



---

**Universidad de Valladolid**

ESCUELA DE INGENIERÍAS INDUSTRIALES

DEPARTAMENTO DE INGENIERÍA ENERGÉTICA Y FLUIDOMECÁNICA

TÍTULO:

**OPTIMIZACIÓN MEDIANTE ALGORITMOS  
GENÉTICOS DE LA CONFIGURACIÓN DE UN  
PARQUE EÓLICO**

Autor: Gonzalo Parrado Hernando

---

Tutores:

Francisco Castro Ruiz

Emilio Migoya Valor

Fecha:

28 de junio de 2017



*A mi abuelo Castor y al resto de mi familia, por el apoyo recibido y los trabajos domésticos que han absorbido mientras estudiaba. Reconocer especialmente el trabajo y la paciencia de mis tutores Francisco, Emilio y Jorge, tanto por los conocimientos transmitidos durante mi estancia en Madrid como durante la realización de la presente memoria, en la corrección de errores, sugerencias y otros saberes de la materia. El logro de los conocimientos adquiridos en ingeniería ha dependido de una gran lista de personas, a todas las que se sientan representadas mi total agradecimiento. Finalizar estas líneas admirando a todas las personas con las que tanto he compartido y de las que tanto he aprendido en los diversos colectivos de los que he participado. A todas ellas un fraternal abrazo. Salud.*



# Índice

Lista de figuras	V
Lista de tablas	VII
<b>1. Preliminares, resumen y palabras clave.</b>	<b>1</b>
<b>2. Introducción: contexto actual de la energía eólica. Límites y futuro.</b>	<b>3</b>
<b>3. Antecedentes del estado del conocimiento.</b>	<b>7</b>
3.1. Modelización del campo fluido en un parque eólico. . . . .	7
3.2. Antecedentes del estudio de optimización de parques eólicos. . . . .	13
<b>4. Marco del presente trabajo</b>	<b>15</b>
4.1. Códigos propios UPMPARK, UNION 2.2 y GRAFICA. . . . .	15
4.2. Código propio optUPMPARK para la optimización de parques eólicos. . . . .	18
4.2.1. Algoritmos genéticos como método de optimización de parques eólicos. . . . .	18
4.2.2. <i>Scripts</i> en Matlab y ficheros utilizados en el algoritmo genético. . . . .	21
<b>5. Simulaciones Realizadas</b>	<b>25</b>
5.1. Mejoras del código optUPMPARK. . . . .	25
5.1.1. Restricción de la distancia mínima entre aerogeneradores. . . . .	25
5.1.2. Almacenamiento de información del algoritmo genético. . . . .	27
5.2. Descripción del caso referencia. . . . .	28
5.3. Evaluación del caso referencia con UPMPARK. . . . .	30
5.4. Optimización con optUPMPARK del caso referencia con el campo discretizado. . . . .	34
5.5. Optimización con optUPMPARK del caso referencia con el campo continuo. . . . .	39

5.6. Optimización con optUPMPARK para una rosa de vientos con dos direcciones perpendiculares entre sí. . . . .	43
<b>6. Conclusiones e ideas de futuro.</b>	<b>45</b>
<b>Bibliografía</b>	<b>47</b>
<b>A. Anexo I: Tendencias actuales en la optimización de parques eólicos.</b>	<b>51</b>
<b>B. Anexo II: Funciones del Genetic Algorithm Toolbox (GAT) utilizadas. Ficheros empleados en el algoritmo genético.</b>	<b>55</b>
<b>C. Anexo III: Fichero implementación de la condición de distancia mínima entre aerogeneradores.</b>	<b>61</b>
<b>D. Anexo IV: Modelo teórico del estudio de Mosetti et al.</b>	<b>63</b>
<b>E. Anexo V: Fichero entrada UPMPARK.</b>	<b>67</b>

## Lista de Figuras

1.	Evolución de la capacidad de generación de energía en Europa. . . . .	4
2.	Evolución de la energía eólica instalada en Europa. . . . .	6
3.	Uso de los principales modelos de estelas en optimizaciones de parques eólicos. . . . .	8
4.	Tipos de modelos de rotor . . . . .	11
5.	Uso de los principales métodos de optimización de parques eólicos . . . . .	14
6.	Entrecruzamiento en algoritmos genéticos . . . . .	19
7.	Mutación en algoritmos genéticos . . . . .	19
8.	Flujograma de un algoritmo genético . . . . .	20
9.	Representación binaria de un algoritmo genético . . . . .	21
10.	Llamada al algoritmo genético en el script <i>main.m</i> . . . . .	21
11.	Generación 1 del ejemplo con restricción de distancia mínima . . . . .	26
12.	Generación 3 del ejemplo con restricción de distancia mínima. . . . .	26
13.	Configuración óptima del ejemplo con restricción de distancia mínima . . . . .	26
14.	Subrutina implementada para mostrar y guardar información del algoritmo genético	27
15.	Parámetros del algoritmo genético guardados con formato <i>.mat</i> . . . . .	28
16.	Configuración optimizada propuesta por Mosetti para el caso referencia. . . . .	29
17.	Campo de velocidad en $ms^{-1}$ de la configuración óptima propuesta por Mosetti para el caso referencia al evaluarlo con UPMPARK . . . . .	32
18.	Campo de la energía cinética turbulenta en $m^2s^{-2}$ para la configuración óptima propuesta por Mosetti para el caso referencia al evaluarlo con UPMPARK . . . . .	32
19.	Parte del código del <i>script</i> <i>main.m</i> implementando el caso referencia con un campo discreto en solución . . . . .	35
20.	Configuración óptima del caso referencia propuesta por optUPMPARK con parque discretizado . . . . .	36

21.	Campo de velocidad en $ms^{-1}$ para el óptimo resultado por optUPMPARK para el caso referencia con campo discretizado. . . . .	37
22.	Campo de la energía cinética turbulenta en $m^2s^{-2}$ para el óptimo resultado por optUPMPARK para el caso referencia con campo discretizado . . . . .	37
23.	Evolución del algoritmo genético para campo discretizado . . . . .	38
24.	Evolución del algoritmo para el caso de parque continuo . . . . .	39
25.	Configuración óptima del parque con continuidad en la solución . . . . .	40
26.	Campo de velocidad en $ms^{-1}$ para el óptimo resultado por optUPMPARK para el caso referencia con campo continuo. . . . .	41
27.	Campo de la energía cinética turbulenta en $m^2s^{-2}$ para el óptimo resultado por optUPMPARK para el caso referencia con campo continuo . . . . .	41
28.	Evolución del algoritmo para una rosa de vientos bidireccional y parque continuo del caso referencia calculado con optUPMPARK . . . . .	43
29.	Configuración óptima para una rosa de vientos bidireccional y parque continuo del caso referencia calculado con optUPMPARK . . . . .	44
30.	Modelado de estela FLORIS (control de la orientación del rotor . . . . .	53
31.	Opciones del algoritmo genético . . . . .	57
32.	Función <i>gaoutfnc.m</i> . . . . .	58
33.	Llamada al algoritmo genético en el script <i>main.m</i> . . . . .	58
34.	Subrutina <i>constrain_dist_min.m</i> . . . . .	61
35.	Modelo de estela de Jensen . . . . .	63
36.	Modelo de interacción de varias estelas de Jensen . . . . .	64
37.	Curva potencia aerogenerador . . . . .	66
38.	Fichero <i>mosetti.dat</i> para UPMPARK . . . . .	67

## Lista de Tablas

1.	Datos de potencia eólica instalada en el Estado español. Información tomada del Instituto para la Diversificación y Ahorro de la Energía [5]. . . . .	5
2.	Posiciones de la configuración óptima del ejemplo con restricción de distancia mínima. . . . .	26
3.	Posiciones óptimas propuestas por Mosetti et al. [17] para el caso referencia. . .	29
4.	Resultados de Mosetti para el caso referencia. [26]. . . . .	30
5.	Parámetros relevantes del UPMPARK. . . . .	31
6.	Muestra del flujo básico calculado por UPMPARK bajo las condiciones de la tabla 5 y de los parámetros recogidos del estudio de Mosetti. . . . .	33
7.	Datos de la configuración óptima propuesta por Mosetti calculados con UPMPARK.	33
8.	Potencia y eficiencia del parque optimizado para el caso referencia. . . . .	34
9.	Datos de la solución propuesta por optUPMPARK para el caso referencia con campo discretizado. . . . .	37
10.	Comparación entre óptimos propuestos por Mosetti y por optUPMPARK con campo discretizado para el caso referencia. . . . .	38
11.	Datos del óptimo, resultado de simular el caso referencia con optUPMPARK y campo continuo. . . . .	41
12.	Comparación entre la configuración óptima de optUPMPARK con continuidad en la solución y los casos de las secciones previas. . . . .	42
13.	Datos del óptimo resultado de simular el caso referencia con optUPMPARK y campo continuo y una rosa de vientos con dos direcciones. . . . .	44



# 1. Preliminares, resumen y palabras clave.

## Comentario Preliminar:

El presente trabajo es el producto de cursar el Trabajo de Fin de Grado (TFG), asignatura 42626/OB/12.0 y cuyo código es 42626. Para llevarse a cabo esta memoria se ha colaborado entre el Departamento de Energía de la Universidad Politécnica de Madrid (UPM) y el Departamento de Ingeniería Energética y Fluidomecánica de la Universidad de Valladolid (UVA). La fecha de inicio del estudio en este trabajo fue el 14/09/2016, con el inicio de las prácticas curriculares en empresa.

La motivación para realizar el TFG en este campo de estudio ha sido el trabajo realizado durante las prácticas en empresa curriculares y extracurriculares en el mencionado departamento de la UPM, en el cual una de sus líneas de investigación es la optimización de parques eólicos. Las herramientas aprendidas en los trabajos realizados son variadas, Matlab ha sido el lenguaje de programación en el que esta construida la optimización. Ha sido necesaria la lectura y comprensión de artículos de investigación en lengua castellana y anglosajona con el vocabulario técnico propio de esta rama (especialmente los que tratan estudios de estela y optimizaciones de parques eólicos). También se necesitan los conocimientos necesarios de Mecánica de Fluidos para saber interpretar los resultados mostrados en los ficheros de salida de los códigos utilizados. Por último añadir la utilización de herramientas de ofimática como  $\text{\LaTeX}$  y Microsoft Office Picture Manager para exponer el contenido de la mejor forma posible.

## Resumen y objetivo:

Se pretende abordar la optimización de la configuración de los aerogeneradores en parques eólicos *offshore* (en mar adentro). Para este fin se ha utilizado el código propio del departamento de la UPM denominado optUPMPARK, que incluye UPMPARK para evaluar del campo fluido y algoritmos genéticos para el proceso de optimización. El objetivo es evaluar el potencial de optUPMPARK como código de optimización de parques eólicos, señalando aspectos positivos y negativos y ayudando a su mejora. Es por tanto un trabajo preliminar del producto final.

En la memoria, primero se expone una breve contextualización de la energía eólica. A continuación la teoría de los códigos que sustentan el presente trabajo. Se prosigue con las simulaciones realizadas para el objetivo expuesto. Finalmente se han interpretado y extraído conclusiones de los resultados de las simulaciones y se citan algunas ideas y recomendaciones para el futuro de esta línea de investigación en el departamento de Ingeniería Energética de la UPM. Al final del documento se podrá encontrar la bibliografía consultada y los anexos necesarios para una mejor comprensión del trabajo, dejando solo las principales ideas en la parte central de la memoria.

**Palabras clave:** parque eólico *offshore*, algoritmo, estela, optimización, función objetivo.



## 2. Introducción: contexto actual de la energía eólica. Límites y futuro.

La energía que incide en la Tierra procedente del sol es de aproximadamente 174 PW (petavatios), siendo 89 PW la que llegue a océanos y continentes por las pérdidas de reflexión y absorción en la atmósfera. La energía eólica surge de la diferencias de temperatura entre las diversas partes del planeta como océanos y continentes o entre los polos y el ecuador. Estas diferencias de temperatura sumadas a los efectos rotativos de la Tierra originan las corrientes del fluido que se conoce como aire, una mezcla homogénea cuyas propiedades varían según la temperatura, composición y presión a la que se ve sometida. El viento es el aire en movimiento, y su energía cinética es la que se aprovecha en la energía eólica.

La energía eólica se considera hoy en día uno de los pilares fundamentales de las energías renovables en la búsqueda de alternativas a las energías provenientes de recursos fósiles (convencionalmente derivados del petróleo, carbón y gas natural). Existe en esta energía renovable un amplio campo de investigación, desde diversas formas de construcción y cimentación, pasando por el estudio de la distribución y configuración de los parques eólicos, la aerodinámica de las palas, aspectos económicos, y hasta los límites productivos a nivel global y local de cada parque. Betz calculó que el máximo porcentaje de energía que se puede aprovechar de la energía cinética del viento es de aproximadamente el 59,26 %. Los estudios de impacto ambiental en parques eólicos son necesarios debido a varios problemas derivados de los aerogeneradores de palas, como la mortalidad de aves en sus flujos migratorios o la contaminación acústica por el ruido generado, entre otros.

En el ámbito mundial, según Carlos de Castro et al. [1], existen varios límites que se deben tener en cuenta sobre la energía eólica: un límite por la disponibilidad del terreno, un límite de los materiales necesarios de construcción, y por el principio de conservación de la energía un límite máximo entorno a 1 TW mundial de potencia eléctrica suministrada (menor a la potencia instalada admisible), aunque este límite energético está muy enfrentado con trabajos que usan la metodología *bottom-up*. Según wikipedia [2], en 1900 la potencia demandada mundial media residía en torno a 0,7 TW, en 2015 fue de media anual 18,292 TW según Enerdata [3], siendo los mayores consumidores China y Estados Unidos. Se deben tener en cuenta también las pérdidas entre los puntos de generación de energía y los puntos de demanda de energía.

En la figura 1 se muestra la evolución que ha tenido Europa respecto a su capacidad de generación de energía. En ordenadas se representa la capacidad en GW, en abscisas los años. Se observa la evolución continua, casi lineal y positiva de la energía eólica, superando en diversos años a la derivada del petróleo, la nuclear y la hidráulica hasta colocarse en segunda posición, por detrás del gas natural, que encabeza la capacidad de producción de energía. Esta evolución de la energía eólica transmite gran interés en su futuro, al igual que se observa con la energía solar fotovoltaica, aunque esta última en menor medida.

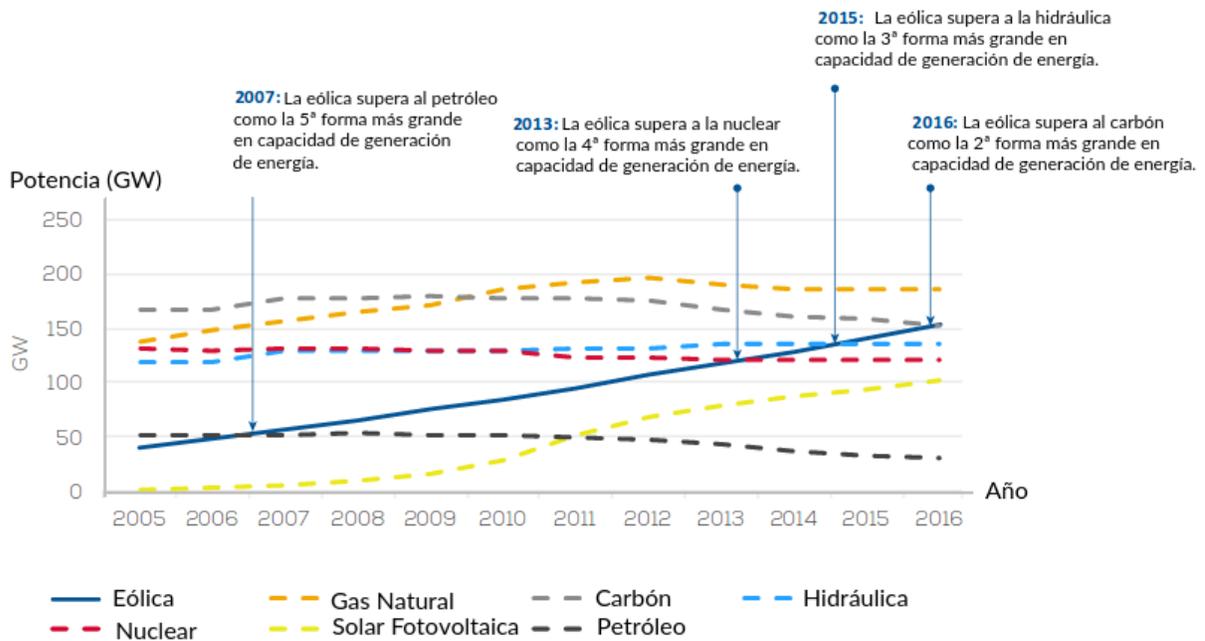


Figura 1. Evolución de la capacidad de generación de energía en Europa. [4].

Según el informe de WindEurope [4], en 2016 se instalaron y conectaron a red 12,5 GW de capacidad de generación de energía a partir de eólica, un 3% menos que el año anterior, aún así ha sido la forma de energía más instalada en Europa ese año. De la potencia eólica instalada en 2016, 10,923 MW fueron *onshore* y 1,567 MW *offshore*. Con una producción de casi 300 TWh en 2016, la energía eólica cubrió el 10.4% de la demanda eléctrica europea. Actualmente hay 153,7 GW de capacidad de energía eólica en la Unión Europea, de los cuales 141,1 GW son *onshore* y 12,6 GW *offshore*. Se debe tener en cuenta el desarrollo posterior de la energía eólica *offshore* respecto a la *onshore*. Alemania encabeza la lista de países con potencia eólica instalada, con el 44% del total de instalación de la Unión Europea, la siguen por orden España, Reino Unido y Francia.

En el Estado español en 2015 la potencia media demandada durante ese año fue de 154.000 megavatios (MW) [3]. De toda esta demanda, la energía eólica pudo aportar 22.981 MW, que son los instalados en ese año tal y como se muestra en la tabla 1, donde se refleja por años la evolución de la potencia eólica instalada anualmente y la acumulada de año en año en el Estado español, recogido del Instituto para la Diversificación y Ahorro de la Energía [5] (IDAE):

Tabla 1. Datos de potencia eólica instalada en el Estado español. Información tomada del Instituto para la Diversificación y Ahorro de la Energía [5].

Año	Potencia Instalada Anualmente (kW)	Potencia Eléctrica Acumulada (MW)
2000	821.859	2.281
2001	990.342	3.272
2002	1.610.156	4.882
2003	1.294.806	6.180
2004	2.000.685	8.181
2005	1.625.701	9.944
2006	1.741.543	11.693
2007	3.124.387	14.818
2008	1.828.335	16.646
2009	2.612.603	19.267
2010	1.413.773	20.675
2011	854.782	21.529
2012	1.259.517	22.789
2013	169.368	22.958
2014	16.266	22.975
2015	5.693	22.981

Se puede observar en la tabla 1 la disminución en los últimos años de la instalación de aerogeneradores, estancándose en los casi 23.000 MW de potencia acumulada instalada. Esto se debe a la legislación y a las políticas en energías renovables principalmente. De los datos de WindEurope se extrae que España ha dado un bajón en la instalación de energía eólica. En 2015 y 2016 solamente se instalaron 49 MW, mientras que Alemania instaló 11.451 MW nuevos en estos dos últimos años. La diferencia es muy grande para ser los dos primeros países de la lista europea en capacidad de producción de energía eólica.

Bajo estas limitaciones antes mencionadas y teniendo en cuenta la evolución de los recursos energéticos fósiles como el petróleo y el carbón conviene optimizar lo mejor posible los parques instalados, aprovechando de la forma más eficiente el recurso eólico disponible. El debate también existe entorno a un posible decrecimiento económico y energético por parte de las regiones más demandantes de energía. Los sistemas basados en el crecimiento permanente no son sostenibles en el tiempo en un entorno finito como es nuestro planeta, las ingenierías y las ciencias deben ser consecuentes con las limitaciones presentes y los efectos producidos del crecimiento descontrolado de las visiones economicistas.

