

Trabajo Fin de Máster  
Máster en Ingeniería Industrial

Estimación de parámetros de células fotovoltaicas  
mediante algoritmos de optimización.

Autor: Fernando Fuentes Sevillano

Tutores: Juan Manuel Roldán Fernández

Miguel Ángel González Cagigal

Escuela Técnica Superior de Ingeniería  
Universidad de Sevilla

Sevilla, 2020





Trabajo Fin de Máster  
Ingeniería Industrial

# **Estimación de parámetros de células fotovoltaicas mediante algoritmos de optimización.**

Autor:

Fernando Fuentes Sevillano

Tutores:

Juan Manuel Roldán Fernández

Miguel Ángel González Cagigal

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2020



Trabajo Fin de Máster: Estimación de parámetros de células fotovoltaicas mediante algoritmos de optimización.

Autor: Fernando Fuentes Sevillano

Tutores: Juan Manuel Roldán Fernández  
Miguel Ángel González Cagigal

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2020

El Secretario del Tribunal

*The good news about computers is that they do what you tell them to do.  
The bad news is that they do what you tell them to do.*

*-Ted Nelson -*







# Agradecimientos

---

En primer lugar, quisiera agradecer la ayuda de mis tutores Juan Manuel Roldán Fernández y Miguel Ángel González Cagigal, por darme la oportunidad de realizar este trabajo y los cuales me han guiado durante todo el desarrollo del mismo, ayudándome a descubrir la metaheurística y las maravillas de aplicar las mecánicas de procesos naturales a problemas de ingeniería.

A mi familia, principalmente a mis padres Fernando y M<sup>a</sup> Carmen y a mi hermana Belén, por haber estado junto a mi durante toda mi vida, por enseñarme aquello que no se puede enseñar en los colegios y a ayudarme a crecer como persona, sin ellos nada de esto habría sido posible. Un recuerdo especial para los que ya no están.

A todos los profesores que he tenido a lo largo de los años, de los cuales siempre ha quedado algo.

Una mención para mis amigos, con los que tanto he compartido, tanto semanas de intenso estudio como viajes y fines de semana inolvidables.

Gracias también a mis compañeros de piso, los cuales me han hecho sentir como en casa durante estos difíciles meses de privación de libertad debido al COVID-19, a pesar de estar a kilómetros de ella.

Por último, pero no menos importantes, a todos aquellos ingenieros e investigadores que dedican sus vidas al desarrollo de la ciencia, es un orgullo pertenecer a este gremio.

*Fernando Fuentes Sevillano*

*Sevilla, 2020*



# Resumen

---

Las células fotovoltaicas son habitualmente modeladas como circuitos. Encontrar los parámetros adecuados del modelo es muy importante para el seguimiento del punto de máxima potencia y maximizar la energía generada en el panel fotovoltaico. El objetivo de este estudio es encontrar la mejor manera de obtener los parámetros de los modelos de diodo simple y diodo doble, así como cuál de los dos modelos es mejor en cuanto a precisión. Para ello se han probado cuatro algoritmos diferentes en MATLAB suponiendo dos escenarios: que se dispone de datos experimentales en distintos puntos de la curva I-V, o que solo se tienen los tres puntos habituales suministrados en el catálogo. Finalmente se obtienen unas conclusiones a partir de los resultados obtenidos en ambos escenarios.



# Abstract

---

Photovoltaic cells are usually modelled as circuits. Finding proper model parameters is very important for power point tracking and maximum energy generation. The purpose of this study is determining the best way to obtain the parameters of single and double diode models, as well as which of both models has better accuracy. In order to accomplish this task four different algorithms have been tested with MATLAB in two different situations: in one case there are several experimental points of the I-V characteristic; in the other case there are not, so datasheet points are used. Finally, the results obtained of both situations are discussed.

<b>Agradecimientos</b>	<b>x</b>
<b>Resumen</b>	<b>xii</b>
<b>Abstract</b>	<b>xiv</b>
<b>Índice</b>	<b>xv</b>
<b>Índice de Tablas</b>	<b>xvii</b>
<b>Índice de Figuras</b>	<b>xix</b>
<b>Notación</b>	<b>xxii</b>
<b>1 Introducción. Objetivos del estudio.</b>	<b>1</b>
1.1. <i>Objetivos y desarrollo del estudio.</i>	2
<b>2 Modelos de células fotovoltaicas</b>	<b>5</b>
2.1. <i>Modelo de célula fotovoltaica ideal.</i>	5
2.2. <i>Modelo de un diodo con resistencia en serie.</i>	6
2.3. <i>Modelo de un diodo con resistencia en paralelo.</i>	7
2.4. <i>Modelo de dos diodos.</i>	7
2.5. <i>Otros modelos.</i>	8
<b>3 PSO, optimización por enjambre de partículas.</b>	<b>9</b>
3.1. <i>Funcionamiento del algoritmo.</i>	9
3.2. <i>Aplicación del algoritmo PSO en MATLAB al problema.</i>	10
3.2.1 <i>Modelo de diodo simple.</i>	10
3.2.2 <i>Modelo de diodo doble.</i>	13
3.2.3 <i>Observaciones.</i>	15
<b>4 Algoritmo genético (GA)</b>	<b>17</b>
4.1. <i>Funcionamiento de un algoritmo genético.</i>	17
4.2. <i>Creación de cada generación.</i>	18
4.2.1 <i>Élites</i>	18
4.2.2 <i>Cruce</i>	18
4.2.3 <i>Mutación</i>	20
4.3. <i>Aplicación del algoritmo genético en MATLAB al problema.</i>	20
4.3.1 <i>Modelo de diodo simple dependiente de tres parámetros.</i>	20
4.3.2 <i>Modelo de diodo simple dependiente de cinco parámetros.</i>	23
4.3.3 <i>Modelo de diodo doble.</i>	26
4.4. <i>Conclusiones de la aplicación del algoritmo genético al problema.</i>	29
<b>5 Algoritmo pattern search (PS)</b>	<b>31</b>
5.1. <i>Funcionamiento del algoritmo.</i>	31
5.2. <i>Pattern search en MATLAB.</i>	32
5.2.1 <i>Modelo de diodo simple.</i>	32
5.2.2 <i>Modelo de diodo doble.</i>	34
5.3. <i>Conclusiones de la aplicación del algoritmo pattern search al problema.</i>	35
5.4. <i>Pattern search con solución inicial de GA.</i>	36

<b>6</b>	<b>Simulated annealing (SA)</b>	<b>39</b>
6.1.	<i>Descripción del algoritmo.</i>	39
6.2.	<i>Aplicación del algoritmo simulated annealing al problema en MATLAB.</i>	40
6.2.1	Modelo de diodo simple.	40
6.2.2	Modelo de diodo doble.	42
6.3.	<i>Conclusiones de la aplicación del algoritmo SA al problema.</i>	44
<b>7</b>	<b>Comparación de los algoritmos PSO, GA, SA y PS.</b>	<b>45</b>
<b>8</b>	<b>Estudio de los algoritmos con disponibilidad de datos experimentales</b>	<b>49</b>
8.1.	<i>Artículo de Easwarakhanthan, datos experimentales.</i>	49
8.2.	<i>Resultados de los algoritmos para el modelo de diodo simple.</i>	51
8.3.	<i>Resultados de los algoritmos para el modelo de diodo doble.</i>	55
8.4.	<i>Conclusiones y observaciones del estudio.</i>	59
<b>9</b>	<b>Conclusiones del estudio en ambos escenarios</b>	<b>63</b>
	<b>Referencias</b>	<b>65</b>
	<b>Glosario</b>	<b>66</b>



# ÍNDICE DE TABLAS

---

Tabla 3-1. Errores PSO diodo simple.	11
Tabla 3-2. Parámetros PSO diodo simple	11
Tabla 3-3. Errores de PSO para diodo doble.	14
Tabla 3-4. Parámetros PSO para el modelo de diodo doble.	14
Tabla 4-1. Errores GA, 3 parámetros independientes.	21
Tabla 4-2. Parámetros GA, 3 parámetros independientes.	21
Tabla 4-3. Errores GA, 5 parámetros independientes.	24
Tabla 4-4. Parámetros GA, 5 parámetros independientes.	24
Tabla 4-5. Errores GA, diodo doble.	27
Tabla 4-6. Parámetros GA, diodo doble.	27
Tabla 5-1. Errores PS diodo simple.	33
Tabla 5-2. Parámetros diodo simple.	33
Tabla 5-3. Errores PS diodo doble.	35
Tabla 5-4. Parámetros PS diodo doble.	35
Tabla 6-1. Errores SA diodo simple.	41
Tabla 6-2. Parámetros SA diodo simple.	41
Tabla 6-3. Errores SA diodo doble.	43
Tabla 6-4. Parámetros SA diodo doble.	43
Tabla 7-1. Errores en % por célula, algoritmo y modelo.	45
Tabla 7-2. Tiempos de ejecución (s).	47
Tabla 8-1. Parámetros del modelo de Easwarakhanthan.	50
Tabla 8-2. Tensiones e intensidades de Easwarakhanthan y a partir de su modelo.	50
Tabla 8-3. Comparación de errores.	51
Tabla 8-4. Parámetros modelo diodo simple.	52
Tabla 8-5. Errores por punto modelo diodo simple.	53
Tabla 8-6. MAE y RMSE de los algoritmos para el modelo de diodo simple.	53
Tabla 8-7. Parámetros modelo diodo doble.	56
Tabla 8-8. Errores modelo diodo doble.	56
Tabla 8-9. MAE y RMSE de los algoritmos para el modelo de diodo doble.	57



# ÍNDICE DE FIGURAS

---

Figura 1.1. Evolución del coste de generación de la energía solar. [1]	1
Figura 1.2. Evolución global de la capacidad instalada anual 2000-2018. [1]	2
Figura 2.1. Superposición de $I_D$ e $I_{PV}$ . [1]	5
Figura 2.2. Puntos característicos de la curva I-V.	6
Figura 2.3. Circuito célula ideal FV. [4]	6
Figura 2.4. Circuito modelo $R_s$ . [4]	7
Figura 2.5. Circuito modelo $R_p$ . [4]	7
Figura 2.6. Circuito con dos diodos. [4]	8
Figura 3.1. La gran migración del Serengueti. [6]	9
Figura 3.2. Actualización de la posición de un elemento en PSO. [7]	10
Figura 3.3. Convergencia de PSO para el modelo de diodo simple.	12
Figura 3.4. Curva I-V de SPR-X22-370	12
Figura 3.5. Curva I-V de IBS60P 265W.	13
Figura 3.6. Convergencia de PSO para el modelo de diodo doble.	15
Figura 3.7. Curva I-V de la célula IBS60P 265W	15
Figura 3.8. Curva I-V PSO de célula <i>thin film</i> (diodo doble).	16
Figura 4.1. Charles Darwin.	17
Figura 4.2. Funcionamiento de un algoritmo genético. [8]	18
Figura 4.3. Convergencia del GA de 3 parámetros independientes.	22
Figura 4.4. Curva I-V de la célula CHSM5001T-100. (GA3)	22
Figura 4.5. Curva I-V de la célula Sanyo HIT 215. (GA3)	23
Figura 4.6. Convergencia del GA de 5 parámetros independientes.	25
Figura 4.7. Curva I-V de la célula SPR-X22-370. (GA5)	25
Figura 4.8. Curva I-V de la célula CHSM5001T-100. (GA5)	26
Figura 4.9. Evolución algoritmo GA diodo doble, vista general.	28
Figura 4.10. Evolución algoritmo GA diodo doble, detalle.	28
Figura 4.11. Evolución algoritmo GA diodo doble, mala convergencia.	29
Figura 4.12. Curva I-V de la célula IBS60P 265W. (GA7)	29
Figura 4.13. Comparación de modelos del GA.	30
Figura 5.1. Funcionamiento del PS. [11]	32
Figura 5.2. Evolución del PS, diodo simple.	33
Figura 5.3. Curva I-V de la célula KC200GT, PS diodo simple.	34
Figura 5.4. Curva I-V de la célula SPR-X22-370, PS diodo simple.	34
Figura 5.5. Comparación algoritmo PS para los dos modelos.	36

