

Tabla H.12: Rendimiento cruzado de CRO-SL con configuración de 50×50 y viento real.

Modelos \ Soluciones	Soluciones							Rendimiento por modelo
	Jensen	Park2	Bastankhah	Frandsen	Larsen	Ishihara	Zhang	
Jensen	99.31 %	99.13 %	99.08 %	99.23 %	98.80 %	99.10 %	98.70 %	99.05 %
Park2	99.53 %	99.58 %	99.50 %	99.63 %	99.27 %	99.50 %	99.21 %	99.46 %
Bastankhah	99.07 %	99.05 %	98.91 %	99.19 %	98.16 %	98.83 %	98.31 %	98.79 %
Frandsen	99.43 %	99.48 %	99.39 %	99.54 %	99.11 %	99.39 %	99.04 %	99.34 %
Larsen	88.97 %	90.20 %	91.65 %	88.97 %	93.48 %	89.95 %	91.84 %	90.72 %
Ishihara	99.62 %	99.59 %	99.55 %	99.65 %	99.25 %	99.55 %	99.25 %	99.50 %
Zhang	96.12 %	96.59 %	96.47 %	96.60 %	96.78 %	96.70 %	97.54 %	96.69 %
Rendimiento por solución	97.44 %	97.66 %	97.79 %	97.54 %	97.84 %	97.57 %	97.70 %	%