Se distinguen tres tipos de parques eólicos: *offshore* (mar adentro), *nearshore* (cercano a tierra) y *onshore* (en tierra). El presente trabajo se enfoca en la optimización de parques eólicos *offshore* considerando dos aspectos principales: maximización de la potencia a nivel de parque y minimización de la turbulencia añadida resultado de la interacción entre las estelas de los

aerogeneradores. Con ello se pretende maximizar el beneficio por la venta de energía minimizando los costes derivados de la turbulencia, ya que se ha constatado la conexión existente entre el aumento de la turbulencia incidente en un aerogenerador y la disminución de la vida útil de este, con los costes de mantenimiento y reducción de la vida productiva del parque que ello conlleva.

El interés por los parques eólicos *offshore* reside en la presencia de un perfil más plano y mayor magnitud para la velocidad del viento en comparación con parques *onshore*, una orografía más sencilla, no tener en cuenta la topografía (que añadiría complejidad al problema) y disponibilidad de terreno ante la falta de poblaciones (teniendo en cuenta rutas marítimas e impactos ambientales). Las zonas más atractivas son regiones ventosas y con profundidades menores de 80 metros [6].

Los costes económicos totales de inversión inicial en parques *offshore* son notablemente mayores que en parques *onshore*, esta problemática ha quedado reflejada en el propio sector, planteando en 2014 una necesaria reducción de costes de entre el 35 % y el 40 % hasta 2020. El vicepresidente de la división de eólica *offshore* de Alstom Power, Anders Soe Jensen, asegura: “*si nosotros, como industria, no logramos reducir el costo nivelado de la electricidad, esta industria va a morir, es así de simple. A largo plazo esta sociedad no va a seguir financiando esta industria si no podemos ser competitivos respecto a otras fuentes de energía*” [7].

Para terminar esta sección resulta interesante la evolución indicada en la figura 2. Se representa en el eje horizontal los años, en el vertical la potencia de este tipo de energía. Señalar el mayor interés año tras año por la energía eólica *offshore* hasta 2015, instalándose en 2016 una menor cantidad de capacidad como se ha evaluado en la figura 1. El futuro de la energía eólica es creciente, pero no exponencial, la tendencia que seguirá esta energía la marcarán las condiciones globales en términos medioambientales, económicos y político-sociales.

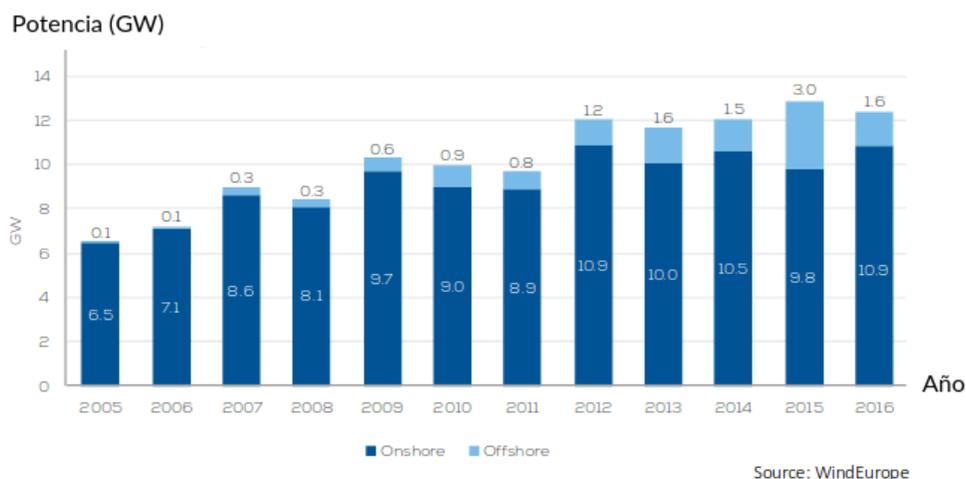


Figura 2. Evolución de la energía eólica instalada en Europa. [4].

### 3. Antecedentes del estado del conocimiento.

En esta sección se presenta la teoría necesaria para la comprensión del trabajo realizado en las optimizaciones y del contexto en el que se presentan.

En el desarrollo del proyecto de un parque eólico se debe distinguir la ponderación del recurso eólico como fase de investigación previa a la optimización de la configuración del parque. En esta última se evalúa el campo fluido y la interacción entre estelas, así como la forma de optimizar el parque. En este sentido se desarrollarán los modelos más empleados en la segunda fase, ya que la ponderación del recurso eólico pertenece a otra línea de investigación.

Los modelos más empleados en el cálculo de las magnitudes fluidas en los puntos de interés del parque eólico incluyen efectos producidos por el rotor, la capa límite atmosférica o el suelo, entre otros. Se profundiza en los modelos de campo por ser UPMPARK uno de ellos. Se mencionan las estrategias de optimización de la configuración de los aerogeneradores en parques eólicos. Hasta 1999 el artículo *"Survey of Modelling Methods for Wind Turbine Wakes and Wind Farms"* [8] de Crespo, Hernandez y Frandsen recoge adecuadamente el desarrollo del problema. Entre 1999 y la actualidad se han dado a conocer estudios de elevado interés, el artículo *"Review of computational fluid dynamics for wind turbine wake aerodynamics"* de Sanderse et al. [9] condensa esta etapa de forma clara y concisa.

#### 3.1. Modelización del campo fluido en un parque eólico.

Las primeras aproximaciones para modelar campos eólicos con elevado número de aerogeneradores consideraban a estos como elementos rugosos. De esta manera el parque eólico es representado como una superficie de rugosidad distribuida que modifica el flujo atmosférico ambiente. Estos modelos difieren en el perfil del flujo no perturbado (flujo básico) empleado. El perfil es normalmente logarítmico, con la rugosidad del suelo como parámetro. Las aeroturbinas incrementaban el valor del parámetro de rugosidad. Calculando los perfiles de velocidad de viento incidentes para cada turbina se puede hallar la potencia de cada una de ellas. Estos modelos pueden ser de interés para estimar efectos generales o características del viento en grandes parques, pero carecen de la exactitud necesaria en problemas de optimización de parques eólicos para el cálculo del campo fluido. El resto de este tipo de modelos se pueden dividir entre modelos cinemáticos y modelos de campo.

- **Modelos cinemáticos:** calculan por separado los efectos del suelo, capa límite atmosférica y estela, en base a la propiedad de autosemejanza de los perfiles de velocidad. La modelización del desarrollo de la estela es bastante precaria al no tener en cuenta su región de expansión, donde se producen los mayores esfuerzos cortantes, ni el desarrollo de la turbulencia añadida por esta. El desarrollo de la estela se implementa mediante perfiles

de velocidad para las tres regiones diferenciadas: cercana, transición y lejana, mayoritariamente desarrollados a partir de estudios de chorros. Los modelos cinemáticos conocidos son el modelo de Jensen [10] (posteriormente desarrollado por Katic et al. [11]), el modelo de Larsen [12] y el modelo de Frandsen [13]. El efecto del suelo se simula con técnicas de imagen bidimensionales que no aportan buenos resultados como se demostró en los estudios con modelos tridimensionales posteriores.

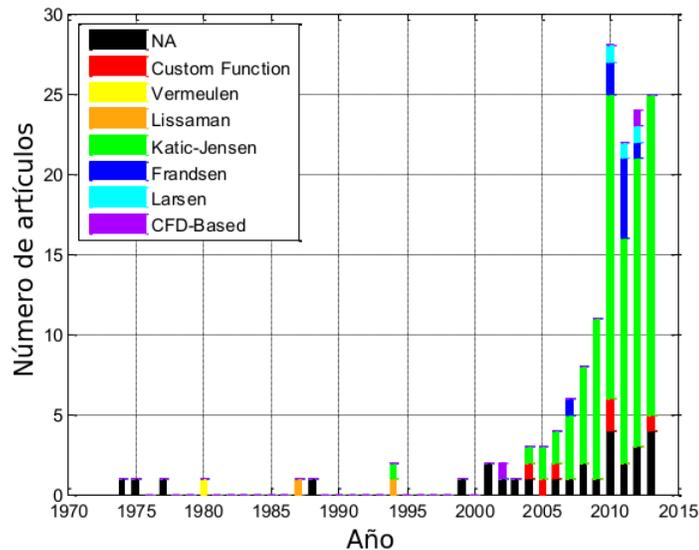


Figura 3. Uso de los principales modelos de estelas en optimizaciones de parques eólicos. [14].

En la figura 3 se representa la evolución en el uso de los modelos cinemáticos en optimizaciones de parques eólicos. En el eje de abscisas se señalan los años, en ordenadas el número de artículos de investigación. Se muestran por colores los distintos modelos de estela empleados en los artículos. Se nota claramente que el más usado es el de Katic–Jensen (barra verde) desde el año 2005. El incremento exponencial en el número de estudios hasta el 2010 y luego manteniéndose entorno a los 25 artículos por año dan cuenta de la evolución positiva e interés actual por este campo de investigación.

- Modelos de campo:** evalúan el campo fluido considerando a los aerogeneradores como elementos imbuidos en él. Simulan todos los efectos bajo un mismo modelo excepto la zona inmediatamente aguas arriba del rotor, una zona de influencia aguas abajo del rotor por la naturaleza elíptica del flujo en esta región. Se rigen por las ecuaciones que gobiernan el problema de mecánica de fluidos: las ecuaciones de Navier–Stokes. Los modelos de campo dan una aceptable representación del campo fluido y una mejor visión en los procesos de comportamiento del desarrollo de estela que los modelos cinemáticos. De los últimos textos a finales de siglo pasado se verifica la necesaria consideración tridimensional de los efectos de la estela, del suelo y de la capa atmosférica que perturban el flujo básico del campo

fluido, lo que hace más eficaces aún a los modelos de campo por las interacciones entre efectos que se reproducen en los modelos.

Como se acaba de exponer, en los modelos de campo se utilizan las ecuaciones de Navier–Stokes: conservación de la masa, cantidad de movimiento y energía. Se tienen 5 incógnitas en los problemas de mecánica de fluidos: 3 componentes de velocidad (ejes cartesianos), presión y temperatura, dependientes de la posición y el tiempo. En casos donde el régimen turbulento es protagonista (elevado número de Reynolds) la solución no es estable y se hace inviable computacionalmente encontrar una solución exacta. Se debe considerar además la naturaleza que se le dan a las distintas ecuaciones, que puede ser parabólica o elíptica en nuestro problema a la hora de evaluar el dominio de influencia y dependencia. En la parabólica la zona de dependencia son las secciones situadas aguas arriba del punto estudiado y las secciones de influencia las que están aguas abajo en la dirección predominante del flujo. En la elíptica se confunden, depende e influye en secciones aguas arriba y aguas abajo en la dirección principal del flujo.

Para la resolución numérica de las ecuaciones se encuentran tres principales estrategias en función de cómo se quiera implementar la evolución de la turbulencia. En el campo de la mecánica de fluidos se suele utilizar la Teoría de Cascada de Energía para interpretar las distintas escalas de turbulencia en función de la energía asociada a los torbellinos. Las escalas mayores (de mayor energía) están asociadas a torbellinos del tamaño característico del problema, que en nuestro caso será el tamaño del rotor. Los torbellinos se van dividiendo y haciéndose más pequeños a causa de los efectos cortantes y la interacción entre ellos, pasando por las distintas escalas intermedias hasta llegar a la más pequeña, la escala de Kolmogorov. A medida que crece el número de Reynolds la escala de Kolmogorov disminuye, necesitando un número exponencial de celdas más en la malla computacional para una buena precisión, llegando a ser inabarcable computacionalmente en algunos casos. Tres son los métodos empleados para resolver computacionalmente las ecuaciones de Navier–Stokes: Simulación Numérica Directa, *Large Eddy Simulation* ó *Reynolds-averaged Navier-Stokes*.

- **Direct Numerical Simulation (*DNS*)**: resuelve las ecuaciones de Navier–Stokes para todo el rango de escalas de la turbulencia. El coste computacional es inviable dados los recursos de hoy día. Como alternativas más empleadas a este tipo de simulaciones se presentan las estrategias LES y RANS.
- **Large Eddy Simulation (*LES*)**: calculan la macroescala e interpreta que los torbellinos de la escala de Kolmogorov tienen carácter más o menos universal y no dependen de la geometría del flujo. Matemáticamente, esta escala de separación se lleva a cabo filtrando espacialmente mediante una integral de convolución el campo de velocidad, dividiéndolo para resolver la velocidad en una parte resoluble (la macroescala) y otra parte sin resolver (escala de Kolmogorov). Han recibido mayor atención que los otros dos métodos en la comunidad que investiga sobre estelas, por su capacidad para manejar inestabilidades transitorias, flujos turbulentos anisotrópicos dominados por estructuras de gran escala y mezcla turbulenta, además de la disponibilidad suficiente de recursos computacionales.

- **Reynolds-averaged Navier-Stokes (RANS):** promedian estadísticamente las fluctuaciones en las ecuaciones de Navier-Stockes, resultando las ecuaciones de Reynolds. Necesitan de una ecuación elíptica implícita (ecuación de Poisson) para el cálculo de la presión. Computacionalmente son los métodos más ligeros.

Los métodos LES tienen ventaja sobre la mayoría de los métodos RANS en que predicen mejor los efectos transitorios o la anisotropía de la turbulencia presentes en la capa límite atmosférica. Réthore [15] realizó una comparación entre un método RANS (standard *con modelo de cierre turbulento  $k - \epsilon$* ) y otro LES (método de Bechmann), siendo claramente superiores los resultados con LES, pero el tiempo computacional de RANS fue de horas respecto al de LES que fue de días. Para una estela aislada la diferencia de tiempo computacional es de un factor de 60 para las mismas condiciones.

Los métodos RANS necesitan un cierre de las ecuaciones a través de modelos de turbulencia debido a la complejidad del tensor de Reynolds de esfuerzos viscosos. Para hallar la solución se utiliza un sistema de 7 ecuaciones: 1 de la conservación de la masa, las 3 de cantidad de movimiento, 1 para la conservación de la energía, más 2 del modelo de cierre para resolver el tensor de Reynolds de esfuerzos viscosos en las ecuaciones de Navier-Stockes. Las incógnitas ahora son 7: 3 de la velocidad, la presión, la temperatura, y dos a mayores de la interpretación del tensor de Reynolds. Los modelos de cierre más empleados se apoyan en la intensidad turbulenta ( $k$ ) y en la tasa de disipación turbulenta ( $\epsilon$ ) que se explicarán más adelante.

Como se ha mencionado, en la implementación del modelo de cierre de las ecuaciones de Reynolds se utilizan principalmente modelos  $k - \epsilon$  semiempíricos mediante la ecuación de la energía cinética turbulenta y la ecuación de la tasa de disipación de la energía cinética turbulenta. Un tipo de modelos alternativos a los  $k - \epsilon$  estándar lineales (EVM) son los de viscosidad turbulenta no lineales (NLEVM), que mejoran el uso tradicional de la hipótesis de Boussinesq (se tienen en cuenta diferencias de densidad que, debidas a variaciones de temperatura, afectan al término gravitatorio en las ecuaciones de cantidad de movimiento). Los NLEVM están basados en una relación no lineal tensión-deformación en la cual los productos de los gradientes de velocidad están presentes. A mayores, los NLEVM suelen tener un flujo dependiente de  $C_\mu$ , una compleja función escalar dependiente del flujo local, añadiendo una nueva ecuación de la tasa de disipación de la energía cinética turbulenta. En términos de déficit de estela y esfuerzos de Reynolds, el desempeño de los NLEVM es mejorado en comparación con los  $k - \epsilon$  EVM. Como conclusiones del estudio de Cabezón et al. [16] el modelo turbulento  $k - \epsilon$  estándar subestima los déficits de velocidad, modelos turbulentos de mayor orden como los *Anisotropic Reynolds Stress Model* (RSM) ofrecen mejores resultados para los déficits de velocidad en la región de estela lejana. Si se asume la hipótesis de Boussinesq los resultados son buenos, especialmente entre distancias aguas abajo entre 5.5 y 8 veces el diámetro de rotor.

Tanto en RANS como en LES la modelización de las palas del rotor es necesaria ya que se producen importantes efectos de compresibilidad y esfuerzos cortantes. Así como en los alrededores inmediatos del rotor. Se utiliza el diámetro expandido, un poco mayor que el diámetro del rotor

físico. Los modelos de rotor tratan a estos como sumideros o fuentes de velocidad y turbulencia respectivamente. Se pueden clasificar en:

- Teoría de elemento de pala.
- Disco actuador (AD).
- Línea actuadora (AL).
- Superficie actuadora (AS).
- Modelos directos o complejos.

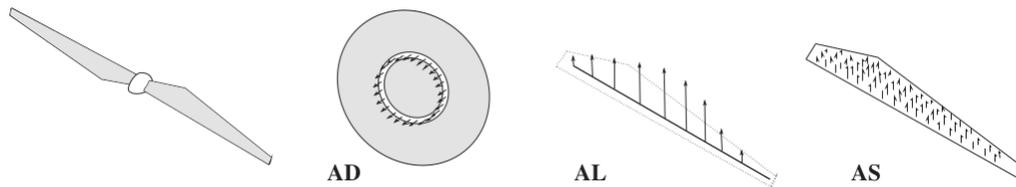


Figura 4. Tipos de modelos de rotor. [9].

En la figura 4 se observan los principales modelos de representación para el rotor: disco actuador (AD), línea actuadora (AL) y superficie actuadora (AS). Es sabido que los modelos de disco actuador y elemento de pala son simples. Los modelos de línea actuadora y de superficie actuadora arrojan mejores resultados. Ganar precisión en los datos aerodinámicos viene acompañado de un necesario mayor coste computacional. La naturaleza no estacionaria de los métodos de línea y de superficie actuadora son más adecuados para simulaciones LES y la naturaleza estacionaria de los métodos disco actuador quedan limitados a su aplicación en simulaciones RANS. Sin embargo, los costes computacionales limitan el uso de la técnica de línea actuadora a investigaciones de estela aislada y la mayoría de simulaciones LES de parques eólicos están siendo modeladas como discos actuadores. Por otra parte, inviables para las optimizaciones de parques eólicos por su coste computacional, son los métodos directos o completos. Estos modelos construyen una malla gradual adaptada al cuerpo, más fina cuanto más nos acerquemos a este. Físicamente, los métodos directos son los más precisos y sólidos. Los efectos de compresibilidad en la punta de las palas requieren abordar las ecuaciones de Navier–Stokes acopladas y con naturaleza elíptica, mientras que la estela permanece incompresible y se puede evaluar con las ecuaciones desacopladas y naturaleza parabólica, sin apenas perder precisión.

Para finalizar esta sección se anotan algunas conclusiones de los estudios sobre estelas aisladas, que antecedieron a la modelización de parques eólicos con interacciones entre varias estelas. Se destacan varios aspectos, resumidos en los párrafos siguientes.

En modelos cinemáticos, para hallar los déficits de velocidad se opta por la conservación de la cantidad de movimiento, con la ecuación de conservación de la masa implícita. El efecto que causa la propia turbina parada es despreciable frente al resto de efectos. No se puede considerar uniformidad para el flujo ambiente, ni linealidad del mismo. Los perfiles usados son normalmente logarítmicos, ya que el suelo realiza un efecto cuanto más cerca de él se encuentre el fluido. El efecto del suelo solo puede tratarse con modelos tridimensionales. La suposición de axilsimetría del flujo necesaria en los modelos cinemáticos supone descartar estos modelos en muchos casos, ya que lleva a errores grandes.

En modelos de campo no pueden justificarse modelos bidimensionales ni linealizaciones para el flujo básico debido al efecto del suelo, además del rol dominante de las estelas en las regiones de estudio del parque eólico. El tamaño de los torbellinos de la macroescala turbulenta es mucho mayor que el de la estela, por lo tanto las fuerzas de Coriolis en la estela pueden despreciarse al compararse con otros efectos presentes en la misma. Si además se asume naturaleza parabólica en la ecuación de cantidad de movimiento, en sus componentes se pueden tener en cuenta solamente variaciones de presión en la dirección transversal al flujo principal del viento. En algunos estudios la superposición lineal de efectos dio lugar a que con varios aerogeneradores en fila alineados en la dirección principal del flujo se dieran potencias negativas, careciendo de sentido. Por todo ello se concluye la necesidad de modelos tridimensionales para la estela y para el perfil incidente en el rotor.

De los estudios sobre UPMWAKE (código propio desarrollado por el Departamento de Energía de la UPM para calcular el desarrollo de una estela aislada) se conoce la posibilidad de asumir, para regiones donde se pueda despreciar el efecto de estela, que toda la convección es debida al flujo ambiente sin perturbar. Esto implica poder reducir el sistema de ecuaciones diferenciales en derivadas parciales de siete a tres manteniendo el carácter tridimensional del problema.