Figura 5.6. Evolución del error del PS al tomar como $x_0$ la solución del GA, célula SPR-X22-370.	37
Figura 5.7. Evolución del error del PS al tomar como $x_0$ la solución del GA, célula IBS60P-265W.	38
Figura 6.1. Recocido de acero. [12]	39
Figura 6.2. Evolución del SA.	41
Figura 6.3. Curva I-V IBS60P 265W. (SA5)	42
Figura 6.4. Curva I-V SPR-X22-370. (SA5)	42
Figura 6.5. Curva I-V de la célula Sanyo HIT 215. (SA7)	44
Figura 7.1. Errores en % por célula, algoritmo y modelo.	46
Figura 7.2. Media de errores por algoritmo y modelo.	47
Figura 7.3. Media de tiempos de los algoritmos.	48
Figura 8.1. Curva I-V GA	54
Figura 8.2. Curva I-V PSO	54
Figura 8.3. Curva I-V PS	55
Figura 8.4. Curva I-V SA.	55
Figura 8.5. Curva I-V GA diodo doble.	57
Figura 8.6. Curva I-V PSO diodo doble.	58
Figura 8.7. Curva I-V PS diodo doble.	58
Figura 8.8. Curva I-V SA diodo doble.	59
Figura 8.9. Comparativa MAE.	60
Figura 8.10. Comparativa RMSE.	60
Figura 8.11. Perfil de errores, PSO.	61
Figura 8.12. Perfil de errores, PS.	61
Figura 8.13. Ruido en los datos experimentales, GA diodo simple.	62
Figura 8.14. Tiempos de ejecución.	62



# Notación

---

$\Sigma$	Sumatorio
$\epsilon$	Relación de pertenencia
$\Theta$	Vector de parámetros del modelo de diodo simple
En	$\cdot 10^n$
exp(n)	Función exponencial $e^n$
log	Logaritmo en base 10
rand	Función aleatorio







# 1 INTRODUCCIÓN. OBJETIVOS DEL ESTUDIO.

En la generación de energía eléctrica, la utilización de energías renovables está experimentando un crecimiento exponencial. Entre las fuentes de energía renovables, la solar fotovoltaica representa la fuente de energía con un crecimiento más rápido, debido en gran medida al decremento de los costes de generación, como se aprecia en el histórico de la Figura 1.1. En la actualidad es la tercera fuente renovable más empleada del mundo.

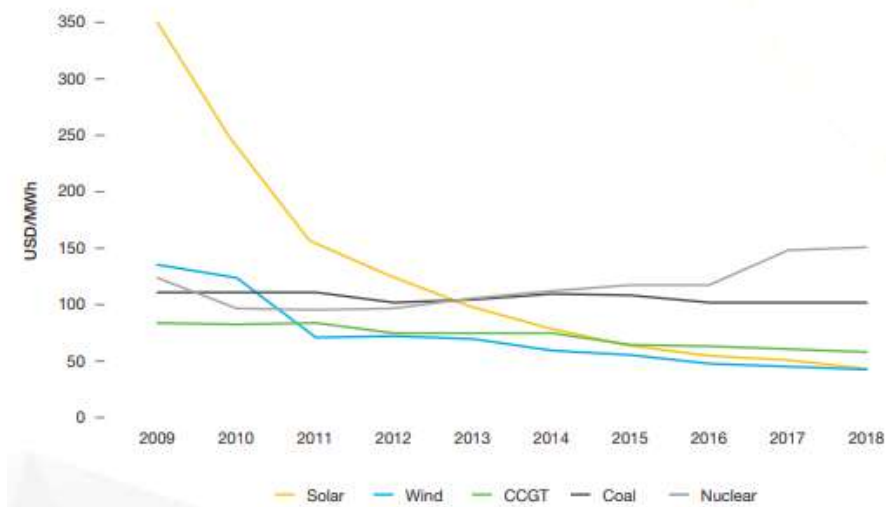


Figura 1.1. Evolución del coste de generación de la energía solar. [1]

Los motivos que han llevado al reemplazo de la energía convencional por solar fotovoltaica son los siguientes, recopilados por A.R. Jordehi [1]:

- La fuente empleada por los paneles fotovoltaicos, el sol, es virtualmente inagotable, mientras que la cantidad de combustibles fósiles es limitada.
- Durante la etapa de generación del ciclo de vida de un panel fotovoltaico no se genera contaminación, a diferencia de la generación con combustibles fósiles. Sin embargo, no hay que olvidar que durante la fabricación de paneles sí que se genera CO<sub>2</sub>.
- Los combustibles fósiles contribuyen al calentamiento global en mayor medida que la energía solar fotovoltaica.
- Los costes de mantenimiento y operación de la energía solar fotovoltaica son reducidos.
- Si comparamos las energías renovables, la energía solar fotovoltaica tiene la mayor densidad de energía.

Estas ventajas hacen que se instalen cada vez más paneles solares en todas sus variantes, integrados en tejados y paredes o en el suelo. Su escalabilidad permite la instalación tanto de grandes plantas centralizadas como de pequeñas unidades de generación distribuida. Además, pueden conectarse a la red eléctrica o pueden utilizarse de forma aislada. La asociación de energía solar Solar Power Europe recoge en su informe periódico el crecimiento año tras año del mercado solar fotovoltaico, elaborando gráficas como la de la Figura 1.2, donde se aprecia el aumento ininterrumpido de la capacidad instalada.

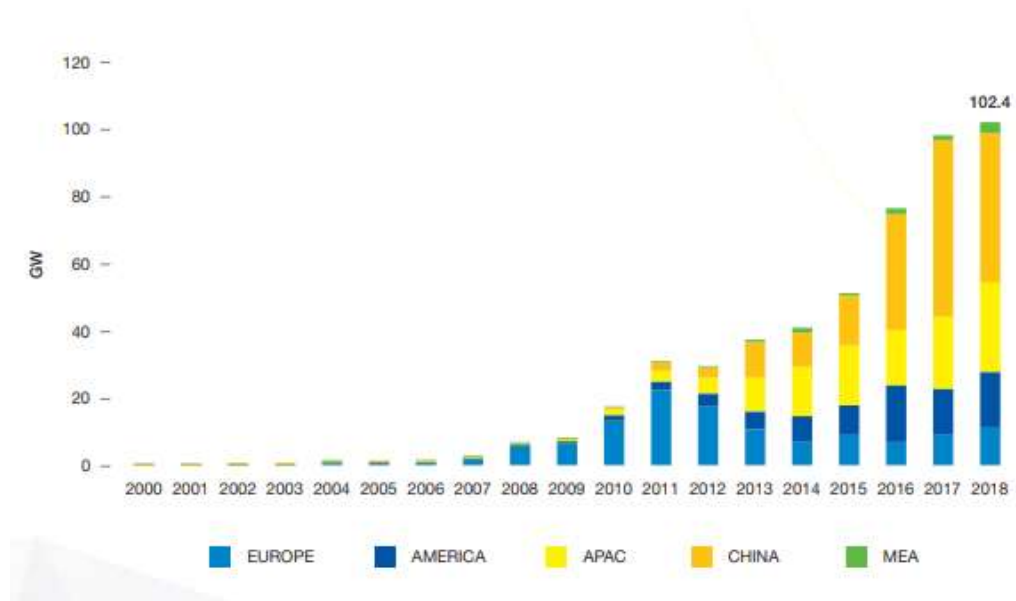


Figura 1.2. Evolución global de la capacidad instalada anual 2000-2018. [1]

La célula fotovoltaica es el componente básico de un sistema de generación fotovoltaica. Es básicamente una unión P-N expuesta a la luz solar, intentando maximizar la absorción de fotones y minimizar la resistencia de los contactos. Varias células fotovoltaicas se conectan para formar un panel fotovoltaico, y varios paneles se conectan a un inversor para convertir la corriente continua en corriente alterna, tras lo que se inyecta en la red o se emplea directamente para autoconsumo. Es por ello que este estudio se ha centrado en encontrar un modelo de este componente básico, del que depende lo que ocurre aguas abajo y que hace posible todo un sector en auge.

## 1.1. Objetivos y desarrollo del estudio.

Un modelado apropiado de las células fotovoltaicas es fundamental para el diseño, la evaluación, la simulación, el control y la optimización de sistemas fotovoltaicos. También es crucial para minimizar el coste computacional del seguimiento del punto de máxima potencia y alcanzar este punto de forma óptima. El modelado incluye un circuito equivalente adecuado, el cual no es suministrado por el fabricante, y unos parámetros del circuito equivalente correctos, lo que constituye un desafío. Esto es debido a que la curva I-V de la célula fotovoltaica es no lineal, lo que da lugar a un problema de optimización no lineal.

Según A.R. Jordehi [2], un método para estimar los parámetros de un modelo de célula fotovoltaica debe tener las siguientes características:

- Debe ser preciso, ajustándose todo lo posible a los puntos de la característica I-V dada por las especificaciones del fabricante o los datos experimentales.
- Debe dar parámetros adecuados para múltiples y diversos sets de datos.
- Debe tener un bajo tiempo de computación, especialmente si se aplica al seguimiento de puntos de máxima potencia.

Sobre este tema se han llevado a cabo numerosas investigaciones, en las que cada autor defiende unos métodos frente a otros, por lo que no hay un consenso claro sobre qué algoritmo es el mejor. El objetivo de este trabajo es analizar de forma imparcial cuatro de los algoritmos utilizados (algoritmo genético o GA, algoritmo de enjambre de partículas o PSO, *pattern search* o PS y *simulated annealing* o SA). Para ello se ha utilizado la *Optimization Toolbox* que ofrece MATLAB, y que incluye estos cuatro algoritmos.

En primer lugar, se ha estudiado el comportamiento de los algoritmos con los datos de las fichas técnicas de cinco placas solares, estos datos se limitan por regla general a tres puntos de la curva (cortocircuito, máxima potencia y circuito abierto).

En segundo lugar, se ha analizado el funcionamiento cuando se dispone de un mayor número de datos

experimentales, empleando los datos del estudio de Easwarakhanthan [3]. Estos datos son los empleados por numerosos autores para probar sus algoritmos, ya que, aunque es un estudio de 1986, apenas hay datos experimentales públicos.

La comparación se ha enfocado fundamentalmente en la precisión, ya que los tiempos de computación son de órdenes similares y dependen en cierta medida de la precisión deseada, aunque estos últimos también se han recogido y comentado.



## 2 MODELOS DE CÉLULAS FOTOVOLTAICAS

Existen diferentes propuestas de circuitos eléctricos más o menos complejos para modelar el comportamiento de una célula solar fotovoltaica [1]. Los autores de ensayos que estudian algoritmos de obtención de parámetros para sus modelos emplean distintos circuitos según el que consideren más apropiado en su caso. A continuación se revisarán y analizarán los más empleados según su complejidad.