Un aspecto insuficientemente tratado en la literatura es la difusión de la estela en atmósferas no neutras. Existen así mismo discrepancias físicas y experimentales sobre el comienzo de la región de expansión de la estela, esta comienza aguas abajo pero la distancia varía por la capa de cortadura de la estela, que comienza inmediatamente detrás del rotor aguas abajo.

Los estudios de parques eólicos se basan en los de estela aislada. Se añade la dificultad de la interacción entre estelas. Los parques eólicos se pueden clasificar en: *onshore* (en tierra), *nearshore* (cercano al mar) y *offshore* (mar adentro). El efecto de la topografía normalmente se superpone a la solución del campo eólico, evaluándolo previamente con códigos específicos. La superposición lineal aporta malos resultados, siendo necesario construir modelos más complejos. En parques *offshore*, al soplar el viento de tierra a mar, se genera una capa límite interna semejante al efecto de la topografía, siendo necesaria su consideración. Pese a ser mucho más sencilla y menor la orografía del terreno en mar respecto a tierra esta también se evalúa.

### 3.2. Antecedentes del estudio de optimización de parques eólicos.

Los métodos anteriormente descritos permiten marcar el centro de interés actual en la optimización de parques eólicos (WFDO, siglas en inglés de Wind Farm Design and Optimization) con objeto de conocer varias variables de interés en el diseño de un parque eólico. Los aspectos más importantes en los estudios se relatan a continuación:

- **Función objetivo:** es la función matemática evaluada para optimizar los parámetros de interés. El resultado de dicha evaluación debe ser un valor de referencia para el algoritmo (*fitness value*). La Producción Anual de Energía (AEP), la conversión instantánea de potencia y el Coste de Energía (CoE) son las principales funciones de referencia en los procesos de optimización.
- **Variables de diseño y restricciones:** el problema WFDO aspira a definir el mejor valor de la función objetivo siguiendo un conjunto de variables tales como la propia localización, el tipo, el tamaño y el número de aerogeneradores, las alturas de buje de las máquinas, el tipo de torre, la infraestructura auxiliar y algunas más. Todas las variables están limitadas por factores físicos como características climatológicas, terreno, tecnología disponible, etc.
- **Métodos de resolución. Algoritmos:** tres grupos diferentes: (1) modelos de programación no lineal de enteros mixtos (MINLP) y métodos basados en cálculos; (2) Métodos heurísticos; (3) Métodos metaheurísticos.

Los métodos de solución más extendidos son poblaciones metaheurísticas basadas en algoritmos genéticos (GA, de *Genetic Algorithm*), no obstante es común encontrarse combinaciones de métodos. En cuanto a objetivos se refiere se pueden encontrar estrategias de un solo objetivo o de varios a la vez. La principal ventaja de estrategias multi-objetivo es que no necesitan formular submodelos complejos que traduzcan valores de variables físicas a valores medibles de comportamiento, sino que pueden considerarse como objetivos del problema.

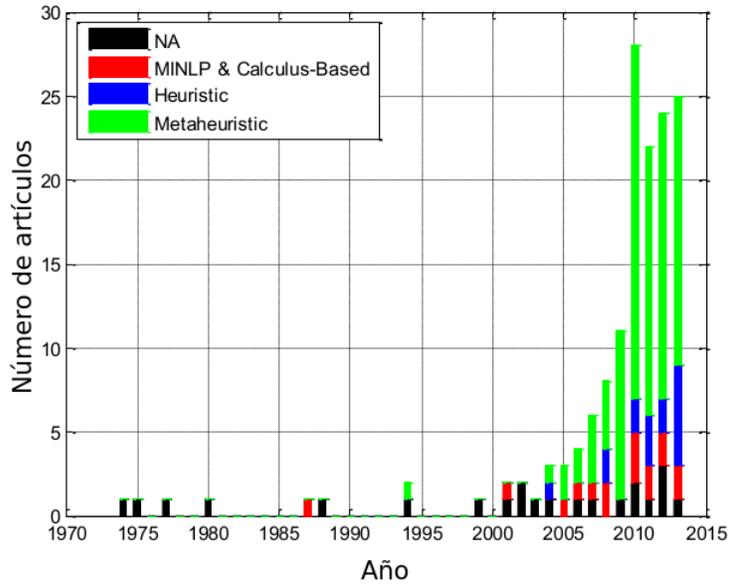


Figura 5. Uso de los principales métodos de optimización de parques eólicos. [14].

En la figura 5 se puede observar la evolución de los métodos empleados en la optimización de parques eólicos. En abscisas se apuntan los años representados, en ordenadas el número de artículos de investigación. Los colores representan los distintos métodos de optimización empleados en los artículos. Se nota el mayor peso de los métodos metaheurísticos (barra verde) a partir del año 2006, dentro de los cuales se encuentran los algoritmos genéticos. Como ocurría en la figura 3, se nota la subida exponencial de este tipo de artículos hasta 2010, fecha donde se estanca sobre los 25 artículos por año.

Algunos ejemplos recientes mencionados en el Anexo A definen bien los últimos avances realizados en el campo de investigación tratado, entre los cuales se encuentra el de Mosetti et al. [17], importante en el presente TFG y que se menciona más adelante con frecuencia, ya que sobre este pesan varias comparaciones, tanto de otros estudios como del presente trabajo. Se relatan diversas estrategias para definir dinámicamente los algoritmos genéticos, diferentes evaluaciones de la función objetivo, así como novedosos puntos de vista como la posibilidad de optimizar la orientación del rotor o la colocación de distintos tipos de aerogeneradores en el mismo campo de varias alturas, añadiendo jugabilidad.

## 4. Marco del presente trabajo

Se explican las hipótesis empleadas en la construcción del código propio UPMPARK, construido en Fortran y que calcula el campo fluido de un parque eólico. A continuación se expone la base teórica y práctica del código propio denominado optUPMPARK, construido en Matlab y que optimiza la configuración de los aerogeneradores para un parque eólico, usando internamente UPMPARK.

### 4.1. Códigos propios UPMPARK, UNION 2.2 y GRAFICA.

El presente Trabajo de Fin de Grado (TFG) se enmarca en la línea de investigación en energía eólica del Departamento de Energía de la Universidad Politécnica de Madrid, cuya más reciente aportación es el Trabajo de Fin de Máster de Francisco Javier Jiménez Murcia [6]. Del tutorial "*Programa UPMPARK para el Modelado de Parques Eólicos*" [18] y la tesis doctoral de Rafael Gómez-Elvira González [19] se extraen las principales características del código propio utilizado para el cálculo del campo fluido, expuestas a continuación.

Los efectos que se tienen en cuenta son los creados por la estela, la rugosidad del terreno y la estabilidad atmosférica. El efecto de la topografía en el caso de parques *onshore* o efecto tierra-mar en *offshore* se ha de estudiar por separado y superponerse a los otros efectos del parque calculados por UPMPARK. Este puede representar el efecto de dispersión atmosférica conocido como *meandering*, que en las simulaciones de este trabajo se ha evitado para obtener un menor coste computacional. Si se tiene en cuenta se obtendrá mayor precisión en el cálculo del campo fluido a costes computacionales mayores.

UPMPARK resuelve las ecuaciones con RANS y un modelo de cierre turbulento  $k - \epsilon$  basado en el propuesto por Launder y Spalding [20]. El cálculo del campo fluido de UPMPARK se basa en el código propio con lenguaje Fortran para una estela aislada denominado UPMWAKE, la cual puede resolver las ecuaciones del problema mecánico y térmico de forma acoplada (las 7 ecuaciones) o desacoplada (3 ecuaciones, al asumir que toda la convección es debida al flujo ambiente sin perturbar). Por lo tanto en el desarrollo de estela aislada UPMWAKE y UPMPARK coinciden, siendo UPMPARK el código que evalúa además la interacción entre estelas. UPMPARK asume las siguientes hipótesis:

- Se desprecian los transportes moleculares frente a los turbulentos.
- Aproximación de capa límite. Las longitudes características en la dirección del flujo básico son mucho mayores que las transversales. Por esta razón, la difusión de las magnitudes fluidas en sentido longitudinal se desprecian frente a la difusión transversal.
- Acepta la hipótesis de Boussinesq: solo se consideran las variaciones de densidad debidas

a una diferencia de temperaturas en el término gravitatorio de las fuerzas másicas de la ecuación de conservación de la cantidad de movimiento, en su componente vertical.

- Adopta el concepto de viscosidad turbulenta isotrópica del método  $k - \epsilon$  basado en la hipótesis de *eddy viscosity*. Esto establece que la transferencia de cantidad de movimiento causada por la turbulencia puede ser modelada con una viscosidad turbulenta, análoga a la viscosidad molecular.
- Se escoge la dirección del flujo incidente como 'x' y su perpendicular en el plano horizontal al suelo como eje 'y'. El eje 'z' es vertical.

Respecto al flujo básico, se supone terreno plano y llano (admite superposición de una topografía no muy compleja, calculada previamente con programas como *Wind Atlas Analysis and Application Program* (WAsP)) y que la dirección del viento no cambia en zonas con extensión del orden de la longitud de estela. Por estas suposiciones se considera un flujo básico unidimensional y logarítmico con la altura, fundado por las propiedades del aire al nivel del suelo, la rugosidad del terreno, la velocidad del viento, la fricción turbulenta y la estabilidad atmosférica (dada por el parámetro longitud de Monin-Obukhov). El flujo básico se describe más detalladamente en Panofski y Dutton [21].

UPMPARK implementa la teoría de disco actuador para simular el rotor de las máquinas, asumiendo las siguientes hipótesis:

- Ausencia de pérdidas por fricción.
- La resultante de fuerzas de presión en la superficie de deslizamiento es despreciable frente al flujo de cantidad de movimiento a través del disco.
- La presión estática en puntos suficientemente alejados del rotor es igual a la del flujo no perturbado.
- La distribución de la fuerza de empuje sobre el disco del rotor es uniforme.

Como se verá en la sección de simulaciones realizadas, el parámetro  $TDWA$  define la longitud en diámetros del rotor para la cual deja de calcular las magnitudes fluidas con la teoría de disco actuador (región flujo compresible) y empieza a calcular el desarrollo de la estela con flujo incompresible.

Respecto a la resolución de las ecuaciones de Reynolds se asume naturaleza parabólica. Se discretizan las ecuaciones con un método en diferencias finitas y se resuelven con el método *Alternating Direction Implicit* (ADI).

Añadir el posible uso del código propio UNION 2.2 [22] que pondera todos los casos de una rosa de vientos completa. Incluye la opción de calcular el porcentaje de disminución en la vida útil

previsible, calculada en base al principio de resistencia de materiales que afirma que el producto de la intensidad de una carga por su frecuencia es un valor constante, de tal forma que si una carga aumenta su magnitud, la frecuencia que aguantará el material antes de romper por fatiga disminuirá proporcionalmente a dicho aumento. UNION 2.2 no se utilizará en el presente trabajo por carecer de interés en nuestros casos no reales. Se utiliza UPMPARK también para evaluar los casos estudiados con las ecuaciones desacopladas, las 3 necesarias, flujo parabólico, para las posiciones salidas del proceso de optimización. La opción de calcular los óptimos con las 7 ecuaciones no se utilizará ya que de estudios previos se sabe que posee problemas de convergencia y los resultados en ambos casos son muy parecidos, siendo la forma simplificada más rápida en términos computacionales.

Una vez expuesta esta teoría conviene aclarar la diferencia en el uso de los códigos UPMPARK y WASP (desarrollado por *The Wind Energy Division* del Risø DTU, Dinamarca), así como la relación entre ellos. WASP puede llegar a calcular todos los parámetros de interés de un parque, como la energía media anual. No obstante, su uso se limita hoy en día al análisis del recurso eólico, evitando el modelo de estela e interacción entre estelas (PARK, modelo cinemático), demasiado simple y poco preciso. UPMPARK no calcula el recurso eólico, solo el desarrollo de las estelas para el viento incidente, así como su interacción.

Con los datos diezminutales de la velocidad del viento (magnitud y dirección) aportados por la estación meteorológica del parque, WASP representa las funciones estadísticas de la distribución de las magnitudes fluidas en el parque (distribución de Weibull para el viento), originando lo que WasP denomina *Observed Wind Climate* (OWC). Esto junto a la topografía y la orografía (debida a la rugosidad del terreno y a los obstáculos), WASP genera el *Wind Atlas Data Sets* (WADS) en el punto del mástil meteorológico. Este viento es extrapolado, ahora inversamente, a los puntos de interés, en nuestro caso al dominio del parque eólico en estudio, suponiendo que los puntos del dominio a estudiar no están a gran distancia del mástil meteorológico. El viento local en el parque se denomina en WASP *Estimated Wind Climate*.

En la primera línea del fichero de entrada principal de UPMPARK se puede reflejar que se le aporta un fichero con la orografía. Este fichero añadido es el resultado del estudio con WASP. De esta forma a partir de la orografía, la topografía y los datos diezminutales del mástil meteorológico se calculan con WASP las magnitudes fluidas del viento en el parque sin aerogeneradores que lo perturben, para después añadirlos y que UPMPARK calcule las magnitudes fluidas en el dominio estudiado en función de las estelas generadas y la interacción entre ellas. Siendo UPMPARK quien calcule las potencias, eficiencias y otros parámetros de interés.

El código propio GRAFICA [23] se utiliza para mostrar algunos resultados gráficamente, aporta con buena resolución y para una dirección de viento y altura determinadas dos gráficas, correspondientes a las magnitudes de energía cinética turbulenta y velocidad del viento. Cada código necesita de determinados ficheros de entrada. UNION 2.2 y UPMPARK generan ficheros de salida, que no se entran a explicar en este trabajo, quedan plasmados adecuadamente en los textos [18], [22] y [23].

El proceso de optimización se ha realizado mediante algoritmos genéticos, que se explicarán a continuación. La función objetivo es `executeUPMPARK.m`, dentro de la cual se ejecuta UPM-PARK. Como objetivo se maximiza la potencia del parque. En la representación del parque eólico se calcula además la turbulencia añadida por los aerogeneradores y la suma de las distancias entre aerogeneradores, pero estos no son unos parámetros en la optimización, solo lo es la potencia.

## 4.2. Código propio `optUPMPARK` para la optimización de parques eólicos.

En este apartado se describe el programa de optimización de un parque eólico utilizado en el presente trabajo, se le ha denominado `optUPMPARK`. Primero, se expone la base teórica de los algoritmos genéticos por ser estos los implementados en el programa. Después, se explican los *scripts* construidos con Matlab para implementar el algoritmo genético.

### 4.2.1. Algoritmos genéticos como método de optimización de parques eólicos.

La naturaleza de los algoritmos genéticos parte de la idea biológica de la selección natural postulada por Darwin [24] como sustento de la optimización. *”Los algoritmos genéticos consisten en un método iterativo y estocástico para resolver problemas de optimización, tanto con restricciones como sin ellas. Estos están basados en la selección natural, el proceso por el que se lleva a cabo la evolución biológica. Por semejanza con el proceso biológico, se denomina individuo a cada una de las posibles soluciones, y población al conjunto de individuos de una generación, siendo las generaciones cada uno de los procesos iterativos del proceso. En cada generación, el algoritmo genético selecciona de forma aleatoria ciertos individuos que tendrán la función de ser los padres que darán lugar a los descendientes de la siguiente generación. En cada generación sucesiva, la población evoluciona hacia una solución óptima”* [6]. Señalar que el enfoque biológico es subjetivo; se da la existencia de estudios que arrojan información de relaciones biológicas más eficientes y realistas que la selección natural, como la cooperación, el apoyo mutuo e incluso la ayuda altruista. De estos otros procesos se podrían sacar ideas interesantes de cara a implementar algoritmos, pero no se entrarán en ellos en el presente trabajo.

La figura 6 visualiza el entrecruzamiento. A partir de dos padres (individuo a cuadros e individuo ondulado) se generan dos individuos contrapuestos entre sí intercambiando genes. La figura 7

refleja el fenómeno de la mutación. En este caso, a partir de un solo individuo, se coge un gen y se le cambia por otro aleatoriamente, generando un nuevo individuo con la mayor parte de genes que el anterior pero diferente en comportamiento.



Figura 6. Entrecruzamiento; padres (izquierda) e hijos (derecha). [6].



Figura 7. Mutación; progenitor (izquierda) y descendiente mutado (derecha). [6].

A continuación, la figura 8 muestra un flujograma que resume el funcionamiento básico de un algoritmo genético. Se observa un inicio y un final. Primero se crea la población de partida, se evalúa y a continuación empieza el bucle de toma de decisiones con el criterio de parada que se imponga. En este bucle se controla y modifica a la población con las tres herramientas definidas anteriormente: selección, entrecruzamiento y mutación. Se repite el bucle hasta llegar al criterio de parada, para entonces finalizar el algoritmo.

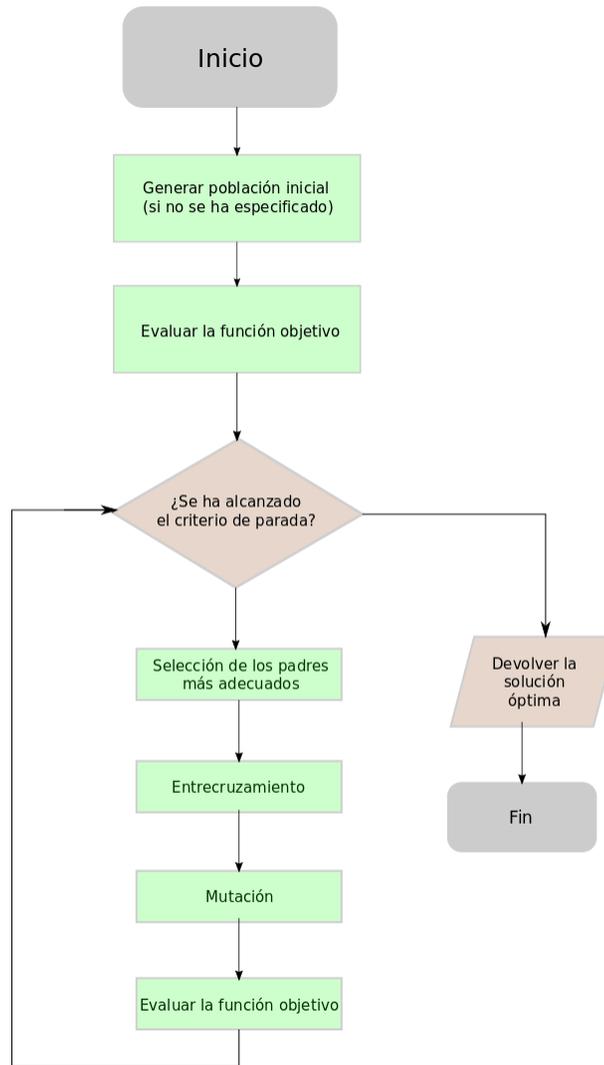


Figura 8. Flujograma de un algoritmo genético. [6].

Los cromosomas (genotipo) contienen la información del individuo. Un individuo puede ser creado a partir de esa información. El fenotipo guarda el rendimiento de un individuo (y por tanto de su genotipo ligado a él). Los individuos más aptos (con mejor fenotipo) son seleccionados para reproducirse. Los procesos evolutivos más comunes son la mutación y el cruce o recombinación, combinando los individuos seleccionados y añadiendo variabilidad con la mutación. De esta forma una posible solución puede ser representada con una serie de parámetros, cada parámetro es un gen en el equivalente biológico, el conjunto de parámetros (genes) queda codificada en una cadena de valores ó cromosoma. La figura 9 representa esta definición, en ella se muestra un ejemplo de un individuo binario que codifica 3 parámetros.

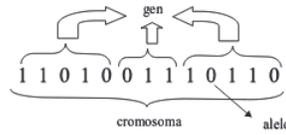


Figura 9. Representación binaria de un algoritmo genético. [25].

En el Anexo B se explican con más detenimiento las distintas funciones escogidas del *GA Toolbox*.

#### 4.2.2. *Scripts* en Matlab y ficheros utilizados en el algoritmo genético.

Se señalan ahora las principales funciones escogidas para implementar el algoritmo genético. En el anexo B se expone de forma más detallada la herramienta de Matlab utilizada. En la figura 10 se observa parte de código del *script main.m*. Concretamente se muestra la línea de llamada al algoritmo con los argumentos y variables de salida, que se explican a continuación.

```
% @constrain_dist_min; Función de restricciones de la distancia mínima.
% Calculates optimum
[x,fval, exitflag] = ga(ObjectiveFunction,nvars,[],[],[],[],LB,UB, @constrain_dist_min,[], options);
```

Figura 10. Llamada al algoritmo genético en el script *main.m*.

- $x \rightarrow$  es la solución final del algoritmo. En nuestro caso es el parque óptimo.
- $fval \rightarrow$  es el *fitness value* del individuo óptimo final.
- $exitflag \rightarrow$  nos dice cómo ha acabado el algoritmo. Si todo ha ido correctamente este valor será un 1.
- *ObjectiveFunction*  $\rightarrow$  *executeUPMPARK.m*. Es la función objetivo del algoritmo. Dentro de esta se ejecuta UPMPARK.
- $nvars \rightarrow$  número de variables de la función objetivo, en nuestro caso el número de coordenadas ( $2 * nTurbines$ ).
- $primer[ ] \rightarrow$  matriz con las restricciones lineales del problema en desigualdades  $\rightarrow A * X \leq B$ .

- *segunda*[ ] → matriz B de constantes para las desigualdades lineales.
- *tercera*[ ] → matriz con las restricciones no lineales del problema en desigualdades → Aeq\*X ≤ beq.
- *cuarta*[ ] → matriz beq de constantes para las desigualdades no lineales.
- *LB* → conjunto de restricciones de contorno inferiores.
- *UB* → conjunto de restricciones de contorno superiores.
- *@constrain\_dist\_min* → NONLCON, función que contiene los vectores de las restricciones no lineales (igualdades y desigualdades) del problema, en nuestro caso las distancias mínimas entre aerogeneradores.
- *quinto*[ ] → vector para indicar las variables que deben ser números enteros. Se utiliza para discretizar el parque en posiciones concretas para la solución. Se emplea también en una de las simulaciones que se muestran más adelante en la presente memoria.
- *options* → se introducen aquí las opciones definidas anteriormente en el algoritmo. Las opciones se han variado en las simulaciones que se muestran más adelante para adecuarlas a las mismas, por ejemplo, para un parque discretizado no hacen falta las funciones de mutación, selección, etc.

Los ficheros creados para el algoritmo y la función objetivo se listan bajo este párrafo. Estos ficheros son el fruto de un largo trabajo teórico y práctico dentro del Departamento de Energía de la Escuela Técnica Superior de Ingenierías Industriales de la UPM. Así mismo reconocer la labor de personas que a través de Internet muestran al mundo sus conocimientos.

- *\*.tur* → contiene las propiedades de los aerogeneradores. Tantos ficheros como tipos se definan en el parque. En nuestro ejemplo solo hay un tipo, un modelo y una altura fijas, el *Vestas60.tur*.
- *windfarmX.dat* → se definen las características y parámetros del parque como entradas de UPMPARK. Este fichero sirve para generar todos los ficheros *wfX.dat* que se irán creando y eliminando automáticamente para simular todos los casos necesarios de la rosa de vientos y de cada configuración del parque. En nuestro ejemplo se denomina *windfarm26.dat* (se detalla en la sección 3.2.2 Tutorial UPMPARK [18]).
- *upmpark.exe* → el código propio UPMPARK.
- *main.m* → es el fichero (*script*) principal.
- *getParameters.m* → en esta función se introducen las dimensiones, el número de aerogeneradores, la distancia mínima entre aerogeneradores y la rosa de vientos del parque.

- `createDAT.m`→ es la función que crea los ***wfX.dat*** necesarios a partir del fichero ***windfarmX.dat***.
- `deleteFiles.m`→ elimina los ficheros después de simular un caso para poder generar otros de nuevo con el mismo nombre y que no se acumulen. Los valores de interés se almacenan durante la simulación.
- `executeUPMPARK.m`→ llama a UPMPARK tantas veces como sea necesario para evaluar toda la rosa de vientos, establecida en el fichero *getWindrose.m*, para cada individuo de la población de cada generación. Solo almacena la potencia de cada individuo para calcular el *fitness function* del mismo. La función `getPower.m` es la encargada de calcular dicha potencia a partir de un fichero creado por UPMPARK (.pow).
- `executeUPMPARK2.m`→ esta función se llama sólo para representar individuos. Es similar a la anterior, hace lo mismo solo que calcula además la turbulencia añadida por los aerogeneradores y la suma de las distancias entre estos. La función encargada de abrir los ficheros creados por UPMPARK necesarios es `getPowerK.m`.
- `getPower.m`→ toma los valores de la potencia de un individuo.
- `getPowerK.m`→ misma función que la anterior solo que además toma valores de las intensidades de turbulencia y la suma de las distancias entre aerogeneradores para tener más información del mejor individuo.
- `getDATparam.m`→ genera todos los parámetros necesarios para crear el fichero ***wfX.dat*** que corresponda.
- `getTextPosition.m`→ simplemente introduce los valores de las posiciones de un individuo en el fichero ***wfX.dat***.
- `getPosition.m`→ evalúa si se introducen las posiciones de los individuos de forma aleatoria o un individuo de forma manual.
- `getIndividual.m`→ genera de forma aleatoria las posiciones de un elemento.
- `getPosition2.m`→ introduce las posiciones del primer individuo de forma manual, el resto de forma aleatoria.
- `RosaVientosOpt.m`→ función que genera las probabilidades de cada suceso de la rosa de vientos en función de la probabilidad de que ocurra un viento en un sector y las distribuciones de Weibull en dichos sectores (parámetros A y k). La matriz resultante será la que se introduzca manualmente en la función `getWindrose.m`. Esta función se realizó durante las prácticas en empresa en la UPM a partir de un foro de internet del cual no se dispone referencia, por lo tanto no se usará en el presente TFG.
- `getWindrose.m`→ Aquí se introduce la rosa de vientos a evaluar.



## 5. Simulaciones Realizadas

El esquema de esta sección es el siguiente: primero se presentan las dos mejoras introducidas en el código `optUPMPARK`, una implementa la subrutina que restringe la distancia mínima entre aerogeneradores y la otra permite almacenar y mostrar información del algoritmo genético de interés. Después, se expone la presentación del caso referencia (caso A de Mosetti et al. [17]) y su cálculo con `UPMPARK`. Posteriormente, se representan y analizan las optimizaciones del caso referencia realizadas con el programa `optUPMPARK` para compararlas respecto a la optimización de Mosetti del caso referencia. Las optimizaciones con `optUPMPARK` se han realizado para un campo discreto en la solución como para un campo con continuidad en la solución. Por último, se ha realizado una simulación con una rosa de vientos con 2 direcciones (Norte y Oeste) bajo las mismas condiciones que las dos anteriores.

### 5.1. Mejoras del código `optUPMPARK`.

Se explican a continuación las mejoras realizadas en el programa de optimización de parques eólicos `optUPMPARK`. La subrutina que restringe la distancia mínima entre aerogeneradores y la subrutina que permite almacenar y mostrar los principales parámetros del algoritmo genético empleado.

#### 5.1.1. Restricción de la distancia mínima entre aerogeneradores.

Como argumento de entrada de la función `ga` se pueden introducir las restricciones no lineales de la distancia mínima entre aerogeneradores, implementando la ecuación 1:

$$S(i, j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (1)$$

La subrutina de la restricción se muestra en el Anexo C. Para verificar el buen comportamiento de la mejora se simuló un caso sencillo de 6 aerogeneradores en un campo de  $600 \times 600$  metros y una dirección del viento de oeste. La simulación duró 2 horas, acabando antes de las 6 generaciones impuestas debido al alcance de la tolerancia prevista para la función objetivo. Se muestran en las figuras 11–13 las generaciones 1, 3 y la óptima (la quinta). Tanto el eje de abscisas como el de ordenadas representa la distancia en metros del terreno respecto a la coordenada (0,0). El cuadrado remarcado representa los límites del parque eólico y los puntos azules a los aerogeneradores, como si de una vista aérea se tratase. Las flechas indican la dirección del viento. En la parte superior de las figuras se anotan la potencia (*pow*), la tasa de energía cinética turbulenta añadida por los aerogeneradores (*k*) y la suma de la distancia entre aerogeneradores (*distance*). Se observa el respeto de la distancia mínima impuesta como restricción, en este caso

200 metros. Las posiciones en el plano de la configuración óptima quedan recogidas en la tabla 2,  $x$  para abscisas e  $y$  para ordenadas, referidas a la coordenada  $[0,0]$  de la figura 13.

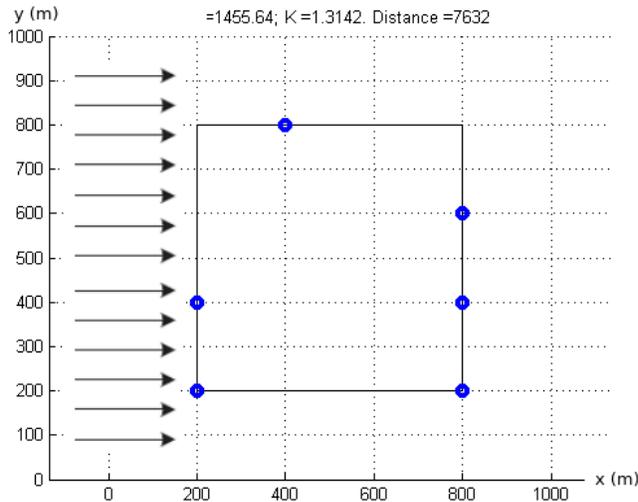


Figura 11. Generación 1 del ejemplo con restricción de distancia mínima.

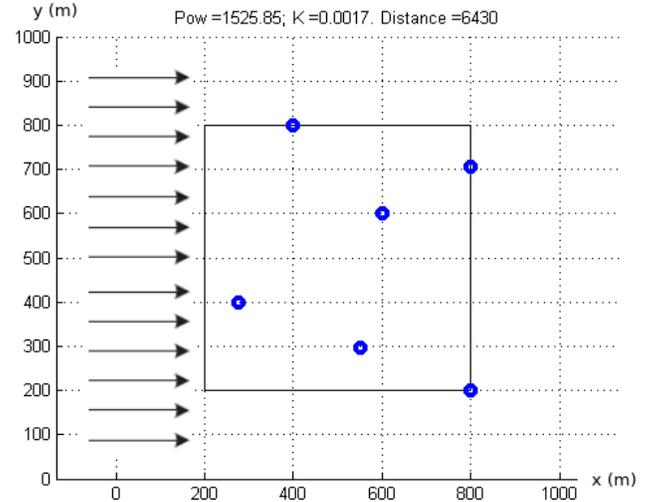


Figura 12. Generación 3 del ejemplo con restricción de distancia mínima.

Tabla 2. Posiciones de la configuración óptima del ejemplo con restricción de distancia mínima.

Aerogenerador	x (m)	y (m)
1	800	200
2	550.3166	296.3442
3	600	600
4	276.7896	400
5	525.6676	800
6	800	705.2177

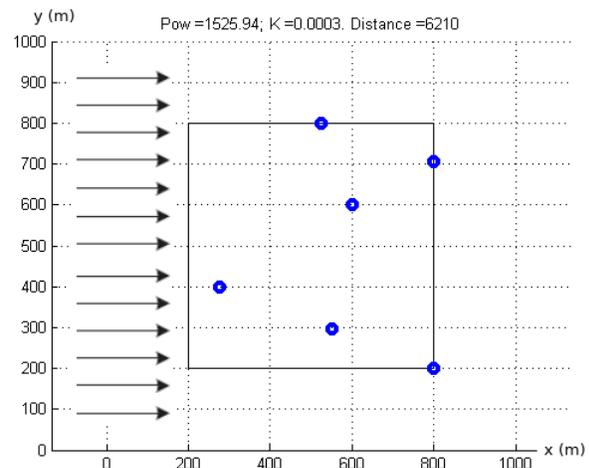


Figura 13. Configuración óptima del ejemplo con restricción de distancia mínima.

### 5.1.2. Almacenamiento de información del algoritmo genético.

Durante las simulaciones surgió la necesidad de almacenar los parámetros más importantes del algoritmo genético para poder evaluarlo al finalizar. Se encontró en un foro especializado información suficiente para implementar el almacenamiento de forma satisfactoria en una subrutina. El foro donde reside la información usada es el siguiente: "How to output the value of each iteration in Matlab for Genetic Algorithm and Simulated Annealing" en la web de "researchgate.net". La subrutina queda expuesta en la figura 14. El algoritmo genético de Matlab tiene tres estados: *init*, *iter* y *done*. Las operaciones se introducen mientras el algoritmo está iterando (caso *iter*). Los parámetros de interés se guardan en la variable interna del algoritmo denominada *state*, de tipo estructura *.mat*.

```
function [state, options,optchanged] = gaoutfnc(options,state,flag)

% Tomada de https://www.researchgate.net/post/How_to_output_the_value_of_each_iteration_in_Matlab_for_Genetic
% Copyright 2004-2006 The MathWorks, Inc.
% $Revision: 1.1.6.5 $ $Date: 2007/08/03 21:23:22 $

optchanged = false;

switch flag
    case 'init'
        disp('Starting the algorithm');
    case 'iter'
        disp('Iterating ...')
        if state.Generation==0
            else
                fname=[pwd,'\ ',num2str(state.Generation),'.mat'];
                mejores_individuos=find(state.Score == state.Best(state.Generation));
                state.CantidadOptimos=length(mejores_individuos);
                save(fname,'state')
                vector=state.Population(mejores_individuos(1),:);

                showResults(vector); % Muestra en cada generación la
                % mejor distribución de aerogeneradores.
                mejores_individuos=[];
            end
        case 'done'
            disp('Performing final task');
            fname=[pwd,'\ ',num2str(state.Generation),'.mat'];
            save(fname,'state')
end
```

Figura 14. Subrutina implementada para mostrar y almacenar información del algoritmo genético.

Esta subrutina sirve para analizar con mayor profundidad errores que puedan darse, guardando la información más importante de cada generación. En concreto representa la configuración óptima de cada generación llamando a la función propia `showResults.m`. Se guarda en la variable `state` de cada generación (por ejemplo para la tercera generación `3.mat`) los parámetros mostrados en la figura 15:

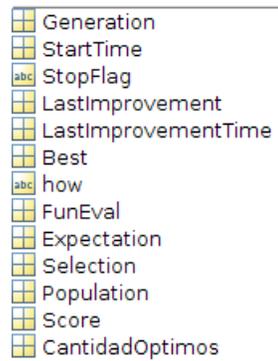


Figura 15. Parámetros del algoritmo genético guardados con formato `.mat`

Con estas dos mejoras y las funciones predefinidas en la función `ga` de Matlab la evolución del algoritmo genético queda reflejada convenientemente para su posible posterior análisis.

## 5.2. Descripción del caso referencia.

Como referencia de las simulaciones se ha escogido el estudio de Mosetti et al. [17], por su sencillez y abundancia de literatura sobre el mismo. En el Anexo D se explican los modelos de estela e interacción entre estelas utilizados por Mosetti, la curva de potencia de sus aerogeneradores (un solo tipo) y la función objetivo utilizada. En su estudio se realizaron varios casos, de los cuales se ha escogido la solución del primero (el caso A), con una rosa de vientos de una sola

dirección y magnitud del viento, 12 m/s del norte . No restringiendo el número de aerogeneradores, al optimizar resultó un parque con 26 máquinas. La configuración óptima de Mosetti se recoge en la tabla 3, correspondiente a las posiciones de la figura 16. Los aerogeneradores solo pueden colocarse en medio de las cuadrículas, cada cuadrícula tiene unas dimensiones de 200 metros en cada eje. Se tiene por tanto un parque discretizado de 100 posibles posiciones separadas 200 metros en direcciones  $x$  e  $y$ . Si Mosetti hubiese realizado esta simulación con el objetivo de maximizar la potencia saldría un parque con 100 aerogeneradores (uno para cada posible ubicación), al introducir el coste la ponderación varía. A partir de cierto número, para el cual interaccionan las estelas en aerogeneradores aguas abajo inevitablemente, la eficiencia y potencia totales del parque disminuye y aumenta respectivamente al aumentar el número de aerogeneradores.

Tabla 3. Posiciones óptimas propuestas por Mosetti et al. [17] para el caso referencia.

Aerogenerador	x	y
1	200	2000
2	600	2000
3	1600	2000
4	2000	2000
5	400	1800
6	1000	1800
7	1800	1800
8	800	1600
9	1200	1600
10	200	1200
11	1400	1200
12	1600	1000
13	800	800
14	1800	800
15	2000	800
16	1200	600
17	400	400
18	1000	400
19	1400	400
20	1600	400
21	200	200
22	600	200
23	800	200
24	1200	200
25	1800	200
26	2000	200

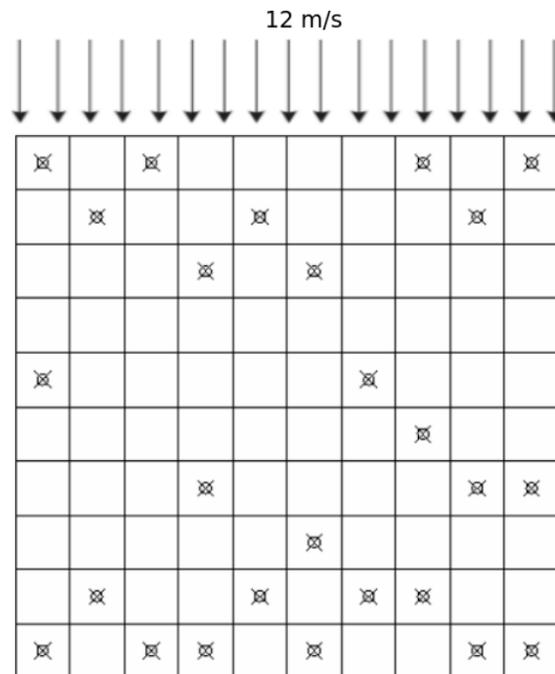


Figura 16. Configuración optimizada propuesta por Mosetti et al. [17] para el caso referencia.

Se observa en la solución dos franjas principales de localización de aerogeneradores, una en la parte más aguas arriba del parque y otra lo más aguas abajo posible en la dirección del flujo. Son características las configuraciones triangulares que se forman. Esto se debe al intento de reducir la interacción entre estelas en los aerogeneradores aguas abajo de los primeros.

Los resultados de las características globales del parque se describen adecuadamente en el trabajo de Bryony L. Du Pont et al. [26], y se muestran en la tabla 4. Estas características son el *fitness value* que señala el rendimiento del óptimo respecto a la función objetivo, la eficiencia y potencia total del parque para evaluar el rendimiento de producción de energía y el número de turbinas presente en el parque. Las funciones objetivo utilizadas por Mosetti y en optUPMPARK son diferentes. En Mosetti se maximiza la potencia y minimizan los costes del parque eólico (ecuación 7 del Anexo D), mientras que en optUPMPARK se maximiza únicamente la potencia.

Tabla 4. Resultados de Mosetti para el caso referencia. [26].

Fitness value	0.0016197
Eficiencia	91.64
Potencia Total (MW)	12.352
Número turbinas	26

Esta configuración óptima es aparentemente buena, ya que la eficiencia alcanzada es de casi el 92 %. Sin embargo, como se verá en la siguiente sección, el modelo empleado para resolver el campo fluido afecta en gran medida a la configuración óptima.

### 5.3. Evaluación del caso referencia con UPMPARK.

A continuación se ha calculado la solución óptima propuesta por Mosetti para el caso referencia con el modelo de campo UPMPARK para observar posibles diferencias entre modelos. Existen parámetros que se desconocen del caso referencia como la dimensión de la malla en dirección predominante del viento, errores admisibles ó el parámetro de relajación para la presión, presentes en el archivo de entrada al UPMPARK (descrito en el anexo E). Los parámetros más relevantes que se utilizarán con UPMPARK en la evaluación del campo fluido para esta y el resto de simulaciones se recogen en la tabla 5.

Tabla 5. Parámetros relevantes del UPMPARK.