### 2.1. Modelo de célula fotovoltaica ideal.

En ausencia de radiación solar la célula fotovoltaica se comporta como un diodo de unión PN, el cual se modela habitualmente con la ecuación de Shockley (1.1)

$$I_D = I_0 \left[ \exp\left(\frac{qV_D}{aKT}\right) - 1 \right] \quad (1.1)$$

donde  $I_D$  y  $V_D$  son la intensidad y la tensión del diodo,  $I_0$  la corriente de saturación,  $q$  la carga del electrón,  $K$  la constante de Boltzmann y  $T$  la temperatura en Kelvin.

Si hay radiación solar incidente en la unión PN, los fotones crean pares electrón-hueco y se produce una corriente aprovechable por un circuito exterior. Esta corriente se representa en el modelo mediante una fuente de intensidad de módulo  $I_{PV}$  en paralelo con el diodo de la unión PN. La suma de las corrientes de estos dos elementos da como resultado la característica I-V de la célula (1.2), cuya curva es la superposición de  $I_D$  e  $I_{PV}$  como se aprecia en la Figura 2.1.

$$I = I_{PV} - I_0 \left[ \exp\left(\frac{qV_D}{aKT}\right) - 1 \right] \quad (1.2)$$

Este modelo tiene tres parámetros de los que dependerá la curva:  $I_{PV}$ ,  $I_0$  y  $a$ .

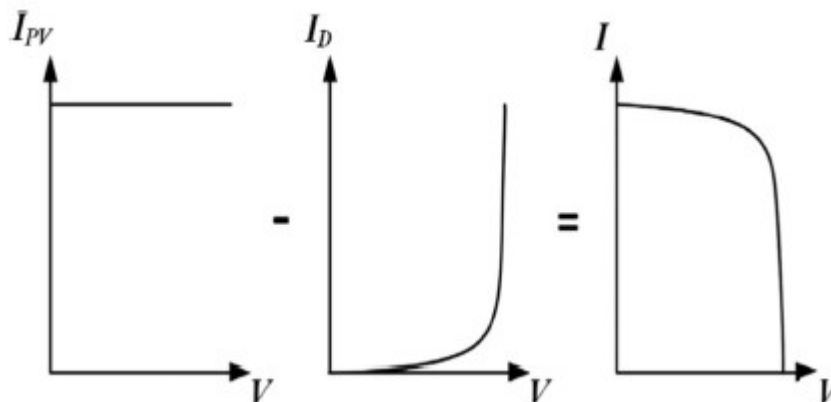


Figura 2.1. Superposición de  $I_D$  e  $I_{PV}$ . [1]

La curva I-V tiene la misma forma para todos los modelos y existen tres puntos relevantes que suelen estar presentes en el catálogo de cualquier fabricante. Los puntos en los que la curva corta los ejes de abscisas y ordenadas son dos de ellos, en el que la tensión es igual a cero es el punto de cortocircuito y el que tiene intensidad cero es conocido como punto de circuito abierto. El tercer punto de interés es el punto de máxima potencia, el

cual está representado entre los otros dos en el ejemplo de la Figura 2.2

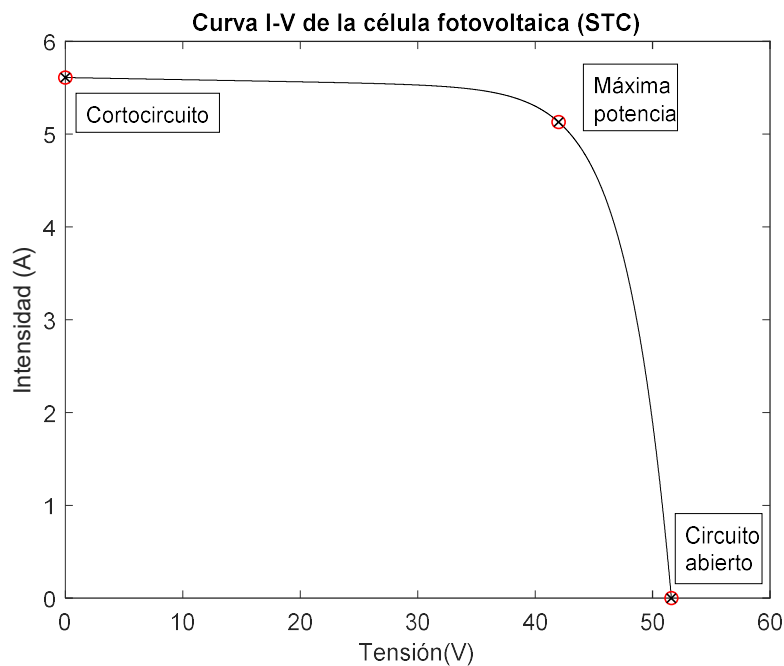


Figura 2.2. Puntos característicos de la curva I-V.

El circuito equivalente del modelo de célula fotovoltaica ideal está representado en la Figura 2.3.

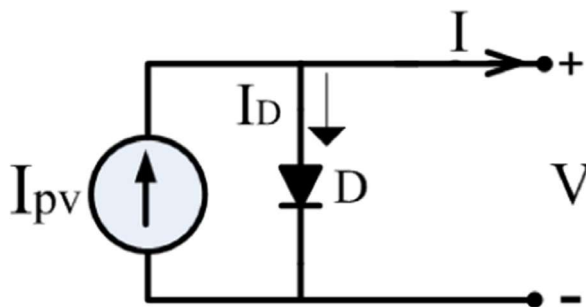
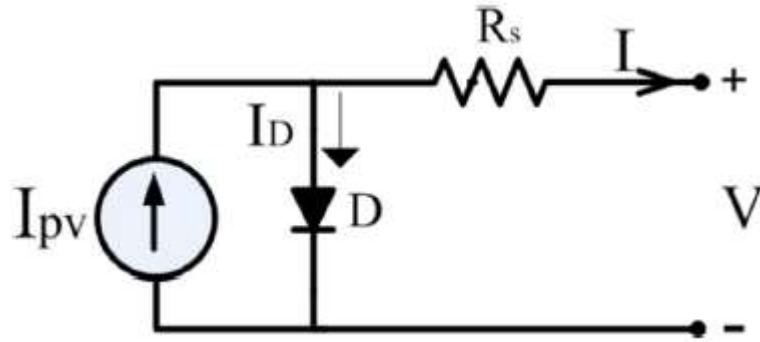


Figura 2.3. Circuito célula ideal FV. [4]

## 2.2. Modelo de un diodo con resistencia en serie.

El modelo ideal no se emplea en simulaciones, siendo utilizado meramente para entender teóricamente el funcionamiento de la célula. Añadiendo al anterior modelo una resistencia en serie que represente la resistencia del contacto entre el silicio y el electrodo se consigue un comportamiento más acorde a la realidad. Al añadir la resistencia se introduce un nuevo parámetro a los tres del modelo ideal, siendo la curva ahora dependiente de  $I_{PV}$ ,  $I_0$ ,  $a$  y  $R_S$ . La curva I-V está definida por la expresión (1.3) y el circuito equivalente es el de la Figura 2.4.

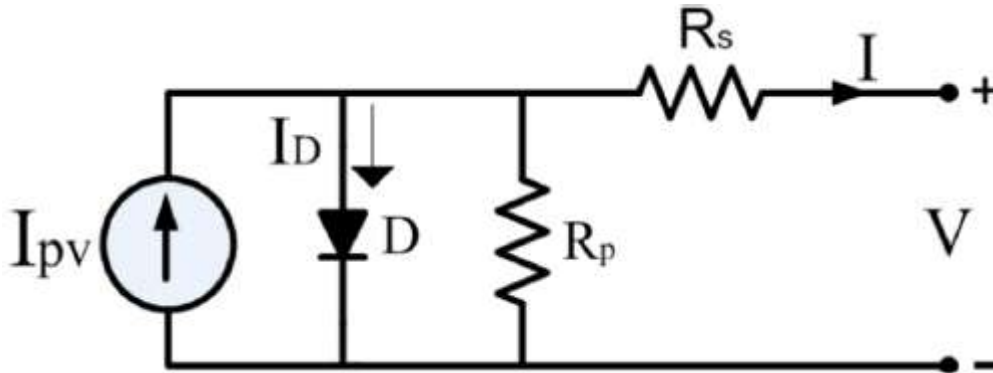
$$I = I_{PV} - I_0 \left[ \exp\left(\frac{q(V + R_S I)}{aKT}\right) - 1 \right] \quad (1.3)$$

Figura 2.4. Circuito modelo  $R_s$ . [4]

### 2.3. Modelo de un diodo con resistencia en paralelo.

El modelo se puede seguir complicando teniendo en cuenta la resistencia de la unión PN, lo que se representa en el circuito como una resistencia en paralelo al diodo y a la fuente de intensidad, como se ha plasmado en la Figura 2.5. Este es el modelo más empleado debido a una simplicidad media con un comportamiento notablemente fiel a la realidad. Los parámetros de los que depende son cinco:  $I_{pv}$ ,  $I_0$ ,  $a$ ,  $R_s$  y  $R_p$ . La característica I-V viene dada por la expresión (1.4).

$$I = I_{pv} - I_0 \left[ \exp\left(\frac{q(V + R_s I)}{aKT}\right) - 1 \right] - \frac{V + R_s I}{R_p} \quad (1.4)$$

Figura 2.5. Circuito modelo  $R_p$ . [4]

### 2.4. Modelo de dos diodos.

Aunque el modelo anterior es bastante completo, no se comporta bien en situaciones de irradiancia baja, lo que se soluciona añadiendo un segundo diodo (Figura 2.6). A pesar de la mayor precisión alcanzada con este modelo, la complejidad aumenta de forma considerable al añadir dos nuevos parámetros relativos a este segundo diodo,  $I_{02}$  y  $a_2$ , a los cinco parámetros del caso de un diodo:  $I_{pv}$ ,  $I_0$ ,  $a$ ,  $R_s$  y  $R_p$ . La expresión de la curva I-V es la (1.5).

$$I = I_{pv} - I_{01} \left[ \exp\left(\frac{q(V + R_s I)}{a_1 KT}\right) - 1 \right] - I_{02} \left[ \exp\left(\frac{q(V + R_s I)}{a_2 KT}\right) - 1 \right] - \frac{V + R_s I}{R_p} \quad (1.5)$$

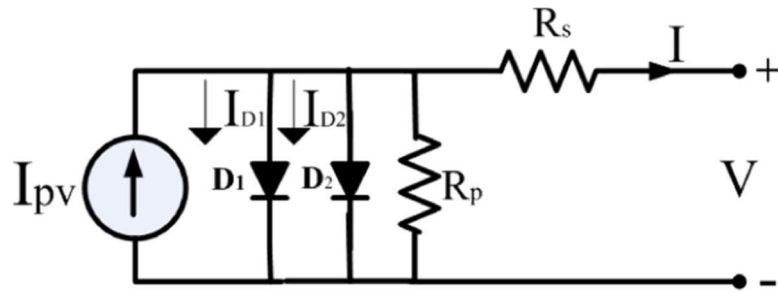


Figura 2.6. Circuito con dos diodos. [4]

## 2.5. Otros modelos.

Existen otros modelos que han sido empleados por algunos autores, aunque son menos frecuentes debido a su complejidad. Algunos ejemplos son el de triple diodo o el de diodo único con condensador.



# 3 PSO, OPTIMIZACIÓN POR ENJAMBRE DE PARTICULAS.

---

*Lo que no beneficia al enjambre, tampoco beneficia a la abeja.*

*- Marco Aurelio -*

El algoritmo PSO, Particle Swarm Optimization, u optimización por enjambre de partículas es un algoritmo de optimización inspirado en el comportamiento de animales organizados en sociedades en la naturaleza. Ejemplos de estas sociedades son bandadas de pájaros, manadas de animales como la de la Figura 3.1, bancos de peces o enjambres de insectos. La autoría de este método se atribuye a J. Kennedy y R. Eberhart en 1995 [5].