Nombre	Valor	Descripción
PL	0	— 0 = 3 ecuaciones (problemas térmico y mecánico desacoplados). 1 = 7 ecuaciones (problema acoplado).
DX	10	— Tamaño de las celdas de la malla en metros, en dirección del flujo dominante.
L	$10^{23}$	— Longitud Monin-Obukhov (para ese valor se considera atmósfera neutra).
TDWA	2	— Longitud en diámetros del rotor para el cual acaba la teoría de disco actuador y empieza el desarrollo de la estela. Para un parque en estudio el valor 2 es el recomendado si no existe experimentación previa.

Se recuerda que la utilización de las ecuaciones en forma desacoplada (PL=0) se debe a la reducción del coste computacional sacrificando levemente la precisión. En cuanto al tamaño de la malla se ha valorado dar el valor de 10 metros por experiencias previas y el diámetro del rotor de los aerogeneradores. Para simular atmósfera neutra a través de la Longitud de Monin–Obukhov se le ha dado un valor muy alto para suponer longitud infinita.

Con estas características se han representado los resultados de forma gráfica. En la figura 17 se puede observar el campo de velocidades de la configuración óptima propuesta por Mosetti para el caso referencia. En la figura 18 se representa la distribución de la energía cinética turbulenta. Se observan claramente las estelas creadas por el viento al interactuar con los aerogeneradores, generando déficits de velocidad y aumentando la energía cinética turbulenta. Ambos resultados se obtienen respecto al flujo básico incidente en el parque, en este caso proveniente del norte. Los límites del parque no se han dibujado porque GRAFICA representa el cálculo con mayores dimensiones, pero los aerogeneradores están dentro de los límites del parque estudiado. Se observa el empleo del método de disco actuador para el rotor de cada aerogenerador en las zonas donde no se han representado las distribuciones de velocidad y de energía cinética turbulenta. La longitud inmediatamente después del aerogenerador en dirección del flujo aguas abajo, para la cual se emplea el método de disco actuador, se define en el parámetro TDWA escrita en la tabla 5.

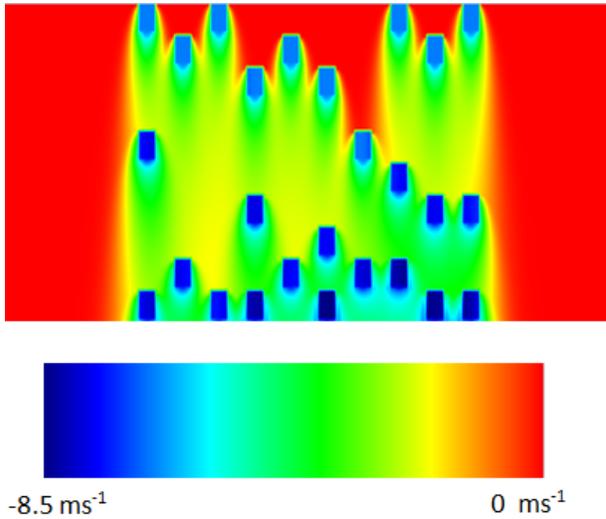


Figura 17. *Campo de velocidad en  $ms^{-1}$  de la configuración óptima propuesta por Mosetti para el caso referencia al evaluarlo con UPMPARK.*

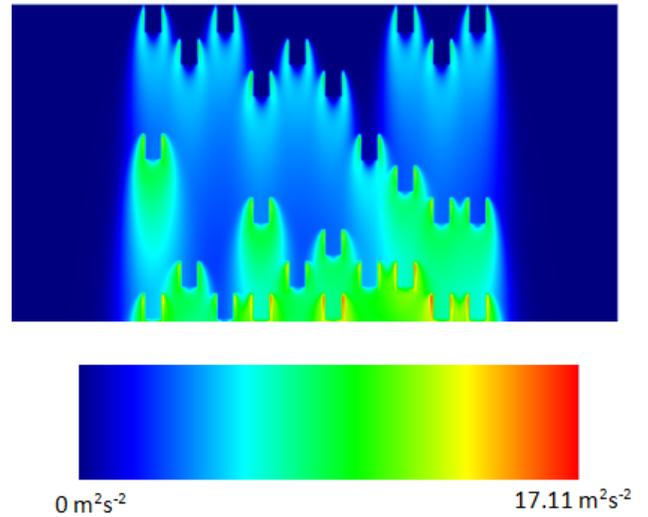


Figura 18. *Campo de la energía cinética turbulenta en  $m^2s^{-2}$  para la configuración óptima propuesta por Mosetti para el caso referencia al evaluarlo con UPMPARK.*

La representación corresponde al código propio GRAFICA para una altura respecto al suelo de  $z = 60,03$  m (altura de buje). Una parte del flujo básico incidente calculado por UPMPARK se muestra en la tabla 6. Se añade aquí solamente hasta una altura de 100 m (espesor aproximado para la zona superficial de la capa límite atmosférica), pero UPMPARK en este caso ha calculado dicho flujo básico hasta 453.33 m. Para cada altura,  $U$  es la velocidad del flujo básico,  $K$  es la energía cinética turbulenta y  $E$  la tasa de disipación de la energía cinética turbulenta ( $\epsilon$  en la teoría de modelos de cierre turbulentos descrita anteriormente). Se puede notar el efecto del suelo, formando si se representara un perfil vertical logarítmico para la velocidad.

Tabla 6. Muestra del flujo básico calculado por UPMPARK bajo las condiciones de la tabla 5 y de los parámetros recogidos del estudio de Mosetti.

Altura (m)	U ( $ms^{-1}$ )	K ( $m^2s^{-2}$ )	E ( $m^2s^{-3}$ )
100.0	13.157	4.518	0.019
93.33	13.001	4.518	0.020
86.67	12.833	4.518	0.021
80.00	12.652	4.518	0.023
73.33	12.454	4.518	0.025
66.67	12.239	4.518	0.028
60.00	12.000	4.518	0.031
53.33	11.733	4.518	0.035
46.67	11.431	4.518	0.040
40.00	11.082	4.518	0.046
33.33	10.669	4.518	0.056
26.670	10.163	4.518	0.070
20.00	9.512	4.518	0.093
13.33	8.593	4.518	0.139
6.67	7.024	4.518	0.279
0.30	0.000	4.518	6.196

Los datos numéricos de las potencias y eficiencias para cada aerogenerador calculadas por UPMPARK para el caso referencia se muestran en la tabla 7.

Tabla 7. Datos de la configuración óptima propuesta por Mosetti calculados con UPMPARK.

Aerogenerador	Potencia (kW)	Eficiencia
1	513.46	1.00
2	513.46	1.00
3	513.46	1.00
4	513.46	1.00
5	511.27	1.00
6	513.46	1.00
7	511.27	1.00
8	503.42	0.98
9	512.37	1.00
10	289.53	0.56
11	476.74	0.93
12	303.89	0.59
13	263.23	0.51
14	276.79	0.54
15	320.01	0.62
16	285.44	0.56
17	279.09	0.54
18	284.47	0.55
19	240.26	0.47
20	182.55	0.36
21	253.79	0.49
22	289.65	0.56
23	191.26	0.37
24	150.09	0.29
25	161.83	0.32
26	199.232	0.39

En la tabla 8 se comparan los resultados en potencia y eficiencia del caso referencia con el modelo cinemático de Jensen N.O (estudio Mosetti) y el de campo UPMPARK.

Tabla 8. Potencia y eficiencia del parque optimizado para el caso referencia.

Modelo cinemático de Jensen		Modelo de campo UPMPARK	
Potencia (kW)	Eficiencia (%)	Potencia (kW)	Eficiencia (%)
12352	91.64	9054	68.0

Como conclusión de esta comparación se destaca la marcada diferencia en eficiencia y potencia total del parque calculadas por UPMPARK en comparación con las calculadas por el modelo de Jensen utilizado por Mosetti. UPMPARK es un modelo más complejo y con mayor número de efectos a tener en cuenta que el modelo cinemático de Jensen tal como se expuso en la sección 3.1 y por la descripción en la sección 4.1 de las hipótesis empleadas en la construcción de UPMPARK, por todo ello aporta resultados más honestos. Se ve reducido en más de un 23 % la eficiencia y en más de 3000 kW la potencia calculadas por Mosetti. Dada la diferencia en los resultados es interesante realizar varias optimizaciones, con UPMPARK como herramienta de cálculo del campo fluido, para luego comparar los resultados de las optimizaciones bajo distintos modelos de cálculo del campo fluido. Se realizan en las siguientes secciones dos optimizaciones con el código `optUPMPARK`, una con un parque discretizado donde los aerogeneradores sólo pueden colocarse en posiciones concretas (números enteros en la solución), y otra para un parque continuo, es decir, que puedan tomar cualquier valor en cuanto a posición se refiere, dentro de los límites del parque.

#### 5.4. Optimización con `optUPMPARK` del caso referencia con el campo discretizado.

Para esta optimización se realizaron unos cambios en el código. Para un campo discretizado las posiciones vienen dadas por números enteros, simplificando el problema. En `ga toolbox` de Matlab no es necesario especificar las funciones de mutación, cruce, hibridación y selección. Tampoco se hace necesaria la función de distancia mínima porque está implícita al poder colocar los aerogeneradores en posiciones concretas ya separadas entre sí dicha distancia. La información sobre la implementación de este tipo de problemas se denomina `Mixed Integer Optimization`, se puede encontrar en el siguiente enlace:

<https://es.mathworks.com/help/gads/mixed-integer-optimization.html>

La llamada a la función se muestra en la figura 19. Se observa la ausencia de definición de las funciones de selección, mutación, hibridación y cruce en las opciones del algoritmo genético (options).

```
28 % Set options
29 - options = gaoptimset('PopulationSize', popSize, 'InitialPopulation', initPop,...
30   'Generations', generations, 'TolCon', 1e-8, 'TolFun', 1e-8, 'OutputFcns', @gaoutfnc,...
31   'PlotInterval', 1, 'PlotFcns', {@gaplotbestf, @gaplotstopping});
32
33
34 %gaoutput es la función que genera una matriz estructura en cada iteración.
35 %Para plotear los fitness values--> http://es.mathworks.com/help/gads/genetic-algorithm-options.html#f14474
36
37 % 'PlotFcns', {@gaplotdistance}
38 % {@gaplotdistance, @gaplotpareto, @gaplotscorediversity}
39
40 %Para discretizar el medio solo permitimos x= numeros enteros --> IntCon
41 - INTCON=1:nvars;
42
43 % Calculates optimum
44 - [x,fval, exitflag] = ga(ObjectiveFunction,nvars,[],[],[],[],LB,UB, [], INTCON, options);
45 - simulationTime = toc;
```

Figura 19. Parte de código del script *main.m* donde se observa la implementación de la variable *INTCON* para realizar una optimización con un campo discretizado en solución.

Para implementar en el código que las posiciones son números enteros, se realizó la restricción en la llamada a la función *ga* con el vector *INTCON* como se señala en el apartado *Characteristics of the Integer ga Solver* del siguiente enlace:

<https://es.mathworks.com/help/gads/mixed-integer-optimization.html#bs1cifg>

La figura 20 representa la configuración óptima del parque discretizado resultado del proceso de optimización con Matlab. Donde el perímetro del parque se limita por líneas rojas. Los puntos azules representan a los aerogeneradores en las posiciones dados los ejes *x* e *y* en metros. Las flechas en la parte superior indican la dirección del viento del norte, con magnitud de  $12 \text{ ms}^{-1}$ .

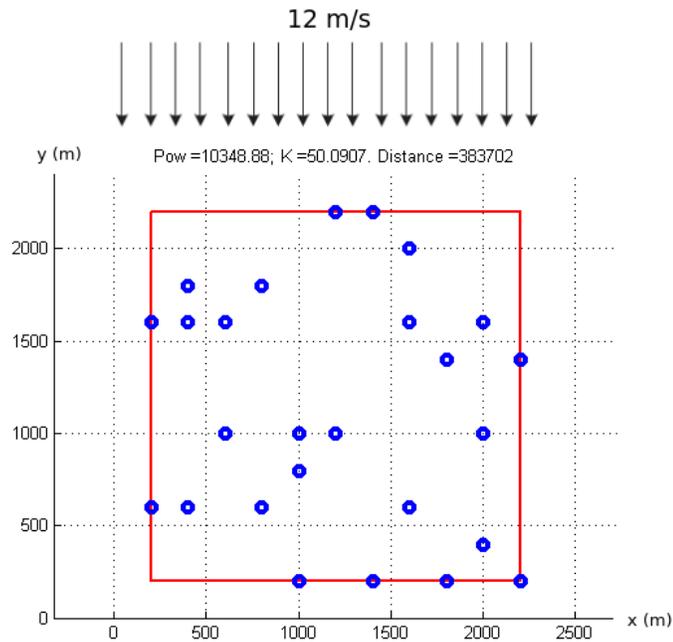


Figura 20. Configuración óptima del caso referencia propuesta por *optUPMPARK* con parque discretizado.

En la figura 21 se puede observar el campo de velocidades de la configuración óptima salida de *optUPMPARK* para el caso referencia con el campo discretizado. En la figura 22 se representa la distribución de la energía cinética turbulenta. Las estelas interactúan más intensamente en los aerogeneradores cercanos entre sí en la dirección predominante del viento, generando déficits de velocidad y aumentos en la energía cinética turbulenta importantes. Los resultados se obtienen respecto al flujo básico incidente en el parque, en este caso proveniente del norte. Los límites del parque no se han dibujado porque *GRAFICA* representa el cálculo con mayores dimensiones, pero los aerogeneradores están dentro de los límites del parque estudiado. Se observa el empleo del método de disco actuador para el rotor de cada aerogenerador en las zonas donde no se han representado las distribuciones de velocidad y de energía cinética turbulenta. La longitud inmediatamente después del aerogenerador en dirección del flujo aguas abajo, para la cual se utiliza el método de disco actuador, se define en el parámetro *TDWA* escrita en la tabla 5.

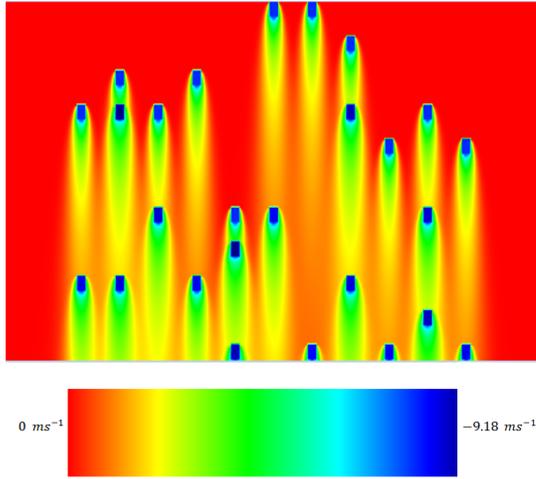


Figura 21. *Campo de velocidad en  $ms^{-1}$  para el óptimo resultado por optUPMPARK para el caso referencia con campo discretizado.*

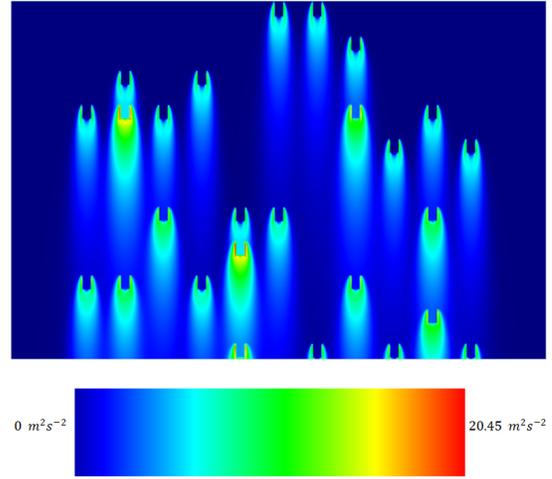


Figura 22. *Campo de la energía cinética turbulenta en  $m^2s^{-2}$  para el óptimo resultado por optUPMPARK para el caso referencia con campo discretizado.*

A continuación se añade la tabla 9 con las potencias y eficiencias de los aerogeneradores en el parque, así como sus posiciones, para la optimización en cuestión.

Tabla 9. Datos de la solución propuesta por optUPMPARK para el caso referencia con campo discretizado.

Aerogenerador	X (m)	Y (m)	Potencia (kW)	Eficiencia
1	800	1800	513.46	1.00
2	800	600	378.33	0.74
3	600	1000	314.94	0.61
4	1400	2200	513.46	1.00
5	1200	1000	377.45	0.74
6	2000	1000	316.28	0.62
7	400	1600	187.41	0.36
8	1800	200	366.72	0.71
9	200	1600	513.46	1.00
10	1600	2000	513.46	1.00
11	1600	1600	275.44	0.54
12	400	600	316.52	0.62
13	200	600	360.03	0.70
14	2000	1600	513.46	1.00
15	1600	600	313.08	0.61
16	1000	1000	509.45	0.99
17	2200	1400	513.46	1.00
18	1800	1400	513.26	1.00
19	1400	200	404.88	0.79
20	1000	200	263.64	0.51
21	1000	800	185.54	0.36
22	400	1800	513.46	1.00
23	1200	2200	513.46	1.00
24	2000	400	269.24	0.52
25	600	1600	513.46	1.00
26	2200	200	375.53	0.73

Tabla 10. Comparación entre óptimos propuestos por Mosetti y por optUPMPARK con campo discretizado para el caso referencia.

Óptimo de Mosetti para el caso referencia evaluado con UPMPARK		Óptimo con optUPMPARK para el caso referencia y campo discretizado	
Potencia (kW)	Eficiencia (%)	Potencia (kW)	Eficiencia (%)
9054	68	10349	78

En la Tabla 10 se observa la notable mejoría del parque en estudio, no obstante da la impresión observando las figuras 21 y 22 que la configuración no es la mejor posible ya que existen varios aerogeneradores muy afectados (los de menor eficiencia y potencia) pudiéndoles colocar en mejores situaciones. En la figura 23 se muestra la evolución del algoritmo para la optimización con campo discretizado. En el eje de abscisas se representan las generaciones y en el de ordenadas los valores de referencia de los *fitness values*, en nuestro caso, las potencias. Los puntos rojos son la media de los *fitness values* de los individuos de una generación y los puntos azules representan la configuración óptima de cada generación a través del mejor *fitness value* de la población. Se observa que en la primera generación se marca un óptimo que se perpetúa durante las siguientes generaciones debido al estancamiento de la media sobre los 6500 kW de potencia, el algoritmo no converge.

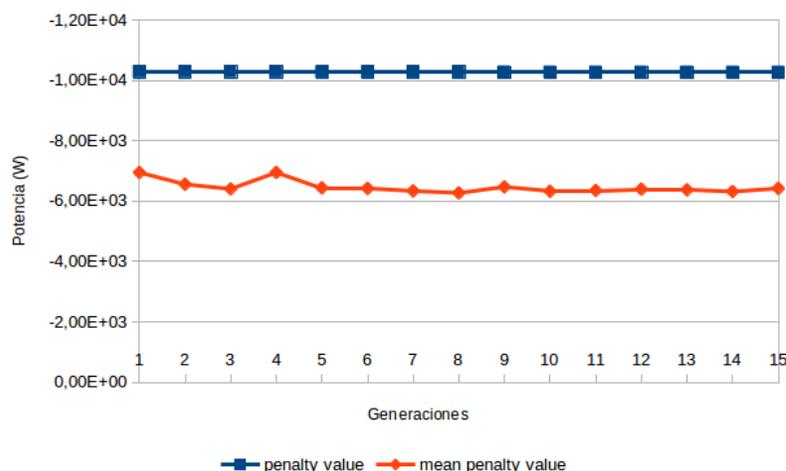


Figura 23. Evolución del algoritmo genético para campo discretizado.

El código tal y como se ha escrito evalúa un determinado número de posibles soluciones, pero no optimiza por él mismo hacia la mejor solución, con este problema deberán lidiar las personas que estudien en la línea de investigación del Departamento de Energía de la UPM. Se ha visto además en las variables *.mat* guardadas que el número de óptimos en cada generación es a veces mayor que uno, esto significa que el individuo óptimo se repite. Ya sea para mejorar el proceso

del algoritmo genético realizando parte de la reproducción de la población con el mejor individuo de la generación, o sea un error, no se ha podido controlar este fenómeno.

## 5.5. Optimización con optUPMPARK del caso referencia con el campo continuo.

En este caso se ha dado libertad de posición a los aerogeneradores en la optimización, pudiendo colocarse en cualquier posición con la única restricción de los límites del parque y el respeto de 200 metros entre aerogeneradores. La evolución del algoritmo genético se muestra en la figura 24.

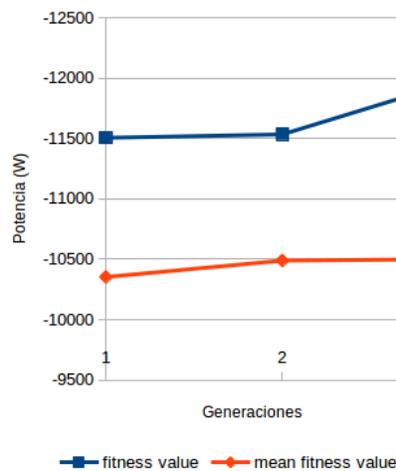


Figura 24. Evolución del algoritmo para el caso de parque continuo.

En el eje de abscisas se representan las generaciones y en el de ordenadas los valores de referencia de los *fitness values*, en nuestro caso, las potencias. Los puntos rojos son la media de los *fitness values* de los individuos de una generación y los puntos azules representan las configuraciones óptimas de cada generación a través del mejor *fitness value* de la población. El tiempo computacional ha sido de 23 horas y 10.19 minutos. El mensaje *"Exitflag: 1. Average change in value of the fitness function over options.StallGenLimit generations less than options.TolFun and constraint violation less than options.TolCon."* indica una salida correcta sin errores internos en el algoritmo, respetando las tolerancias mínimas. No obstante sorprende la corta evolución del algoritmo, hallando la solución óptima en la tercera generación.

En la figura 25 se representa la configuración óptima del parque eólico, con continuidad en la solución.

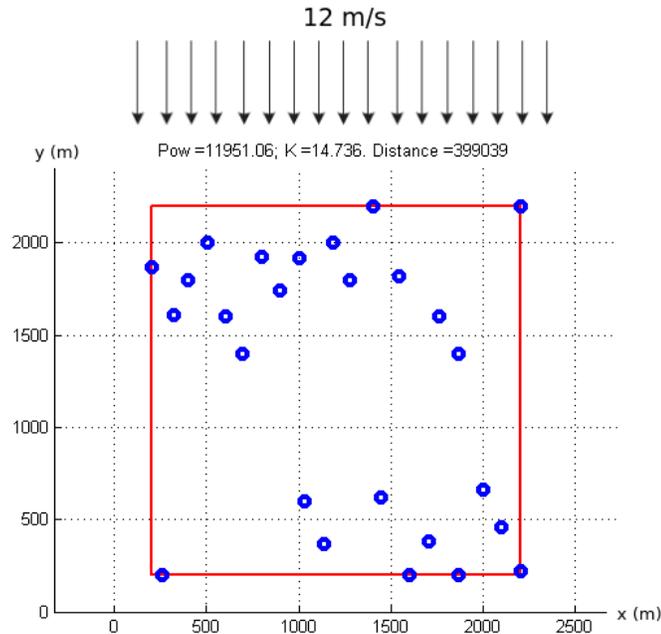


Figura 25. Configuración óptima del parque con continuidad en la solución.

Se puede observar la relativa predisposición en diagonales y en *zig-zag* de los aerogeneradores, la causa de este comportamiento se observa mejor en las representaciones de los campos de velocidad y de energía cinética turbulenta. En la figura 26 se puede observar el campo de velocidades de la configuración óptima salida de optUPMPARK para el caso referencia con un campo continuo en solución. En la figura 27 se representa la distribución de la energía cinética turbulenta. Las estelas interactúan más intensamente en los aerogeneradores cercanos entre sí en la dirección predominante del viento, generando déficits de velocidad y aumentos en la energía cinética turbulenta importantes (las estelas tienden a juntarse). Los resultados se obtienen respecto al flujo básico incidente en el parque, en este caso proveniente del norte. Los límites del parque no se han dibujado porque GRAFICA representa el cálculo con mayores dimensiones, pero los aerogeneradores están dentro de los límites del parque estudiado. Se observa el empleo del método de disco actuador para el rotor de cada aerogenerador en las zonas donde no se han representado las distribuciones de velocidad y de energía cinética turbulenta. La longitud inmediatamente después del aerogenerador en dirección del flujo aguas abajo, para la cual se emplea el método de disco actuador, se define en el parámetro TDWA escrita en la tabla 5.

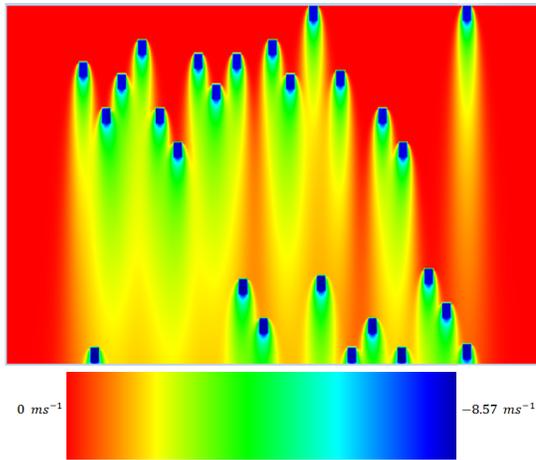


Figura 26. Campo de velocidad en  $ms^{-1}$  para el óptimo resultado por *optUPMPARK* para el caso referencia con campo continuo.

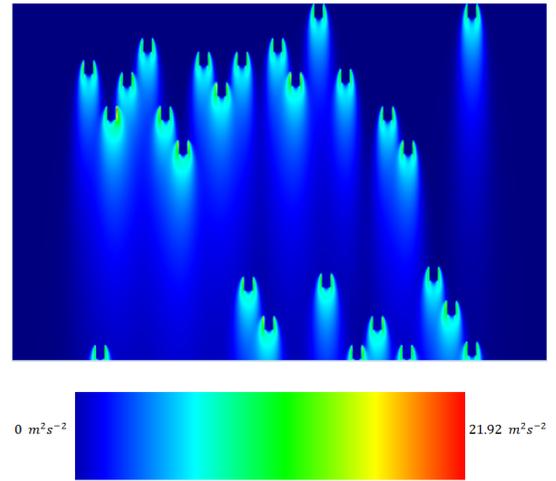


Figura 27. Campo de la energía cinética turbulenta en  $m^2s^{-2}$  para el óptimo resultado por *optUPMPARK* para el caso referencia con campo continuo

Se presentan ahora las posiciones optimizadas en la tabla 11, así como sus potencias y eficiencias de la configuración optimizada por *optUPMPARK* con continuidad en la solución.

Tabla 11. Datos del óptimo, resultado de simular el caso referencia con *optUPMPARK* y campo continuo.

Aerogenerador	X (m)	Y (m)	Potencia (kW)	Eficiencia
1	1034.9	600	365.99	0.71
2	800	1927.9	513.46	1.00
3	2000	665.8	491.48	0.96
4	1542.9	1821.2	511.22	1.00
5	1706.3	379.5	387.45	0.75
6	1186.3	2003.2	513.46	1.00
7	200	1871.6	513.46	1.00
8	895.4	1745	497.08	0.97
9	1868.4	1400	508.52	0.99
10	259.3	200	342.96	0.67
11	1442.7	621.9	360.48	0.70
12	1600	200	400.93	0.78
13	1000	1921.7	513.46	1.00
14	1759.8	1605	513.46	1.00
15	1400	2200	513.46	1.00
16	600	1600	473.48	0.92
17	400	1800	508.52	0.99
18	503.8	2000	513.46	1.00
19	1280.4	1800	487.78	0.95
20	2200	221.2	414.59	0.81
21	2095.2	461.9	447.76	0.87
22	1139.4	370.8	368.14	0.72
23	2200	2200	513.46	1.00
24	695	1400	466.36	0.91
25	1863.2	200	341.34	0.66
26	321.9	1607.1	469.30	0.91

Tabla 12. Comparación entre la configuración óptima de optUPMPARK con continuidad en la solución y los casos de las secciones previas.

Óptimo de Mosetti para el caso referencia evaluado con - UPMPARK		Óptimo con optUPMPARK para el caso referencia y campo - discretizado		Óptimo con optUPMPARK para el caso referencia y campo - continuo	
Potencia (kW)	Eficiencia (%)	Potencia (kW)	Eficiencia (%)	Potencia (kW)	Eficiencia (%)
9054	68	10349	78	11951	90

La mejora es notable en resultados, aumentando hasta el 90 % la eficiencia del parque y en más de 1600 KW la potencia respecto al óptimo del caso con campo discretizado. Se observa en la distribución de energía cinética turbulenta que varios aerogeneradores están afectados solo en parte del rotor por las estelas, esto afecta de forma más negativa que si estuviesen afectadas de forma homogénea en todo el rotor por el efecto de las estelas, un aspecto que no se entra a evaluar en la presente memoria pero que ha de tenerse en cuenta en un posible estudio de reducción en la vida útil de los aerogeneradores.

No obstante, se ha visto en las variables *.mat* guardadas que el número de óptimos en cada generación ha sido 1, 9 y 12 respectivamente. En un campo continuo cada individuo debe tener un *fitness value* distinto y por tanto solo un óptimo en cada generación. Se ha verificado que en generaciones con más de un óptimo estos son el mismo individuo repetido, no pudiendo saber como controlar este problema o si es normal que el algoritmo se comporte así. En cualquier caso la repetición en la población del óptimo puede hacer que se llegue antes a la tolerancia impuesta, debido al peso que da tener el mismo *fitness value* (potencia) varias veces repetido, pero no se puede afirmar que esta sea la causa de salida del algoritmo. Se ha realizado otra simulación para este mismo caso pero sin la restricción implementada de la distancia mínima. Para la simulación sin la restricción el algoritmo evoluciona con más generaciones y hacia una óptima, convergiendo de forma más lenta y sin respetar esa distancia mínima. En ese caso también existen generaciones con más de un individuo repetido, por lo tanto se descarta que la restricción esté mal implementada.

## 5.6. Optimización con optUPMPARK para una rosa de vientos con dos direcciones perpendiculares entre sí.

Como última simulación se ha evaluado el caso referencia cambiando la rosa de vientos. Ahora se han configurado dos direcciones de viento, una del norte y otra del oeste, ambas a  $12 \text{ m s}^{-1}$ . El resto de parámetros son iguales. El objetivo de esta simulación es observar el número de generaciones en el algoritmo y la apariencia de la configuración óptima salida del mismo.

La evolución del algoritmo genético en este caso es el representado en la figura 28. En el eje horizontal se representan las generaciones y en el vertical los *fitness values* (potencias). los puntos rojos son las medias de las potencias de los individuos en cada generación y los azules las mejores configuraciones de las mismas a través del mejor *fitness value* de la población.

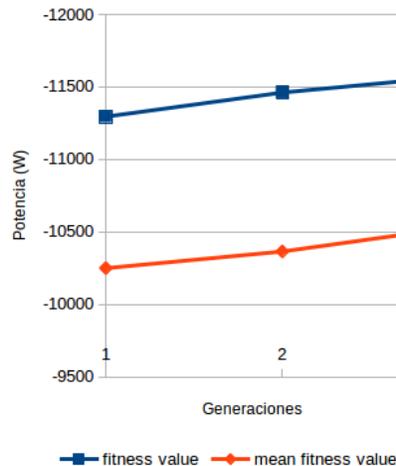


Figura 28. Evolución del algoritmo para una rosa de vientos bidireccional y parque continuo del caso referencia calculado por optUPMPARK.

El mensaje de salida del algoritmo genético no se recogió, pero se sabe que fue igual que para el caso de parque continuo con una rosa de vientos de una sola dirección (Norte). Este mensaje indica que la salida del algoritmo ha sido normal, sin errores internos. Además se sabe que el tiempo computacional ha sido mayor que en caso anterior. Sorprende que el número de generaciones en esta optimización también haya sido 3. Según se observa en la evolución del algoritmo (figura 28), este debería haber continuado simulando, ya que, si bien no se sabe exactamente la causa de salida, la tendencia es claramente positiva y seguramente podría seguir mejorando el óptimo. El número de óptimos (individuos repetidos) en las generaciones 1, 2 y 3 han sido 10, 3 y 20 respectivamente.

La configuración óptima para este caso se ha representado en la figura 29, ejes en metros. La potencia ponderada del parque asciende a 11575,99 KW.

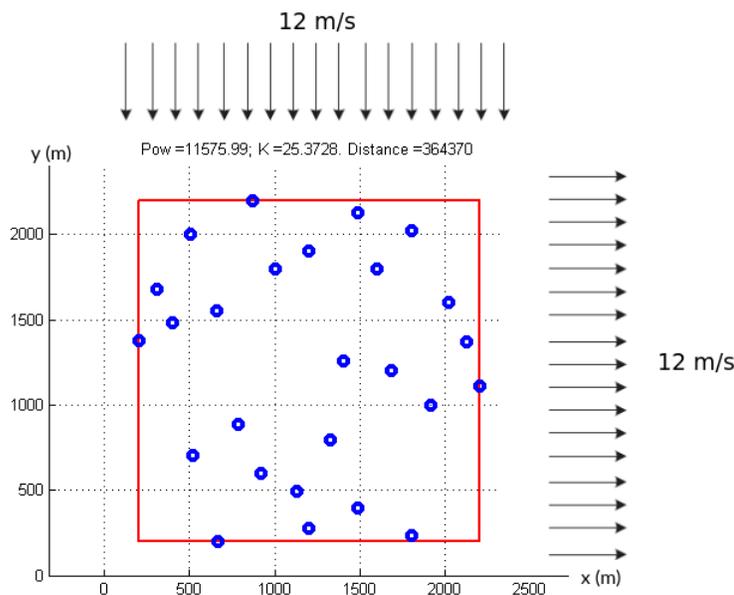


Figura 29. Configuración óptima para una rosa de vientos bidireccional y parque continuo del caso referencia calculado con optUPMPARK.

Se presenta ahora la configuración optimizada por optUPMPARK con continuidad en la solución en la tabla 13, así como sus potencias y eficiencias. La potencia y eficiencias ponderadas del parque son 11.576 y 0,87 respectivamente.

Tabla 13. Datos del óptimo resultado de simular el caso referencia con optUPMPARK y campo continuo y una rosa de vientos con dos direcciones.

Aerogenerador	X (m)	Y (m)	Potencia ponderada (kW)	Eficiencia ponderada
1	1800	233.2	319.13	0.62
2	1800	2023.9	434.63	0.85
3	1684.7	1200	395.74	0.77
4	1917.7	1000	457.96	0.90
5	310	1680.7	513.46	1.00
6	784.4	887.1	469.41	0.92
7	1325.8	800	388.2	0.76
8	200	1376.8	506.635	0.99
9	1600	1800	387.84	0.76
10	1490.3	2131.5	449.54	0.88
11	665.3	200	430.42	0.84
12	2125.6	1373	426.87	0.84
13	870.5	2200	513.46	1.00
14	1000	1800	498.11	0.97
15	1200	278.7	382.96	0.75
16	657	1557.6	482.65	0.94
17	1200	1907.9	481.83	0.94
18	2024.2	1600	422.62	0.83
19	920.9	600	425.05	0.83
20	2200	1109	437.39	0.86
21	400	1482.5	486.18	0.95
22	1129.5	494.3	454.46	0.89
23	522	703.8	433.17	0.85
24	1400	1261	463.36	0.91
25	1486.1	395.7	401.485	0.78
26	506.4	2000	513.46	1.00

## 6. Conclusiones e ideas de futuro.

Para finalizar se exponen las conclusiones extraídas de las simulaciones realizadas e ideas para seguir desarrollando el código propio optUPMPARK.

Existe un gran potencial desarrollado en los trabajos de esta línea de investigación. Se tiene una importante cantidad de estrategias para la modelización del campo eólico y para el proceso de optimización. En cuanto a la optimización de parques eólicos con el código propio optUPMPARK empleado en el departamento de Energía de la UPM se anotan las siguientes consideraciones para su mejora:

- El código de optimización actualmente sólo maximiza la potencia del parque, pero se puede tener en cuenta la turbulencia añadida por los aerogeneradores para ponderar su reducción de vida útil como lo hace UNION 2.2. También se puede ponderar en la función objetivo el coste debido al cableado, a través de la suma más baja de la distancia entre aerogeneradores de la configuración.
- Introducción de algoritmos de control para mejorar el proceso de optimización, ajustando dinámicamente la hibridación y la mutación del algoritmo genético.
- Añadir la posibilidad de instalar diferentes aerogeneradores con distintas alturas de buje y diámetros de rotor comerciales, como en el texto de Souma et al. [34].
- Optimizar la orientación del rotor para mejorar el rendimiento global del parque, desviando las estelas individuales de los aerogeneradores, como en el texto de Gebraad et al. [35].
- Proveer de una mayor complejidad a los contornos de la optimización, actualmente rectangulares, pudiendo existir parques complejos en contorno y zonas interiores en las cuales sea inviable la instalación de aerogeneradores.
- Introducir en la función objetivo aspectos económicos como el coste de la energía, de la infraestructura eléctrica ó del mantenimiento a lo largo de su vida útil.

El empleo de los modelos cinemáticos como el de Katic-Jensen están más indicados para la comparación de parámetros, algoritmos y estrategias de optimización. Por otra parte, los modelos de campo como UPMPARK están más indicados para su utilización en la evaluación de parques eólicos reales, ya que se tiene mayor coste computacional pero lleva a soluciones más consecuentes con los efectos presentes, tal como muestran las tablas 8 y 12. El proceso de optimización necesita de un código eficiente en tiempo y precisión y optUPMPARK lo cumple.

Como el artículo de Serrano [31] con el parque eólico de Middelgrunden ya existen estudios de parques eólicos *offshore* reales (Vindeby, Horns Rev), poseen mayor número de variables en la función objetivo, aún así en este artículo se utiliza también un modelo cinemático basado en

Katic-Jensen. Sería por tanto interesante la utilización de optUPMPARK en parques eólicos reales, para comparar la solución óptima salida con las mediciones reales y con los estudios previos de esos mismos parques que usen modelos cinemáticos. Actualmente sólo existe el Trabajo de Fin de Máster de Jimenez [6], en el que se optimiza el parque eólico de Vindeby, aportando una solución muy precisa respecto a la real existente.

Se ha verificado el buen funcionamiento de las restricciones no lineales para la distancia mínima entre aerogeneradores implementada, necesarias en la optimización. Otras subrutinas necesarias son la realización de unos límites del parque eólico más complejos o realistas (actualmente solo admite parques rectangulares) y la posibilidad tener zonas internas en el parque donde no se puedan instalar aerogeneradores.

El presente trabajo forma parte del estudio preliminar del código optUPMPARK. Dada la complejidad del mismo no ha sido posible solucionar algunos errores. En parques eólicos con solución discretizada el código optUPMPARK evalúa una población (tan grande como se quiera), pero no optimiza de una generación a otra, resultando siempre la configuración óptima de la primera generación. Un error de optUPMPARK en parques eólicos con continuidad en la solución es la repetición del individuo óptimo al crear la nueva población en cada generación, puesto que para los dos casos evaluados en esta memoria con un campo continuo en solución, tanto si se añade o no la restricción de la distancia mínima entre aerogeneradores, siempre se obtienen algunas generaciones con el individuo más óptimo repetido. Ya sea para mejorar el proceso del algoritmo genético realizando parte de la reproducción de la población con el mejor individuo de la generación, o sea un error, no se ha podido controlar este fenómeno.

Una última valoración a tener en cuenta deja la posibilidad de construir optUPMPARK con otro lenguaje de programación como Python, lenguaje de programación libre de alto nivel con potencia de cálculo en métodos numéricos y en representatividad. Esto llevaría a menores tiempos computacionales al construirse específicamente para el problema de parques eólicos, además de poder controlar todo el proceso del algoritmo genético más fácilmente.