Figura 3.1. La gran migración del Serengueti. [6]

El objetivo del algoritmo es como en todos los algoritmos de optimización minimizar una función objetivo alterando una serie de variables. Para ello, se coloca en el espacio vectorial de las dimensiones necesarias una serie de individuos (partículas del enjambre), que pueden moverse libremente por el espacio comunicándose entre ellos para compartir información.

## 3.1. Funcionamiento del algoritmo.

Al comienzo del algoritmo se asigna al número de partículas definido posiciones y velocidades aleatorias. Para actualizar las posiciones del enjambre se utiliza la expresión (2.1), en la que se suma la posición a la velocidad para definir la siguiente posición.

$$x_i(t+1) = x_i(t) + v_i(t) \quad (2.1)$$

La velocidad se calcula con la expresión (2.2), un sumatorio de tres términos. El primer término es el asociado a la inercia de la partícula, escrito como el producto de la velocidad anterior por  $w_i$ , que depende del comportamiento de la función objetivo. El segundo y el tercer término son la atracción al mínimo encontrado por la partícula y el encontrado por la totalidad del enjambre respectivamente. Estas atracciones son expresadas como la distancia a las posiciones de los mínimos multiplicadas por constantes de atracción ( $y_1$  e  $y_2$ ), que se definirán de una forma u otra para establecer un comportamiento más social o más individual, y por números aleatorios entre 0 y 1 ( $u_1$  y  $u_2$ ).

$$v_i(t+1) = w_i v_i(t) + y_1 u_1 (p_i - x_i(t)) + y_2 u_2 (g_i - x_i(t)) \quad (2.2)$$

De esta manera, la nueva posición es el resultado de la suma de la inercia, la memoria del individuo que lo atrae al mínimo visitado y la cooperación del enjambre que atrae a la partícula al mínimo encontrado por la totalidad de la comunidad, como se aprecia en la Figura 3.2.

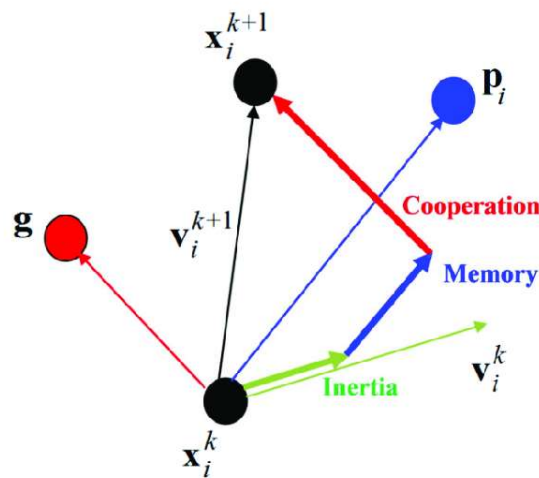


Figura 3.2. Actualización de la posición de un elemento en PSO. [7]

### 3.2. Aplicación del algoritmo PSO en MATLAB al problema.

Con el propósito de resolver el problema de este estudio se ha elegido como función objetivo a minimizar el sumatorio de los errores en intensidad para los puntos definidos en el catálogo (cortocircuito, circuito abierto y máxima potencia). Para ello se calcula la intensidad dada por los parámetros de cada individuo del enjambre en los tres puntos de catálogo para las tensiones en estos puntos y se comparan entre sí. La expresión de la curva I-V y el número de parámetros depende del modelo utilizado, en este capítulo se ha estudiado el comportamiento del algoritmo con los modelos de diodo simple y diodo doble.

#### 3.2.1 Modelo de diodo simple.

Como ya se ha desarrollado en el capítulo de modelos de células fotovoltaicas, para este modelo existen cinco parámetros de los que depende la curva:  $I_{PV}$ ,  $I_0$ ,  $a$ ,  $R_S$  y  $R_P$ . Los límites entre los que se pueden mover estos parámetros se han definido en línea con los valores que se suelen emplear en el estudio de estos modelos y son los siguientes:

$$a \in [1, 2] \quad R_S \in [0.01, 1.2] \quad R_P \in [50, 1000]$$

$$I_0 \in [1e-12, 1e-5] \quad I_{PV} \in [1, 9]$$

Las constantes de atracción  $y_1$  e  $y_2$  son 1.49 ambas, las que vienen por defecto en MATLAB. El rango en el que

puede estar el factor de inercia que escala la velocidad de la iteración anterior es  $[0.1, 1.1]$ . El enjambre está formado por 100 miembros inicialmente distribuidos por el espacio de manera uniforme. El algoritmo se detiene al alcanzar el número máximo de iteraciones, establecido en 1000, o un cambio relativo de la función objetivo en las últimas 20 iteraciones menor a la tolerancia, establecida en  $1e-16$ . El proceso se ejecuta diez veces mediante un bucle *for*, guardando la mejor solución de todas las obtenidas.

Finalmente se imprime por pantalla el error de la mejor solución, los parámetros de la curva, las intensidades en los puntos del catálogo y el tiempo de CPU empleado en ejecutar el algoritmo diez veces. Además, se representa gráficamente la curva I-V con puntos hallados mediante el comando *fsolve* de MATLAB y la evolución del mínimo encontrado en cada iteración por el algoritmo.

Se ha tomado un total de cinco placas fotovoltaicas para probar el comportamiento del algoritmo. Las placas elegidas son dos policristalinas, la Sanyo HIT 215 y la IBS60P 265W; dos monocristalinas, la Kyocera KC200GT y la SPR-X22-370; y una de película fina, la CHSM5001T-100.

En la Tabla 3-1 se han recogido las intensidades de catálogo en los puntos de cortocircuito y de máxima potencia, y se han comparado con las intensidades del modelo, además de la intensidad del modelo en el punto de circuito abierto, que debe ser cero. También se ha calculado el error medio en intensidad de estos tres puntos dividido entre la intensidad del punto de máxima potencia, para tener un valor comparable entre células de intensidades tan dispares como la Kyocera KC200GT (7.61 A) y la CHSM5001T-100 (1.188 A).

Tabla 3-1. Errores PSO diodo simple.

	Sanyo HIT 215	Kyocera KC200GT	SPR-X22- 370	IBS60P 265W	CHSM5001T- 100
Isc catálogo (A)	5.61	8.21	6.66	8.98	1.499
Impp catálogo (A)	5.13	7.61	6.26	8.58	1.188
Isc modelo (A)	5.61	8.21	6.74	9.00	1.499
Impp modelo (A)	5.13	7.61	6.07	8.39	1.188
Ioc modelo (A)	0	0	-2.68E-06	-7.37E-04	2.22E-16
Error medio en intensidad	0.00%	0.00%	1.49%	0.80%	0.00%
tiempo (s) 10 iter.	47	45	65	47	39

El ajuste a los puntos es perfecto para las dos primeras células y excelente en la última, errando por  $2.22e-16$  en el punto de circuito abierto. Para las otras dos células, aunque es un buen ajuste, es mejorable siendo el error 1.49% y 0.8%. En la Tabla 3-2 se ha reflejado el valor de los parámetros para estos errores, cabe destacar que estos parámetros tienen numerosos decimales y se ha redondeado para una mejor visualización.

Tabla 3-2. Parámetros PSO diodo simple

	Sanyo HIT 215	Kyocera KC200GT	SPR-X22-370	IBS60P 265W	CHSM5001T-100
a	1.97	1.73	1.76	1.72	1.89
Rs (ohmios)	0.052	0.010	0.010	0.011	0.016
Rp (ohmios)	822	424	998	995	271
I0 (A)	3.97E-06	9.16E-06	7.72E-07	4.97E-06	1.00E-12
Ipv (A)	5.6	8.2	6.7	9.0	1.5

El tiempo empleado en ejecutar el algoritmo se encuentra en torno a 50 segundos, lo que puede parecer excesivo a priori, sin embargo, hay que destacar que es el total de las 10 ejecuciones del PSO. Por otra parte, en el ejemplo de la Figura 3.3 se observa una rápida convergencia a valores del error próximos a cero, aunque tarda en eliminar los últimos decimales, por lo que se puede lograr una considerable precisión en un tiempo relativamente corto.

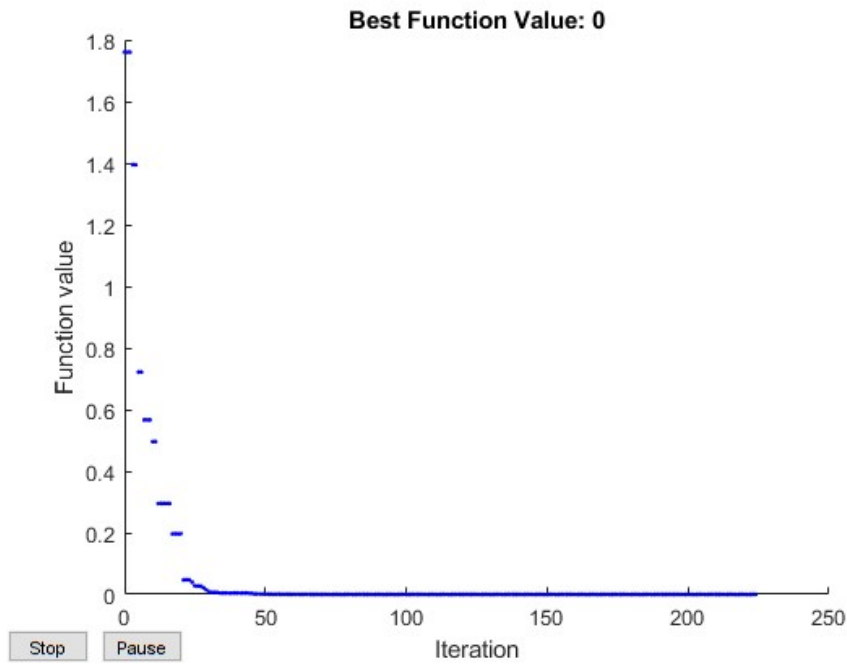


Figura 3.3. Convergencia de PSO para el modelo de diodo simple.

A continuación, se ha representado la curva I-V de las células con errores destacables, la SPR-X22-370 y la IBS60P 265W en la Figura 3.4 y la Figura 3.5 respectivamente. Los puntos de catálogo por los que debería pasar la curva han sido dibujados mediante círculos rojos, y los puntos de la curva con cruces negras. Se puede comprobar a simple vista que la forma de la curva es la habitual de una célula fotovoltaica y que pasa muy cerca de los puntos, ya que los errores eran relativamente pequeños.