## Bibliografía

- [1] CASTRO CARRANZA C., MEDIAVILLA PASCUAL M., MIGUEL GONZALEZ L.J., FRECHOSO ESCUDERO F.: *"Global wind power potential: Physical and technological limits"*. Energy Policy 39, no. 10 (October 2011): 6677–82. doi:10.1016/j.enpol.2011.06.027.
- [2] <https://es.wikipedia.org/>
- [3] <https://www.enerdata.net/>
- [4] 2016 European statistics. Published in February 2017. <https://windeurope.org/>
- [5] <http://informeestadistico.idae.es/t10.htm>
- [6] JIMENEZ MURCIA F.J.: *"Optimización de la disposición de aerogeneradores en un parque eólico offshore"*, Trabajo de Fin de Master. Directores del trabajo: Migoya Valor E. y Muñoz Paniagua J. ETSII | UPM. Septiembre 2015.
- [7] <http://elperiodicodelaenergia.com/los-costes-de-la-eolica-offshore-mantienen-en-vilo-a-la-industria/>
- [8] CRESPO MARTINEZ A., HERNANDEZ RODRIGUEZ J. Y FRANDBSEN S.T.: *"Survey of Modelling Methods for Wind Turbine Wakes and Wind Farms"*. En: WIND ENERGY. Wind Energ., 2, 1–24 (1999).
- [9] SANDERSE B., VAN DER PIJL S. P. Y KOREN B.: *"Review of computational fluid dynamics for wind turbine wake aerodynamics"*. En: WIND ENERGY. Wind Energ. 2011; 14:799–819 Published online 4 February 2011 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/we.458
- [10] JENSEN N. O.: *"A note on wind generator interaction"*. Risø National Laboratory, Roskilde Denmark, 1984. Risø-M-2411.
- [11] KATIC I., HØJSTRUP J. Y JENSEN N. O.: *"A simple model for cluster efficiency"*. Proc. EWEC'86, Rome, 1986, Vol. 1, pp. 407–410.
- [12] LARSEN G.C., MADSEN H.A., AND SØRENSEN N.N.: *"Mean wake deficit in the near field"*. European Wind Energy Conference and Exhibition 2003, Madrid, 2003.
- [13] FRANDBSEN S.T., BARTHELMIE R., PRYOR S., RATHMANN O., LARSEN S., HOJSTRUP J., AND THØGERSEN M.L.: *"Analytical modelling of wind speed deficit in large offshore wind farms"*. En: Wind Energy, 9, 2006. DOI: 10.1002/we.189. SN: 1099–1824.
- [14] HERBERT-ACERO J.F., PROBST O., RÉTHORÉ P.E., LARSEN G.C. Y CASTILLO-VILLAR K.K.: *"A Review of Methodological Approaches for the Design and Optimization of Wind Farms"*. En: Energies 2014, 7, 6930–7016; doi:10.3390/en7116930. ISSN 1996-1073 [www.mdpi.com/journal/energies](http://www.mdpi.com/journal/energies).

- [15] RÉTHORÉ P.E.: "*Wind turbine wake in atmospheric turbulence*". PhD Thesis, Aalborg University, 2009. ISSN 1901-7294.
- [16] CABEZÓN D., MIGOYA E. AND CRESPO A.: "*Comparison of turbulence models for the computational fluid dynamics simulation of wind turbine wakes in the atmospheric boundary layer*". En: Wind Energ. 2011; 14:909–921.
- [17] MOSETTI G., POLONI C., DIVIACCO B.: "*Optimization of wind turbine positioning in large windfarms by means of a genetic algorithm*". En: Journal of Wind Engineering and Industrial Aerodynamics, Issue 51, 105–116. 1994.
- [18] GÓMEZ ELVIRA R., MIGOYA VALOR E.: "*Programa UPMPARK para el modelado de parques eólicos*". E.T.S.I. Industriales | UPM, Madrid, España, 2010.
- [19] GÓMEZ-ELVIRA GONZÁLEZ R.: "*Aplicación de un modelo algebraico del tensor de Reynolds a la modelización de estelas de aeroturbinas en atmósfera neutra*". Tesis doctoral. E.T.S.I. Industriales | UPM, Madrid, España, 1999.
- [20] LAUNDER B.E. AND SPALDING D.B.: "*The numerical computation of turbulent flows*". En: Computer Methods in Applied Mechanics and Engineering 3 (2): 269–289. ISSN 00457825. 1974.
- [21] PANOFSKY H.A. AND DUTTON J.A.: "*Atmospheric Turbulence*". Willey Interscience. 1984.
- [22] DEP. INGENIERÍA ENERGÉTICA Y FLUIDOMECÁNICA. "*Programa UNION 2.2 para emplear el UPMPARK con una Rosa de Vientos*". E.T.S.I. Industriales | UPM, Madrid, España, 2010.
- [23] DEP. INGENIERÍA ENERGÉTICA Y FLUIDOMECÁNICA. "*Guía breve programa GRAFICA para representar los resultados del UPMPARK*", E.T.S.I. Industriales | UPM, Madrid, España, 2002.
- [24] CHARLES DARWIN, M.A.: "*The Origin of Species*". 1859.
- [25] GESTAL M., RIVERO D., RABUÑAL J.R., DORADO J., PAZOS A.: "*Introducción a los Algoritmos Genéticos y la Programación Genética*". ISBN: 978-84-9749-422-9. Depósito legal: C 3193-2010. Servicio de Publicacións, Universidade da Coruña. 2010.
- [26] DU PONT B.L. AND CAGAN J.: "*An Extended Pattern Search Approach to Wind Farm Layout Optimization*". En: Journal of Mechanical Design. August 2012, Vol. 134 / Issue 081002, 1–18. DOI: 10.1115/1.4006997. 2012.
- [27] MARMIDIS G., LAZAROU S., PYRGIOTI E.: "*Optimal placement of wind turbines in a wind park using Monte Carlo simulation*". En: Renewable Energy. 2008; 33(7):1455–60.

- [28] MITTAL A.: *"Optimization of the layout of large wind farms using genetic algorithm"*. MS Thesis. Department of Mechanical and Aerospace Engineering, Case Western Reserve University; 2010.
- [29] EMAMI A., NOGHREH P.: *"New approach on optimization in placement of wind turbines within wind farm by genetic algorithms"*. En: Renewable Energy. 2010; 35(7):1559–64.
- [30] SHAKOOR R., HASSAN M.Y., RAHEEM A. AND RASHEED N.: *"The Modelling of Wind Farm Layout Optimization for the Reduction of Wake Losses"*. En: Indian Journal of Science and Technology, Vol 8(17), 69817, August 2015. ISSN (Print): 0974-6846 ISSN (Online): 0974-5645.
- [31] SERRANO GONZÁLEZ J., BURGOS PAYÁN M., RIQUELME SANTOS J.M, GONZÁLEZ-LONGATT F.: *"A review and recent developments in the optimal wind-turbine micro-siting problem"*. En: Renewable and Sustainable Energy Reviews 30 (2014) 133–144.
- [32] YANG J., ZHANG R., SUN Q., AND ZHANG H.: *"Optimal Wind Turbines Micrositing in Onshore Wind Farms Using Fuzzy Genetic Algorithm"*. En: Hindawi Publishing Corporation. Mathematical Problems in Engineering. Volume 2015, Article ID 324203, 9 pages. <http://dx.doi.org/10.1155/2015/324203>
- [33] TURNER S.D.O., ROMERO D.A., ZHANG P.Y., AMON C.H., CHAN T.C.Y.: *"A new mathematical programming approach to optimize wind farm layouts"*. En: Renewable Energy 63 (2014) 674–680.
- [34] CHOWDHURY S., ZHANG J., MESSAC A., CASTILLO L.: *"Optimizing the arrangement and the selection of turbines for wind farms subject to varying wind conditions"*. En: Renewable Energy 52 (2013) 273–282.
- [35] GEBRAAD P.M.O., TEEUWISSE F.W., VAN WINGERDEN J.W., FLEMING P.A., RUBEN S.D., MARDEN J.R. AND PAO L.Y.: *"Wind plant power optimization through yaw control using a parametric model for wake effects—a CFD simulation study"*. En: WIND ENERGY Wind Energ. (2014). Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/we.1822
- [36] INFORMATION MATLAB *"Chapter 8. Genetic Algorithm Implementation Using Matlab"*. Manual de la *toolbox* de Matlab.
- [37] KATIC I., HØJSTRUP J. AND JENSEN N.O.: *"A simple model for cluster efficiency"*. European Wind Energy Conference and Exhibition. Rome, Italy, 7-9 October 1986.
- [38] SAMORANI M.: *"The Wind Farm Layout Optimization Problem"*. P. M. Pardalos et al. (eds.), Handbook of Wind Power Systems, Energy Systems, DOI: 10.1007/978-3-642-41080-2\_2. Copyright Springer-Verlag Berlin Heidelberg 2013.



## A. Anexo I: Tendencias actuales en la optimización de parques eólicos.

En la actualidad el centro de interés de la optimización de parques eólicos (WFDO, siglas en inglés de Wind Farm Design and Optimization) se enfoca en resolver preguntas como: ¿dónde colocar los aerogeneradores?, ¿qué diámetros de rotor deben tener? o ¿qué alturas de buje para los aerogeneradores?, todo con el objetivo que se plante. Los métodos de cálculo del campo fluido son clave para conseguir la mejor configuración de parque posible bajo el objetivo que perseguimos, debe ser preciso y fiable. No solo caben en el objetivo aspectos de beneficio económico, sino también encontrar un equilibrio con cuestiones sociales y medioambientales. Bajo ese marco se han de evaluar la potencia, la turbulencia, costes de cableado (son altos en parques *offshore*), costes derivados del impacto ambiental, etc. Los artículos resumidos son muy recientes, la mayoría entre 2013 y 2015, y por lo tanto son de gran vigencia, aportándonos una visión detallada de qué se está realizando para resolver las preguntas antes planteadas y las diversas estrategias seguidas para sus resoluciones.

### Ejemplos de estudios recientes:

Marmadis et al. [27] intentó la optimización de Masetti para una dirección y magnitud de viento constantes usando el algoritmo de Monte Carlo para la optimización.

Mital A. [28] recomendó usar algoritmos genéticos, simuló el diseño del campo eólico para 3 escenarios y propuso incrementar el tamaño del parque en  $2000 \times 2200$  metros en vez de un parque cuadrado de 2000 metros lado. El espacio entre posiciones se redujo a 1 metro cuadrado, con esta precisión en el dominio se consiguió disminuir el coste por unidad de potencia de parque (función objetivo CoE).

Emami et al. [29] propuso mejorar el diseño de la optimización con el modelo de estela de Jensen, modificando la función objetivo, teniendo en cuenta la eficiencia de los aerogeneradores (*Wind Turbines*, WTs) y el coste del parque.

En el trabajo de Rabia Shakoor et al. [30] se evalúa un modelo de estela de Jensen con una función sencilla para calcular la potencia de cada aeroturbina, una función objetivo CoE y otra en función de la eficiencia del parque, desarrollando las técnicas de optimización de algoritmos genéticos. Lo distinto al resto de trabajos es que la dirección del viento es en diagonal, añadiendo una mayor dimensión en la sección transversal de la dirección del viento, un truco que no aporta mucho, ya que la configuración del ejemplo es distinta respecto a los anteriores, pero el modelo de estela y el método de optimización son iguales.

Serrano et al. [31] optimiza el parque de Middelgrunden (*offshore*), evaluándolo en la forma actual, con restricciones simétricas y optimizándola en una región de dominio continua, con

una función objetivo AEP (maximización de la producción de energía anual). El modelo de cálculo de estelas empleado es el de Katic. Tuvieron en cuenta otros factores como el terreno disponible (delimitación más compleja del parque), zonas de imposibilidad de instalación de WTs, distancia mínima entre aerogeneradores, consideraciones de impacto medioambiental y aspectos económicos de infraestructura eléctrica. Utilizaron varios algoritmos de optimización metaheurísticos.

Jun Yang et al. [32] tomaron el modelo de cálculo de estela de Jensen, tuvieron en cuenta el efecto del terreno (topografía) en parques *onshore*, un modelo económico de Valor Anual Neto (*Net Annual Value*, NAV) para la función objetivo, optimización mediante algoritmos genéticos con control *fuzzy* (método para ajustar dinámicamente la hibridación y mutación del GA). Todo ello para evaluar un parque *onshore* chino, lo que supone la extensión de los trabajos de optimización de parques eólicos *onshore*, que escasean en la literatura.

S.D.O. Turner et al. [33] modelaron la estela siguiendo a Jensen, cambiaron la estrategia para condicionar las localizaciones de los WTs, asumiendo que la potencia es proporcional al cubo de la velocidad. Bajo un dominio discreto, su objetivo fue maximizar una función dependiente de los déficits de velocidad generados para calcular las velocidades en los puntos donde existen aerogeneradores. Esta función objetivo la generalizaron para poder interpretar una rosa de vientos completa. Luego, linealizaron esta función y la restringieron, comparando su modelo de aproximación WFLO cuadrática de enteros para la interacción entre estelas con otros estudios previos como la formulación MILP (Programación lineal de enteros mixta), obteniendo buenos resultados. Con su método optimizaron los casos de Mosetti et al. [17].

Souma Chowdhury et al. [34] apostaron por avanzar en la metodología UWFL0 (Unrestricted Wind Farm Layout Optimization). Sus objetivos fueron calcular la producción de energía anual del parque, modificar el modelo de generación de potencia para permitir múltiples tipos de turbinas comerciales en el parque (diferentes alturas de buje, diámetros de rotor y otras características), evaluar el coste del parque eólico usando el modelo *Response Surface-based Wind Farm Cost* (RS-WFC) e implementar un nuevo desarrollo del algoritmo de optimización de enjambre de partículas (PSO) para resolver el problema. Este estudio arroja una posibilidad importante, el hecho de que que la estela sea aproximadamente cónica aporta "jugabilidad" en la dirección vertical, pudiendo tener configuraciones interesantes con aerogeneradores de varias alturas de buje, entre otros aspectos.

Otro texto de interés es el estudio de P. M. O. Gebraad et al. [35]. El aspecto más destacado es la optimización del ángulo de orientación del rotor. Deflectando las estelas se empeoran las eficiencias de algunos aerogeneradores pero se mejora la global del parque eólico al reducir la incidencia de las estelas a WTs aguas abajo de las primeras. Utilizaron el simulador SOWFA (*Simulator for Onshore/Offshore Wind Farm Applications*), el cual integra un método LES de flujo de viento tridimensional en la capa límite atmosférica. Para modelar el rotor se usó el método de línea actuadora. Cada turbina en la simulación puede ser controlada usando un algoritmo implementado en FAST (*Fatigue, Aerodynamics, Structures and Turbulence*, herramienta

de simulación para sólidos aeroelásticos). Para el modelo de estela se realizó una combinación estudiada previamente basada en la de Jensen, mejorado y añadiendo el control de giro de la orientación del rotor (FLORIS). El modelo FLORIS se esquematiza como se ve en la figura 30. En esta se observan los distintos ángulos relativos, fuerzas sobre el aerogenerador y evolución de la estela aguas abajo de este, desde una vista aérea. La optimización se realizó siguiendo la aproximación de la teoría de juego, realizando simulaciones con 6 aerogeneradores.

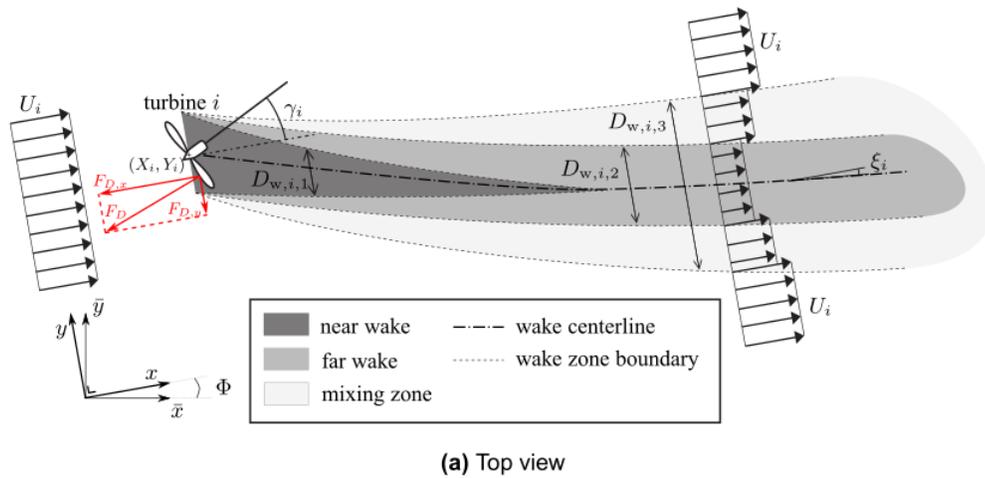


Figura 30. Modelado de estela FLORIS (control de la orientación del rotor). [35].



## B. Anexo II: Funciones del Genetic Algorithm Toolbox (GAT) utilizadas. Archivos empleados en el algoritmo genético.

En esta sección se va a intentar guiar a la persona lectora en la comprensión de la herramienta de Matlab utilizada en el presente TFG. Se basará en el texto “*Genetic-Algorithm-Implementation-Using-Matlab*” [36]. Se focalizará la explicación al código optUPMPARK desarrollado por el Departamento de Energía de la UPM.

Existen dos principales cajas de herramientas (*toolbox*) en Matlab para la optimización mediante algoritmos genéticos:

- “*The Genetic Algorithm Toolbox*”. Esta es la utilizada para la construcción de optUPMPARK.
- “*The Genetic and Evolutionary Algorithm Toolbox*”. Más amplia en opciones que la anterior.

La puesta a punto del algoritmo genético en “*The Genetic Algorithm Toolbox*” se puede realizar mediante dos posibles vías, bien introduciendo los parámetros manualmente en los archivos *\*.m* que se construyan para aportar dichas entradas, bien mediante la interfaz “*Genetic Algorithm Graphical User Interface Toolbox*”, cargándola al teclear en la línea de comandos de Matlab `>>gatool`. En el caso de optUPMPARK las entradas se realizan manualmente en el script principal `main.m`, el cual lleva programada la secuencia de llamada a otras funciones para recoger los datos necesarios en el algoritmo, así como otros aspectos gráficos.

Principalmente un algoritmo genético en Matlab consta de las siguientes estructuras de datos:

- Cromosomas (*chromosomes*).
- Valores de la Función Objetivo (*objective function values*).
- Valores de comportamiento o rendimiento (*fitness values*).

La estructura de datos de los cromosomas almacena la población entera en una matriz de tamaño  $N_{ind} \times L_{ind}$ , donde  $N_{ind}$  es el número de individuos de la población y  $L_{ind}$  la longitud del genotipo que representa a dichos individuos. En optUPMPARK esta estructura es la población, cada individuo está compuesto por  $n$  turbinas y  $2n$  variables (posiciones  $x$  e  $y$ ), que sería el genotipo.  $N=popSize$  y  $L=nvars$  en optUPMPARK.

Los fenotipos son las variables de decisión, calculadas mediante una transformación de la representación de los cromosomas. Aquí, cada cadena decodifica la estructura de los cromosomas en un vector fila de orden  $Nvar$ , según el número de dimensiones en el espacio de búsqueda y al valor correspondiente del vector de variables de decisión. Estas variables de decisión se almacenan en una matriz numérica  $Nind \times Nvar$ , donde cada fila corresponde a un fenotipo particular de un individuo. La función objetivo se usa para evaluar el rendimiento de los fenotipos en el dominio del problema. Los valores de la función objetivo resultantes pueden ser escalares o vectoriales (si el problema es multiobjetivo). Los *objective function values* no tienen porque ser los mismos que los *fitness values*. Estos valores se almacenan en una matriz numérica de tamaño  $Nind \times Nobj$ , donde  $Nobj$  es el número de objetivos. Cada fila corresponde a un vector objetivo de un individuo particular. En optUPMPARK la función objetivo es `executeUPMPARK.m`, dentro de la cual se ejecuta `upmpark.exe`. Los *fitness values* son una derivación de los valores salidos de la función objetivo escalando o elaborando un ranking de los mismos, los *fitness values* son escalares no negativos y se almacenan en vectores columna de longitud  $Nind$ . En optUPMPARK se tiene un *fitness value* para cada individuo de la población de cada generación, que será la potencia total del parque representando de esta manera lo buenas que son las distintas configuraciones de las aeroturbinas (Wind Turbines, abreviado WTs).

Son varias las funciones disponibles en Matlab para los algoritmos genéticos, la *GA Toolbox Functions* es una colección de funciones que ayudan y facilitan la interpretación del entorno de optimización. Las *toolboxes* son librerías estándar de algoritmos u otras rutinas matemáticas para resolver problemas, consistentes en funciones predefinidas. El algoritmo genético busca directamente en la *toolbox* correspondiente de Matlab la función que se le especifica mediante otro paquete llamado *Direct Search Toolbox*. Para construir optUPMPARK se ha usado la caja de herramientas denominada *Genetic Algorithm Toolbox*. Se llama a las funciones deseadas a través de *options*. Se puede llamar a funciones ya hechas o predefinidas ó crear una función propia con el formato y sintaxis que tiene preestablecido el GA (Gentic Algorithm) para lo que se desee.

Algunas funciones predefinidas de interés para entender optUPMPARK son:

- **Solvers:**
  - `ga` → *solver* del algoritmo genético.
- **Creation Function:**
  - `@gacreationnonlinearfeasible` → Para crear puntos disponibles en un problema no lineal.
- **Accesing Options (Acceso a *options*):**
  - `@gaoptimset` → crea/modifica la estructura de las opciones del GA.
- **Fitness scaling for GA (sobre la forma de escalar el rendimiento del GA)**

- @fitscalingrank → Escala basada en el *fitness value*.
- Selection for the GA (forma de elección de los padres en el GA):
  - @selectionroulette → Elige a los padres mediante el método de la ruleta.
- Crossover Recombination functions for GA (modo de entrecruzamiento o recombinación del GA).
  - @crossoversinglepoint → Un solo punto de cruzamiento.
- Mutation functions for GA (modo de tratar las mutaciones en el GA).
  - @mutationuniform → Mutación de puntos de forma uniforme.
- Plot functions for GA (gráficas de salida del GA):
  - @gaplotbestf → grafica la mejor marca y y la media de las marcas de los individuos.
  - @gaplotbestindiv → Representa el mejor individuo en forma de gráfica de barras.
  - @gaplotstopping → grafica que aporta información de los criterios de parada.
- Output functions for the GA (funciones para mostrar parámetros del GA y de sus opciones).
- Custom search functions for pattern search (juega con los métodos de búsqueda).
- Plot functions for pattern search (funciones para graficar búsquedas).
- Output functions for pattern search (funciones de salida de búsquedas).
- Utility functions (funciones que pueden ser útiles).
- Help files for GA (ficheros de ayuda para GA).

```

% Set options
options = gaoptimset('PopulationSize', popSize, 'InitialPopulation', initPop,...
    'CrossoverFraction', 0.8, 'Generations', generations, 'CreationFcn',...
    {@gacreationnonlinearfeasible, 'UseParallel', true, 'NumStartPts', 20},...
    'FitnessScalingFcn', @fitscalingrank, 'SelectionFcn', @selectionroulette,...
    'TolCon', 1e-8, 'TolFun', 1e-8, 'CrossoverFcn', @crossoversinglepoint,...
    'MutationFcn', {@mutationuniform 0.2 }, 'OutputFcns', @gaoutfnc, 'PlotInterval', 1,...
    'PlotFcns', {@gaplotbestindiv, @gaplotbestf, @gaplotstopping});

%gaoutput es la función que genera una matriz estructura en cada iteración.

```

Figura 31. Opciones del algoritmo genético

En la figura 31 se observa la sintaxis de Matlab para activar estas funciones, también algún otro parámetro como la probabilidad de mutación, que se realiza en este caso de forma uniforme con probabilidad del 20%. Las funciones predefinidas utilizadas para graficar son las que se introducen en *PlotFcns*. Además existe una personalizada, que guarda los principales valores de cada generación, mostrada en la figura 32, denominada *gaoutfnc.m*:

```
function [state, options,optchanged] = gaoutfnc(options,state,flag)

% Tomada de https://www.researchgate.net/post/How_to_output_the_value_of_each_iteration_in_Matlab_for_Genetic_Algorithm_and_Simulated_Annealing
% Copyright 2004-2006 The MathWorks, Inc.
% $Revision: 1.1.6.5 $ $Date: 2007/08/03 21:23:22 $

optchanged = false;

switch flag
case 'init'
disp('Starting the algorithm');
case 'iter'
disp('Iterating ...')
if state.Generation==0
else
fname=[pwd, '\', num2str(state.Generation), '.mat'];
mejores_individuos=find(state.Score == state.Best(state.Generation));
state.CantidadOptimos=length(mejores_individuos);
save(fname, 'state')
vector=state.Population(mejores_individuos(1),:);

showResults(vector); % Muestra en cada generación la
% mejor distribución de aerogeneradores.
mejores_individuos=[];
end
case 'done'
disp('Performing final task');
fname=[pwd, '\', num2str(state.Generation), '.mat'];
save(fname, 'state')
end
end
```

Figura 32. Función *gaoutfnc.m*

En el siguiente enlace se puede encontrar información de cómo realizar una función personalizada con su estructura y sintaxis (Ver “Output Function Options”):

<https://es.mathworks.com/help/gads/genetic-algorithm-options.html#f17837>

Para definir en la función *ga* un dominio espacial continuo para la solución en la optimización se ha realizado la subrutina que se ve en la figura 33:

```
% @constrain_dist_min; Función de restricciones de la distancia mínima.
% Calculates optimum
[x,fval, exitflag] = ga(ObjectiveFunction,nvars,[],[],[],[],LB,UB, @constrain_dist_min,[], options);
```

Figura 33. Llamada al algoritmo genético en el script *main.m*.

Las restricciones no lineales se introducen en la función `ga` con la variable `NONLCON`. En `optUPMPARK` se ha creado una función para que se respete la distancia mínima entre aerogeneradores que se desee, esta distancia se introduce en la variable `D` de la función `getParameters.m`. Esto se muestra en la figura 35 del anexo C.

En concreto se definen en la función los vectores `C` y `Ceq` que representan las ecuaciones en desigualdades y en igualdades no lineales. Estas restricciones actúan dentro del contorno impuesto en las variables LB (*Lower Bound*, contorno inferior) y UB (*Upper Bound*, contorno superior). La función `ga` posee la siguiente sintaxis:

Variabes de salida del algoritmo genético:

- `x` → óptimo de la evaluación.
- `fval` → valor fitness de dicho óptimo.
- `exitflag` → describe las condiciones de salida del GA (ver `>>helpwin ga`)

Argumentos de entrada al algoritmo genético:

- `x` → es la solución final del algoritmo. En nuestro caso es el parque óptimo.
- `fval` → es el *fitness value* del individuo óptimo final.
- `exitflag` → nos dice cómo ha acabado el algoritmo. Si todo ha ido correctamente este valor será un 1.
- `ObjectiveFunction` → `executeUPMPARK.m`. Función objetivo.
- `nvars` → número de variables de la función objetivo, en nuestro caso numero de coordenadas ( $2*nTurbines$ ).
- `primer[ ]` → matriz con las restricciones lineales del problema, desigualdades →  $A*X \leq B$ .
- `segunda[ ]` → matriz B de constantes para las desigualdades lineales.
- `tercera[ ]` → matriz con las restricciones no lineales del problema, desigualdades →  $Aeq*X \leq beq$ .
- `cuarta[ ]` → matriz beq de constantes para las desigualdades no lineales.
- `LB` → conjunto de restricciones de contorno inferiores.

- *UB* → conjunto de restricciones de contorno superiores.
- `@constrain\_dist\_min` → NONLCON, función que porta los vectores de las restricciones no lineales (igualdades y desigualdades) del problema (distancias mínimas entre turbinas).
- *quinto*[ ] → vector para indicar que variables deben ser números enteros (para discretizar el parque y hacer posiciones fijas).
- *options* → se introducen aquí las opciones definidas anteriormente en el GA.

## C. Anexo III: Fichero implementación de la condición de distancia mínima entre aerogeneradores.

Como se ha visto en la parte principal de la memoria se han implementado a la función `ga` las restricciones no lineales de la distancia mínima entre aerogeneradores, concretamente la ecuación 2:

$$S(i, j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (2)$$

Se implementó según la teoría y ayuda de Matlab *online*. La forma de procesar la ecuación 2 es crear una función personalizada e introducirla como parámetro `nonlcon` como una *handle function*. Se muestra a continuación la subrutina implementada.

```
% Condición distancia mínima.
function [c, ceq]=constrain_dist_min(vector_positions)

[nTurbines, ~, ~, ~, ~, ~] = getParameters();
c=[];
% Para nTurbines.
[x,y]=splitVector(vector_positions);

k=0;
for i=1:(nTurbines-1)
    for j=(i+1):(nTurbines)
        k=k+1;
        combinaciones(k,1)=i;
        combinaciones(k,2)=j;
    end
end

n=length(combinaciones);
for i=1:n
    c(i)=1-sqrt((x(combinaciones(i,1))-x(combinaciones(i,2))).^2+(y(combinaciones(i,1))-y(combinaciones(i,2))).^2);
end
ceq=[];
end
```

Figura 34. Subrutina `constrain_dist_min.m` para restringir la distancia mínima entre aerogeneradores.

Como se puede observar en la figura 34, para automatizar la entrada de restricciones no lineales en forma de desigualdades se dispuso de la creación mediante bucles de las posibles combinaciones sin repeticiones.



## D. Anexo IV: Modelo teórico del estudio de Mosetti et al.

Mosetti et al. [17] fueron pioneros en la optimización de parques eólicos mediante algoritmos genéticos. El modelo utilizado en ese trabajo para evaluar las estelas producidas por los aerogeneradores es el desarrollado por N.O. Jensen et al. [37], modelo numérico bastante simple. Este modelo asume la conservación de cantidad de movimiento dentro de la estela, aumentando las dimensiones de esta linealmente en dirección predominante del viento, tal como se muestra en la ecuación 3 :

$$u = u_0 \left[ 1 - \frac{2a}{\left(1 + \alpha \left(\frac{x}{r_1}\right)\right)^2} \right] \quad (3)$$

donde  $a$  es el factor de inducción axial,  $x$  la distancia aguas abajo de la turbina,  $r_1$  el radio del rotor aguas abajo y  $\alpha$  la constante de arrastre. La figura 35 muestra un esquema visual de este modelo y de los parámetros que se acaban de mencionar ( $u_0 = v_0$  es la velocidad incidente del flujo básico).

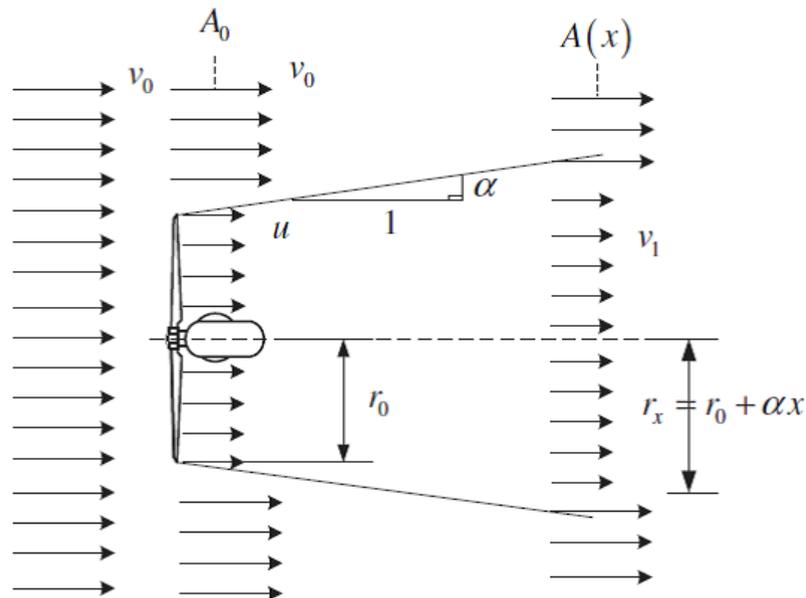


Figura 35. Modelo de estela de Jensen. [10].

El coeficiente de empuje se supone constante mediante la ecuación 4 :

$$C_T = 4a(1 - a) \quad (4)$$

y el coeficiente  $\alpha$  se calcula con la ecuación 5 :

$$\alpha = \frac{0,5}{\ln\left(\frac{z}{z_0}\right)} \quad (5)$$

donde  $z_0$  es la rugosidad de la superficie. Jensen supone un modelo numérico sencillo para evaluar la interacción y superposición entre estelas asumiendo que el cuadrado del déficit de energía cinética de una estela mixta es la suma de los cuadrados de los déficits de energía cinética de las estelas que la formen. La velocidad resultante aguas abajo de  $n$  turbinas se puede calcular a partir de la ecuación 6:

$$\left(1 - \frac{\bar{u}}{u_o}\right)^2 = \sum_{i=1}^n \left(1 - \frac{u_i}{u_o}\right)^2 \quad (6)$$

Este modelo se representa visualmente en la figura 36.

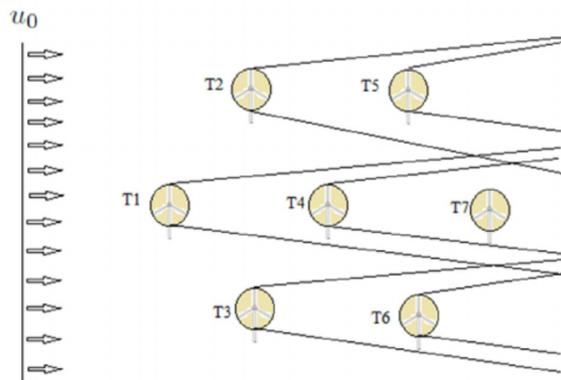


Figura 36. *Modelo de interacción de varias estelas de Jensen.* [10].

La función objetivo que utiliza Jensen es la representada en la ecuación 7. Tiene por tanto en cuenta minimizar el coste y maximizar la potencia generada en el parque.

$$Objective = \frac{1}{P_{tot}}w_1 + \frac{cost_{tot}}{P_{tot}}w_2 \quad (7)$$

donde  $P_{tot}$  es la energía (no la potencia) total producida anualmente,  $w_1$  y  $w_2$  son elegidos arbitrariamente y  $cost_{tot}$  es el coste/año del parque eólico. El parámetro  $cost_{tot}$  lo calcularon con la ecuación 8:

$$cost_{tot} = N_t \left( \frac{2}{3} + \frac{1}{3}e^{-0,00174N_t^2} \right) \quad (8)$$

donde  $N_t$  es el número de turbinas instaladas en el parque eólico. En cuanto a las aeroturbinas, Mosetti et al. no restringieron su número, pero dispusieron de un solo tipo con una altura de buje de  $h = 60$  m, 40 m de diámetro del rotor, coeficiente de empuje igual a  $C_T = 0.88$  y constante para todas las velocidades, la rugosidad de la superficie es  $z_0 = 0.3$  y una curva de potencia dada por el sistema de ecuaciones 9 sacada del trabajo de Michele Samorani [38]:

$$P(U) = \begin{cases} 0 & \text{si } U < 2 \\ 0,3U^3 & \text{si } 2 \leq U < 12.8 \\ 629,1 & \text{si } 12.8 \leq U < 18 \\ 0 & \text{si } U \geq 18 \end{cases} \quad (9)$$

donde  $U$  es la velocidad, expresada en m/s y  $P$  la potencia, en kW, representada en la figura 37. En el eje de abscisas se representa la velocidad del viento en  $ms^{-1}$  y en ordenadas la potencia del aerogenerador.

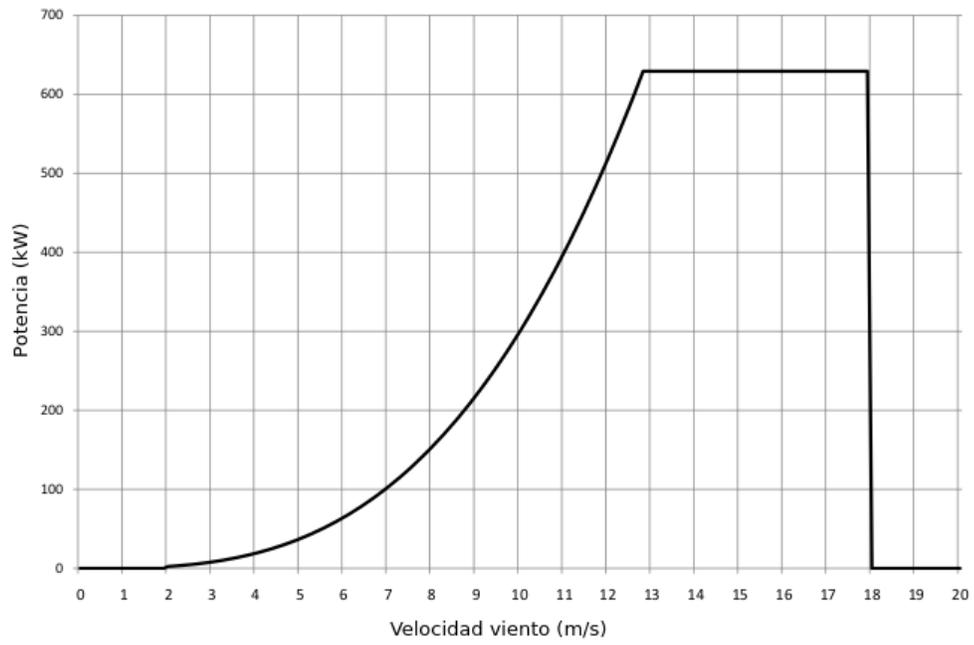


Figura 37. *Curva potencia aerogenerador.* [17].

## E. Anexo V: Fichero entrada UPMPARK.

El fichero \*.dat que sirve de referencia para generar los datos que necesita el código UPMPARK es el mostrado en la figura 38:

```

ho
0
10.
0.3
1.E23
282.6
10.
0.8
1.0
1.0
.5
40
10
-1
Sec., U, V, W, P, T, K, E, VI
1,1,0,0,0,0,1,0,0
26
100.
1000.
60.
Vestas60.tur
300.
1000.
60.
Vestas60.tur
800.
1000.
60.
Vestas60.tur
1000.
1000.
60.
Vestas60.tur
200.
900.
60.
Vestas60.tur
1000.
100.
60.
Vestas60.tur
60.
1
90.
12.
0
2.0
0
1
10.
16
4
0.5
1
65.
2.657
0.0016
1.
200
1

```

OROGRAPHY FILE	
PL	7 ec.(1) or 3 ec.(0)
DX	Grid size in X direction (m)
Z0	Surface roughness (m)
L	Monin-Obukhov length (m)
TS	Ground temperature (K)
NDIAM	Size of boundary conditions
CTMAX	Maximum CT
ERRM	Error in Poisson eq. iter. (Pa)
ERRORP	Error in pressure distribution (Pa)
RELAJ	Relaxation parameter for pressure
ITM	Maximum number of iterations
NSL	Number of sections behind the last turbine
NS	Number of sections to save results
S	Number of turbines (X=East, Y=North)
X TURBINE 1	(Coordinates in m)
Y	
Z	
TURBINE FILE	
X TURBINE 2	(Coordinates in m)
Y	
Z	
TURBINE FILE	
X TURBINE 3	(Coordinates in m)
Y	
Z	
TURBINE FILE	
X TURBINE 4	(Coordinates in m)
Y	
Z	
TURBINE FILE	
X TURBINE 5	(Coordinates in m)
Y	
Z	
TURBINE FILE	
X TURBINE 26	(Coordinates in m)
Y	
Z	
TURBINE FILE	
HM	Mast height (m)
ND	Number of cases (X=East, Y=North)
DI(1)	Wind direction(referred:clockwise from X)
VH(1)	Incident speed measured in mast (m/s)
NTS	number of points to analyse spectra
TDWA	Lenght attenuation (number of turbine diameters)
CASO	Meandering yes (1) no (0)
KIND	Meandering type sinusoidal (1) or spectrum (2)
TETA0	Meandering angle (\BA) (KIND=1)
NANG	Number of subangles (KIND=1)
PER	Number of time air cross the windpark
TID:	Time interval definition dt=TID*DIAMETER/VELOCITY (KIND=2)
ST:	Spectrum type (1=Kaimal, 2=Von Karman) (KIND=2)
LSX:	Integral Scale Paramenter x direction (m) (KIND=2)
VA	Variance x direction (m2/s2) (KIND=2)
LF	Lower frequency (Hz) (KIND=2)
UF	Upper frequency (Hz) (KIND=2)
NF	Number of frequency (KIND=2)
ponde:	Average type (0=last air cross , 1=all cases)

Figura 38. Fichero mosetti.dat para UPMPARK.

Señalar que no se han mostrado todas las líneas de las aeroturbinas por cuestión de espacio.