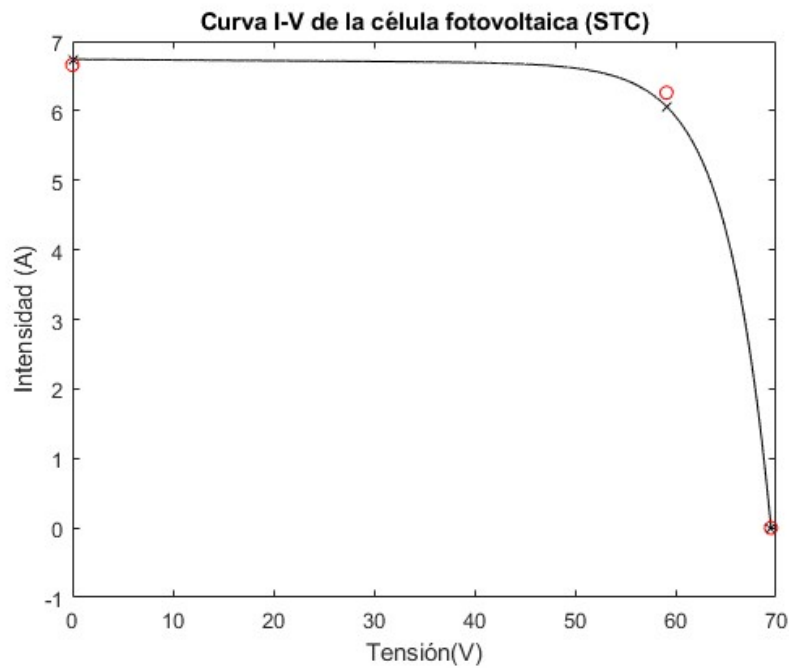


Figura 3.4. Curva I-V de SPR-X22-370

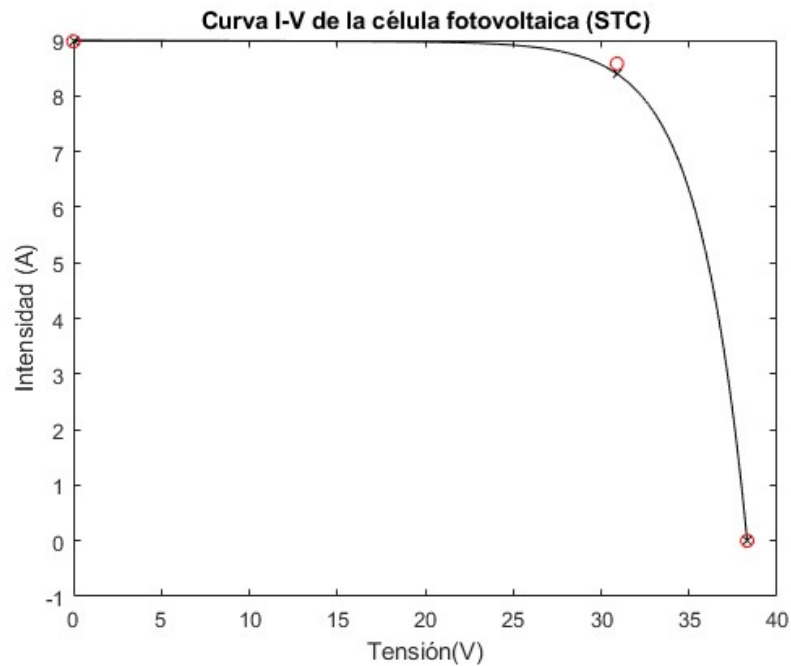


Figura 3.5. Curva I-V de IBS60P 265W.

### 3.2.2 Modelo de diodo doble.

En esta subsección se estudia el comportamiento del algoritmo con el modelo de diodo doble, menos utilizado que el de diodo simple debido a su complejidad a pesar de su mejor capacidad de ajuste a los puntos de la curva. Se comprobará si en este caso se mejoran los ya buenos resultados del modelo de diodo simple, y si aumenta el coste computacional.

A los parámetros del diodo simple del caso anterior, cuyos intervalos se han respetado, se añaden dos nuevos relativos al segundo diodo. Los límites de los valores que pueden tomar los parámetros son los siguientes:

$$\begin{array}{lll}
 a \in [1, 2] & R_S \in [0.01, 1.2] & R_P \in [50, 1000] \\
 I_0 \in [1e-12, 1e-5] & I_{PV} \in [1, 9] & a_2 \in [1, 2] & I_{02} \in [1e-12, 1e-5]
 \end{array}$$

Las constantes de atracción, así como el rango del factor de inercia se han mantenido los tomados por defecto en MATLAB debido a los buenos resultados obtenidos anteriormente. Los miembros del enjambre y los requisitos de detención del algoritmo son también los mismos.

En la Tabla 3-3 se ha tabulado las intensidades, el error y el tiempo al igual que en el caso anterior. Se han logrado curvas que pasan por los tres puntos de forma exacta en todos los casos excepto en la célula IBS60P 265W, que se desvía una media de  $6.57e-11$  Amperios, un valor despreciable.

Tabla 3-3. Errores de PSO para diodo doble.

	Sanyo HIT 215	Kyocera KC200GT	SPR-X22-370	IBS60P 265W	CHSM5001T- 100
Isc catálogo (A)	5.61	8.21	6.66	8.98	1.499
Impp catálogo (A)	5.13	7.61	6.26	8.58	1.188
Isc modelo (A)	5.61	8.21	6.66	8.98	1.499
Impp modelo (A)	5.13	7.61	6.26	8.58	1.188
Ioc modelo (A)	0	0	0	0	0
Error medio en intensidad	0	0	0	7.66e-10 %	0
tiempo (s) 10 iter.	59	65	78	48	43

El tiempo empleado es superior que en el modelo de diodo simple para todas las células evaluadas, lo que era de esperar como se ha tratado anteriormente debido a la mayor complejidad de este modelo. Los parámetros extraídos por el algoritmo se han recogido en la Tabla 3-4.

Tabla 3-4. Parámetros PSO para el modelo de diodo doble.

	Sanyo HIT 215	Kyocera KC200GT	SPR-X22-370	IBS60P 265W	CHSM5001T- 100
a	1.82	1.72	1.33	1.92	2.00
Rs (ohmios)	0.03	0.01	0.01	0.02	0.62
Rp (ohmios)	815	358	738	947	270
I0 (A)	5.02E-09	8.28E-06	3.65E-09	1.63E-07	1.00E-12
Ipv (A)	5.61	8.28E-06	6.66	8.98	1.50
a2	2.00	8.21	1.99	1.46	1.91
I02 (A)	4.73E-06	8.96E-11	3.40E-07	3.72E-07	1E-12

La convergencia es similar a la del caso anterior, como se aprecia en el ejemplo de la Figura 3.6, en el cual se acerca alrededor de la iteración 50 a un error cercano a cero, alcanzándolo tras más de 500 iteraciones.

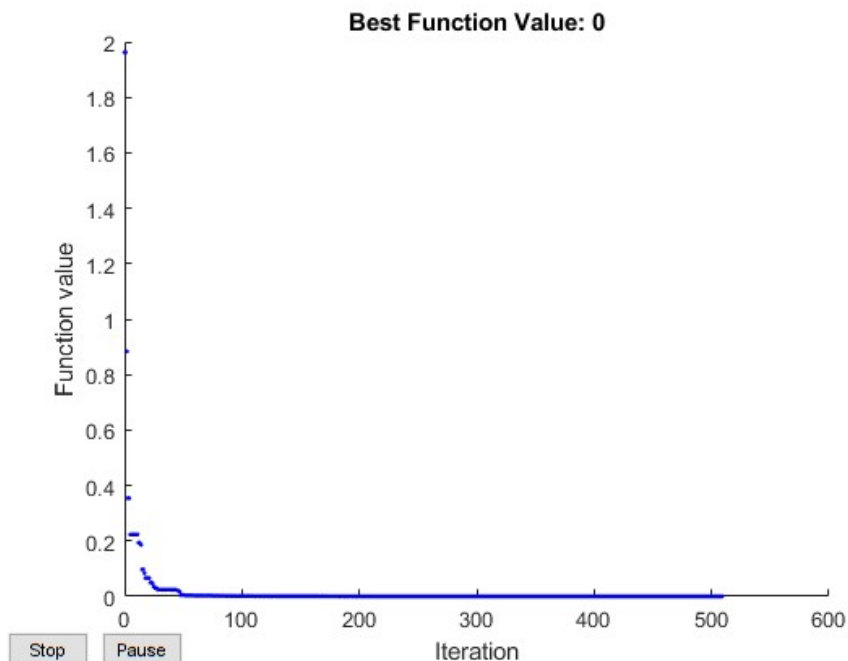


Figura 3.6. Convergencia de PSO para el modelo de diodo doble.

En la Figura 3.7 se ha trazado la curva I-V de la única célula con error distinto a cero, la célula IBS60P 265W. Como era de esperar la diferencia entre los puntos de catálogo y los calculados mediante los parámetros del modelo es insignificante.

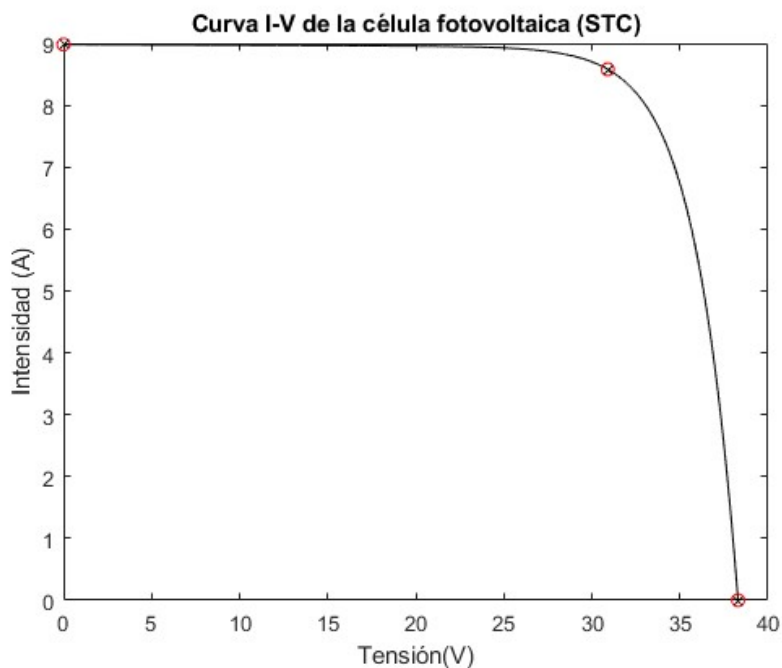


Figura 3.7. Curva I-V de la célula IBS60P 265W

### 3.2.3 Observaciones.

La precisión obtenida mediante el algoritmo Particle Swarm Optimization o PSO es elevada incluso para la célula solar de capa fina o *thin film*, cuya curva tiene una mayor pendiente en la zona de tensión inferior a la del punto de máxima potencia, como se puede apreciar en la Figura 3.8.

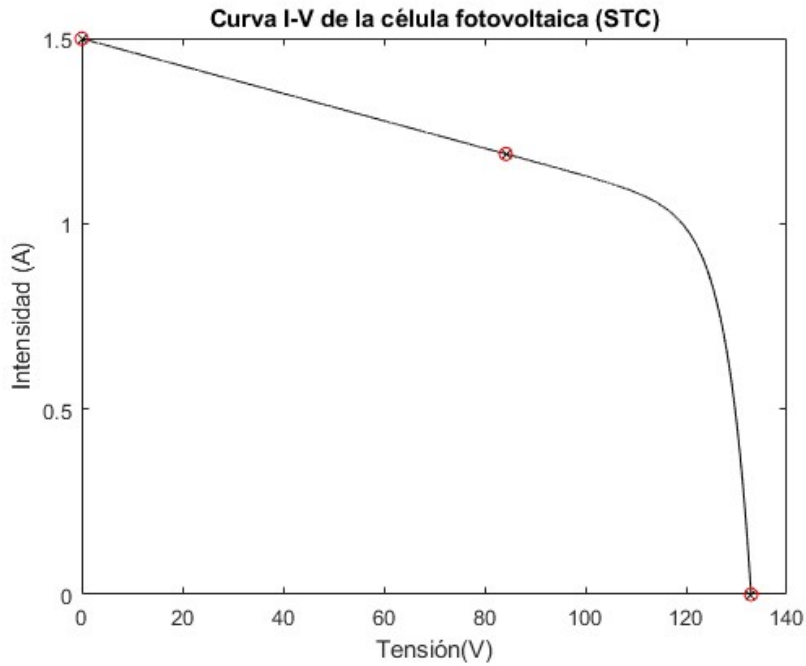


Figura 3.8. Curva I-V PSO de célula *thin film* (diodo doble).

El algoritmo se puede emplear tanto para el modelo de diodo simple, con un coste computacional y una precisión algo menores; como con el modelo de diodo doble, con el que se alcanza un ajuste perfecto y un tiempo de CPU mayor. Aunque los tiempos de ejecutar diez veces el algoritmo son de unos 50 a 70 segundos, el tiempo para obtener una precisión razonable es inferior a un segundo, ya que se alcanza en las primeras iteraciones de una sola ejecución del algoritmo.



## 4 ALGORITMO GENÉTICO (GA)

---

*No es el más fuerte de las especies el que sobrevive, tampoco es el más inteligente el que sobrevive. Es aquel que es más adaptable al cambio.*

*- Charles Darwin -*

Un algoritmo genético es un conjunto ordenado y finito de operaciones que permite hallar la solución de un problema inspirado en la evolución biológica y la estructura del ADN. Según la teoría de la evolución de Darwin (Figura 4.1), las especies mejor adaptadas son más susceptibles de sobrevivir y tener descendencia, lo que se trata de simular en el algoritmo creando los miembros de cada generación. El ácido desoxirribonucleico o ADN tiene una estructura, simplificando notablemente, similar a un vector que almacena información. Esta estructura se utiliza para representar a cada individuo de la simulación.

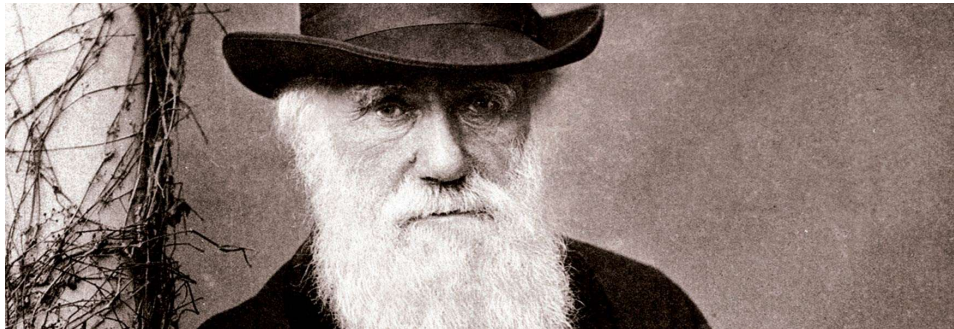


Figura 4.1. Charles Darwin.

El origen de los algoritmos genéticos se remonta a los años 60, cuando John Holland añadió el cruzamiento y la recombinación a la técnica de estrategia evolutiva de Ingo Rechenberg, en la cual solo estaba presente la mutación. En un inicio, la aplicación de estas técnicas era puramente teórica, aunque no tardaron en ser empleadas para problemas prácticos con el desarrollo de ordenadores más potentes, siendo ahora muy empleadas en diversos campos.

### 4.1. Funcionamiento de un algoritmo genético.

Como se ha explicado anteriormente, el algoritmo imita la evolución biológica. Para ello, se definen de forma aleatoria una serie de individuos con características diferentes. Estas características se pueden codificar de diferentes formas, por ejemplo, como un vector de dígitos binarios, en el que cada dígito representa un aspecto del individuo; o con un vector de números reales en el que cada componente representa también un aspecto del individuo.

La mayor o menor idoneidad de estos individuos se cuantifica mediante una función objetivo que se desea minimizar, y la cual depende de las características codificadas en cada sujeto.

Al tratarse de individuos definidos aleatoriamente es improbable que alguno de ellos minimice al máximo la función objetivo, pero habrá algunos mejores que otros. Los mejores se copian en la siguiente generación, recombinándose entre ellos y mutando ligeramente de forma aleatoria, dando lugar a nuevos individuos entre los cuales habrá también unos mejores que otros y con los que se repite el proceso. Al ir mejorando generación tras generación el algoritmo se va acercando a la solución deteniéndose al alcanzarla, al no apreciarse cambios en varias iteraciones, o al llegar a un límite de iteraciones. La secuencia del algoritmo está esquematizada en la Figura 4.2.

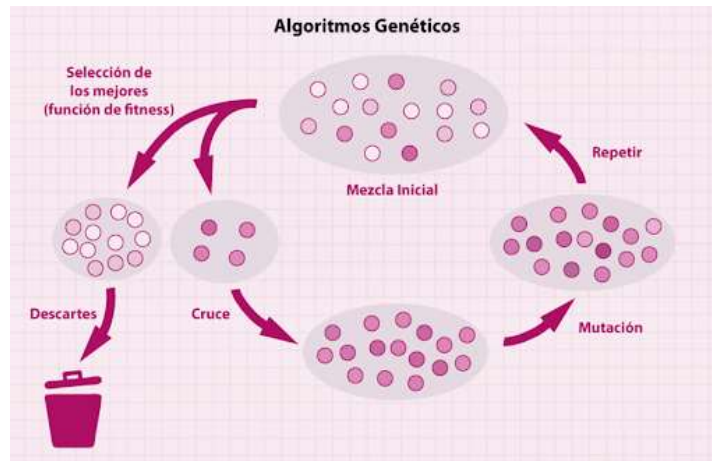


Figura 4.2. Funcionamiento de un algoritmo genético. [8]

## 4.2. Creación de cada generación.

En cada iteración el algoritmo emplea los individuos de dicha iteración para crear los de la siguiente. Existen tres tipos de descendientes para la siguiente generación:

- Élites: son los mejores individuos que sobreviven de forma idéntica a la siguiente iteración.
- Cruzados: son el resultado de una combinación de dos padres.
- Mutados: se obtienen al cambiar de forma aleatoria cambios a un único individuo.

### 4.2.1 Élites

Los élitos se conservan para asegurar la preservación de la mejor solución frente a posibles cambios que puedan empeorarla. Es importante que sean un número pequeño frente al total de la población, ya que de lo contrario no se conseguirá una convergencia rápida y efectiva hacia la solución global.

### 4.2.2 Cruce

El cruce simula la reproducción natural de la mayoría de seres vivos, en la que dos individuos tienen descendientes que son una combinación de las características de ambos progenitores. Existen diversas formas de cruzar padres, algunas de ellas son:

- Dispersa: se toman los elementos (genes) del vector de un progenitor u otro de forma aleatoria.

Padre 1: ABCDEFGH

Padre 2: 12345678

Hijo: A234E6GH

- Punto único: se toman los primeros genes de un padre y los últimos de otro padre.

Padre 1: ABCDEFGH

Padre 2: 12345678

Hijo: ABC45678

- Dos puntos: se toman genes de la cabeza y de la cola de un progenitor y los intermedios del otro.

Padre 1: ABCDEFGH

Padre 2: 12345678

Hijo: AB3456GH

- Intermedia: se toma para cada gen un valor aleatorizado entre los valores de ambos progenitores en dicho gen.

$$\text{Hijo} = \text{Padre 1} + \text{rand} * (\text{Padre 2} - \text{Padre 1}) \quad \text{rand} \in [0,1]$$

- Heurística: es similar a la intermedia, pero es más probable que tome valores más cercanos al progenitor más apto.
- Aritmética: crea hijos con la media aritmética ponderada de los padres.

Para elegir qué padres tienen descendencia en la siguiente generación y cuántos descendientes, existen varios criterios de selección, siendo los más empleados los siguientes:

- Selección uniforme estocástica: se traza una línea en la que cada padre es representado como una porción de la línea proporcional a su aptitud. El algoritmo se mueve por la línea en intervalos de igual tamaño eligiendo a los padres cuyos sectores de la línea son visitados por el algoritmo. La posición inicial es aleatoria.
- Selección por ruleta: se simula una ruleta en la que el área del sector circular de cada padre es proporcional a su aptitud y se elige de forma aleatoria el ángulo de la aguja para seleccionar a los padres.
- Selección por torneo: se dividen los individuos en grupos, escogiendo al mejor de cada grupo como padre.
- Selección *remainder*: se toma el valor escalado de la aptitud de cada candidato y se separa la parte entera de la decimal. La parte entera determina las veces que es seleccionado como padre y la parte decimal su probabilidad en una selección estocástica de ser incluido más veces.

- Selección uniforme: no tiene en cuenta la aptitud de cada padre, al ser menos efectiva que el resto se emplea para depurar el programa únicamente.

### 4.2.3 Mutación

Los individuos resultantes de un cruce tienen una pequeña probabilidad de mutar viéndose alteradas algunas de sus características de forma aleatoria. En algunas ocasiones el resultante de este proceso es mejor que el anterior individuo. Existen diferentes opciones para determinar cómo se hacen estos cambios, por ejemplo, según una función gaussiana, uniforme o *adaptive feasible*. Este proceso es importante para asegurar la diversidad de la población y también existe en la naturaleza si se da en células reproductoras pudiendo dar lugar a descendientes mejores, peores o iguales.

## 4.3. Aplicación del algoritmo genético en MATLAB al problema.

La función objetivo a minimizar es el sumatorio de los errores en intensidad para los puntos definidos en el catálogo (cortocircuito, circuito abierto y máxima potencia). Para cuantificar la función objetivo de cada individuo en cada generación se calcula la intensidad dada por los parámetros del genoma de cada miembro en los tres puntos de catálogo para las tensiones en estos puntos y se comparan entre sí. La expresión de la curva I-V y el número de parámetros depende del modelo utilizado, en este capítulo se ha estudiado el algoritmo con el modelo de diodo simple dependiente de tres y cinco parámetros y el modelo de diodo doble.

### 4.3.1 Modelo de diodo simple dependiente de tres parámetros.

El modelo empleado en primer lugar es el de diodo simple, el cual tiene cinco parámetros de los que depende la curva I-V. Para simplificar y acelerar el cálculo de los parámetros en ocasiones se definen solo tres parámetros independientes ( $R_s$ ,  $R_p$  y  $a$ ), siendo  $I_{PV}$  e  $I_0$  dependientes de estos tres según las expresiones (3.1) y (3.2).

$$I_{PV} = I_{SC} \frac{R_p + R_s}{R_p} \quad (3.1)$$

$$I_0 = \frac{I_{SC}}{\exp\left(\frac{qV_{OC}}{aKT_{STC}N_C}\right) - 1} \quad (3.2)$$

Para cada generación se calcula el valor de estos dos parámetros a partir de los tres independientes y se evalúa la función objetivo.

En cuanto a las opciones del algoritmo se han tomado 100 generaciones como máximo con poblaciones de 50 individuos, de los cuales 3 son los élites de la generación anterior, 12 son mutaciones de individuos y 35 son el resultado del cruce de dos miembros de la anterior iteración. La función de selección elegida es la *remainder* y la función de cruce es heurística. Las condiciones de detención son alcanzar el límite de generaciones, una solución tan parecida a cero como permite MATLAB, o no tener cambios en la mejor solución durante 15 iteraciones (condición *stall*).

Los límites de los parámetros independientes son los siguientes:

$$a \in [1, 2] \quad R_s \in [0.01, 1.2] \quad R_p \in [50, 1000]$$

Después de ejecutar el algoritmo diez veces se ha imprimido por pantalla el mínimo error obtenido y las intensidades en los tres puntos importantes de la curva I-V, así como los parámetros para dicho error. Los resultados se recogen en la Tabla 4-1.

Tabla 4-1. Errores GA, 3 parámetros independientes.

	Sanyo HIT 215	Kyocera KC200GT	SPR-X22-370	IBS60P 265W	CHSM5001T- 100
Isc catálogo (A)	5.61	8.21	6.66	8.98	1.499
I <sub>mpp</sub> catálogo (A)	5.13	7.61	6.26	8.58	1.188
Isc modelo (A)	5.61	8.21	6.66	8.98	1.499
I <sub>mpp</sub> modelo (A)	5.13	7.61	6.26	8.58	1.415
I <sub>0</sub> modelo (A)	0.0463	0.0298	6.88E-02	3.57E-02	-1.31E-01
Error medio en intensidad	0.30%	0.13%	0.37%	0.14%	10.05%
tiempo (s) 10 iter.	24	20	29	20	40

De la tabla se desprende que, aunque la curva pasa exactamente por los puntos de cortocircuito y máxima potencia en la mayoría de los casos, el algoritmo tiene dificultad para lograr una curva que pase también por el punto de intensidad nula. Por otra parte, el error medio para la célula *thin film* es excesivo, siendo de 10.05%. Los parámetros para estas curvas son los reflejados en la Tabla 4-2.

Tabla 4-2. Parámetros GA, 3 parámetros independientes.

	Sanyo HIT 215	Kyocera KC200GT	SPR-X22-370	IBS60P 265W	CHSM5001T -100
a	1.00	1.00	1.33	1.00	2.00
R <sub>s</sub> (ohmios)	0.952	0.382	0.101	0.289	1.200
R <sub>p</sub> (ohmios)	1000	1000	1000	1000	1000
I <sub>0</sub> (A)	4.31E-12	4.13E-10	4.33E-09	1.46E-10	6.76E-12
I <sub>pv</sub> (A)	5.6	8.2	6.7	9.0	1.5

Hay que destacar que el algoritmo alcanza mínimos con parámetros de valores iguales a los límites inferiores o superiores en numerosos casos, como por ejemplo la resistencia en paralelo de 1000 ohmios o el factor de idealidad del diodo que toma valores de 1 y 2. Este comportamiento no se debe a una mala elección de los límites, puesto que se ha llegado a soluciones de cero error con los mismos límites, sino que es debido probablemente a la excesiva simplicidad y la reducida libertad de trabajar con tres parámetros independientes.

Se han conseguido soluciones con este algoritmo para parámetros distintos a los límites cambiando la función de cruce de heurística a la *adaptive feasible*, la cual aleatoriza completamente los parámetros que toma de cada progenitor en vez de aumentar algo la probabilidad de tomar parámetros del progenitor más apto. Sin embargo, estas soluciones son peores que las obtenidas en los límites con las opciones elegidas.

En la Figura 4.3 se ha plasmado un ejemplo para apreciar la convergencia del algoritmo, el mejor individuo (en negro) llega al mínimo antes de las primeras diez generaciones. A pesar del alto número de mutaciones, que altera la media de la función objetivo de toda la población (en azul) durante toda la simulación, no se mejora demasiado la solución obtenida al inicio.

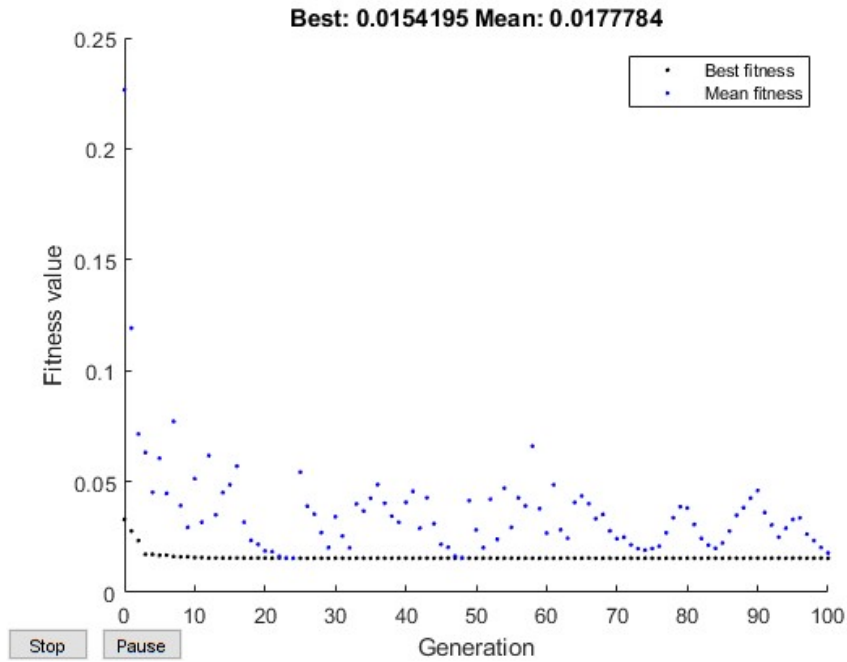


Figura 4.3. Convergencia del GA de 3 parámetros independientes.

En la Figura 4.4 se ha representado la curva I-V de la célula de película fina al tener un error considerablemente superior al resto. Se aprecia que se aleja en gran medida de los puntos por los que debería pasar. El resto de curvas tienen un buen comportamiento, notándose únicamente una ligera desviación en el punto de circuito abierto, como en la célula Sanyo HIT 215 del ejemplo de la Figura 4.5.

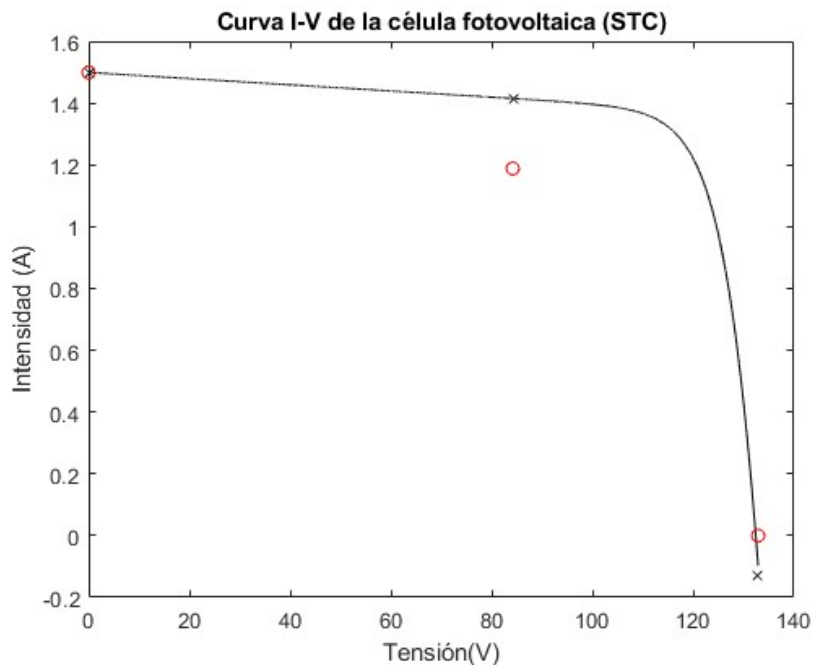


Figura 4.4. Curva I-V de la célula CHSM5001T-100. (GA3)

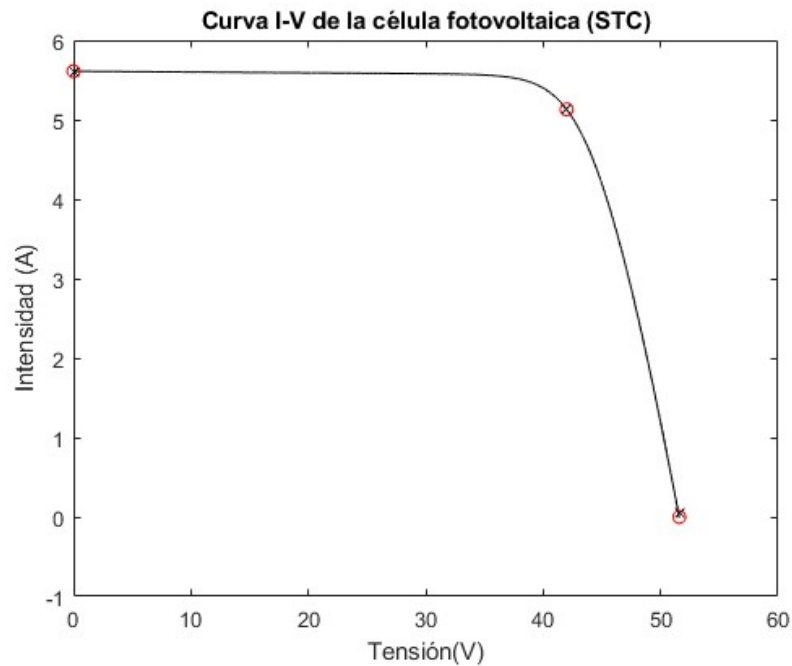


Figura 4.5. Curva I-V de la célula Sanyo HIT 215. (GA3)

#### 4.3.2 Modelo de diodo simple dependiente de cinco parámetros.

Con el propósito de intentar mejorar los resultados del modelo de diodo simple de tres parámetros, se ha procedido a liberar también los parámetros  $I_0$  e  $I_{PV}$  para trabajar ahora con cinco parámetros independientes. Esta modificación debería resultar en un aumento del tiempo de cálculo respecto al caso anterior al haber aumentado la complejidad, pero a cambio es posible que mejoren las soluciones alcanzadas.

En cuanto a las opciones del algoritmo se han mantenido las del caso de tres parámetros independientes, se han tomado 100 generaciones con poblaciones de 50 individuos, de los cuales 3 son los élites de la generación anterior, 12 son mutaciones de individuos y 35 son el resultado del cruce de dos miembros de la anterior iteración. La función de selección elegida es la *remainder* y la función de cruce es heurística. Las condiciones de detención también se mantienen.

Los límites de los parámetros independientes son los siguientes:

$$a \in [1, 2] \quad R_S \in [0.01, 1.2] \quad R_P \in [50, 1000]$$

$$I_0 \in [1e-12, 1e-5] \quad I_{PV} \in [1, 9]$$

El algoritmo se ejecuta diez veces y se imprime por pantalla el mínimo error obtenido y las intensidades en los tres puntos importantes de la curva I-V, así como los parámetros para dicho error. Los resultados se recogen en la Tabla 4-3.

Tabla 4-3. Errores GA, 5 parámetros independientes.

	Sanyo HIT 215	Kyocera KC200GT	SPR-X22-370	IBS60P 265W	CHSM5001T-100
Isc catálogo (A)	5.61	8.21	6.66	8.98	1.499
Impp catálogo (A)	5.13	7.61	6.26	8.58	1.188
Isc modelo (A)	5.61	8.21	7.00	8.99	1.499
Impp modelo (A)	5.13	7.61	6.26	8.54	1.188
loc modelo (A)	8.98E-10	-1.83E-12	5.65E-06	-6.54E-10	-2.80E-09
Error medio en intensidad (%)	6.09E-07	1.94E-06	1.79	0.49	1.70E-05
tiempo (s) 10 iter.	15	25	17	18	20

En comparación con los errores del caso anterior, se mejora en varios órdenes de magnitud para la primera, segunda y quinta célula, mientras que se empeora ligeramente en la tercera y cuarta célula; pasando de un 0.37% a 1.79% de error en la SPR-X22-370 y de un 0.14% a un 0.49% de error en la IBS60P 265W. Es evidente que este empeoramiento de la solución es eclipsado por la mejora en las otras células, de hasta siete órdenes de magnitud.

Hay que destacar que, aunque las primeras cifras de las intensidades del modelo de cortocircuito y máxima potencia son idénticas en algunos casos, existe cierto error en estos puntos que no se aprecia en la tabla, ya que para la representación se ha redondeado para una mejor visualización. El error medio de la penúltima fila sí que incluye estos errores al haberse obtenido directamente del programa.

En los tiempos no se aprecia diferencia a pesar del aumento de parámetros independientes. Se ha comprobado que en general se agotan las 100 generaciones establecidas tanto en el caso de tres parámetros independientes como en el de cinco, por lo que una convergencia más rápida no puede ser la causa de equilibrar el coste computacional derivado de aumentar los parámetros. Es probable que los tiempos sean similares debido a que en el caso simplificado de tres parámetros hay que calcular continuamente los otros dos parámetros, lo que puede ser incluso más costoso que trabajar con los cinco parámetros libres. El hecho de que la suma de tiempos en el caso anterior es de 107 segundos mientras que en este es de 96 segundos respalda esta teoría, aunque hay que destacar la gran dispersión de los tiempos, que cambian notablemente cada vez que se ejecuta el código.

Los parámetros para la mejor solución hallada están reflejados en la Tabla 4-4.

Tabla 4-4. Parámetros GA, 5 parámetros independientes.

	Sanyo HIT 215	Kyocera KC200GT	SPR-X22-370	IBS60P 265W	CHSM5001T-100
a	1.99	1.60	1.97	1.55	2.00
Rs (ohmios)	0.023	0.084	0.010	0.010	0.061
Rp (ohmios)	705	630	866	1000	271
IO (A)	4.46E-06	3.06E-06	4.17E-06	9.78E-07	4.40E-12
Ipv (A)	5.6	8.2	6.8	9.0	1.5

En este caso, a diferencia del anterior, los parámetros obtenidos son distintos a los límites de los intervalos en general, de lo que se deduce un mejor comportamiento del algoritmo y mayor probabilidad de hallar errores inferiores.

El algoritmo suele alcanzar las 100 generaciones establecidas como límite. Aunque puede parecer que este límite impide seguir iterando para afinar la solución, se ha comprobado que antes de la generación 40 la reducción de la función objetivo es residual.

En la Figura 4.6 se ha representado un ejemplo de la evolución de la mejor solución con el transcurso de las generaciones, en este caso se puso un límite de 1000 generaciones para que la condición de detención sea la



ausencia de cambios en la mejor solución durante 15 iteraciones consecutivas, este ejemplo es el resultado de ampliar una de las gráficas obtenidas. El algoritmo solía detenerse entre las 200 y 600 generaciones, con formas siempre parecidas a la de la figura. Al parecer la solución alcanzada con los primeros cruces y mutaciones es determinante, por lo que para alcanzar mejores soluciones la línea a seguir no es aumentar el número de generaciones, sino repetir el algoritmo para partir de una solución previa diferente.

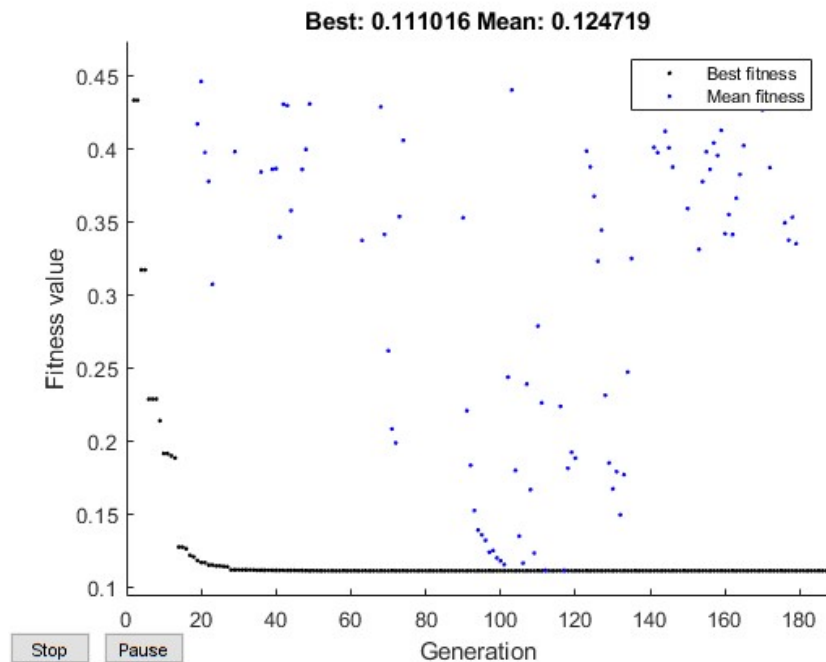


Figura 4.6. Convergencia del GA de 5 parámetros independientes.

En la Figura 4.7 y en la Figura 4.8 se ha representado la curva I-V de la célula con mayor error y la de la célula de película fina cuyo error en el caso anterior era del 10.05% respectivamente. En la primera gráfica se puede ver que el error medio de 1,79% es excesivo, la diferencia en el punto de cortocircuito es claramente visible, mientras que en la segunda figura los resultados son altamente satisfactorios.

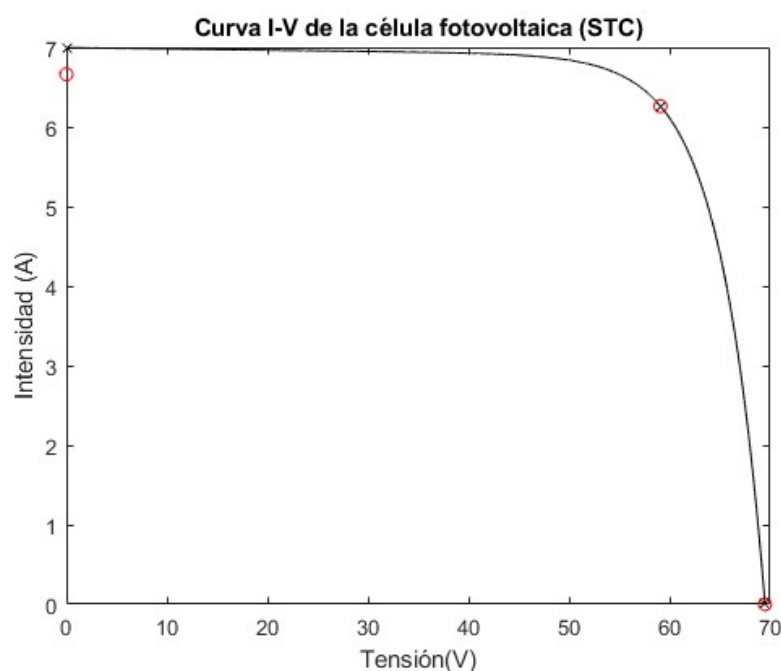


Figura 4.7. Curva I-V de la célula SPR-X22-370. (GA5)

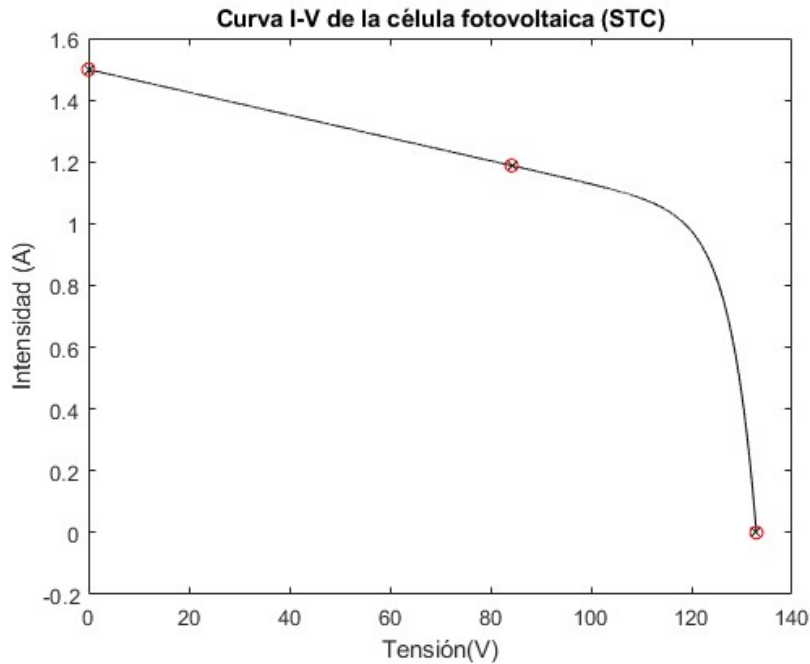


Figura 4.8. Curva I-V de la célula CHSM5001T-100. (GA5)

### 4.3.3 Modelo de diodo doble.

Los resultados de los modelos anteriores son mejorables, por lo que es necesario seguir complicando el modelo para alcanzar errores más pequeños. En este epígrafe se ha probado el algoritmo con el modelo de diodo doble, para el que hay que añadir los dos parámetros relativos al segundo diodo a los cinco del modelo de diodo simple.

La mayoría de las opciones del algoritmo se han mantenido, tomando poblaciones de 50 individuos, de los cuales 3 son los élites de la generación anterior, 12 son mutaciones de individuos y 35 son el resultado del cruce de dos miembros de la anterior iteración. La función de selección elegida es la *remainder* y la función de cruce es heurística. Las condiciones de detención también se mantienen. Respecto a los casos anteriores se ha tenido que aumentar el número de generaciones de 100 a 500, ya que la media de los errores de las soluciones iniciales es normalmente del orden de decenas de miles y se ha necesitado un mayor número de iteraciones para alcanzar el error mínimo.

A los límites de los parámetros del diodo simple de antes, cuyos intervalos se han respetado, se añaden dos nuevos relativos al segundo diodo. Son los siguientes:

$$\begin{array}{lll}
 a \in [1, 2] & R_S \in [0.01, 1.2] & R_P \in [50, 1000] \\
 I_0 \in [1e-12, 1e-5] & I_{PV} \in [1, 9] & a_2 \in [1, 2] & I_{02} \in [1e-12, 1e-5]
 \end{array}$$

Se ejecuta el algoritmo diez veces y se extrae una serie de resultados como en los casos anteriores, representado en la Tabla 4-5.

Tabla 4-5. Errores GA, diodo doble.

	Sanyo HIT 215	Kyocera KC200GT	SPR-X22-370	IBS60P 265W	CHSM5001T- 100
Isc catálogo (A)	5.61	8.21	6.66	8.98	1.499
Impp catálogo (A)	5.13	7.61	6.26	8.58	1.188
Isc modelo (A)	5.610	8.210	6.662	8.981	1.499
Impp modelo (A)	5.130	7.610	6.076	8.443	1.188
loc modelo (A)	-4.44E-09	1.14E-10	-4.48E-05	2.97E-13	-7.86E-04
Error medio en intensidad (%)	1.34E-06	1.13E-07	0.99	0.54	5.04E-05
tiempo (s) 10 iter.	28	31	36	65	71

Los errores se han reducido ligeramente en las células segunda y tercera, y ha aumentado ligeramente en las otras tres. El orden de los errores es similar en cada caso, siendo necesario un mayor número de generaciones y en algunas ocasiones no ha bastado con ejecutar diez veces el algoritmo, teniendo que repetir el código, como por ejemplo en la última célula. Los tiempos son superiores debido al aumento de 100 a 500 generaciones.

Los parámetros del modelo para estos errores son los de la Tabla 4-6.

Tabla 4-6. Parámetros GA, diodo doble.

	Sanyo HIT 215	Kyocera KC200GT	SPR-X22-370	IBS60P 265W	CHSM5001T- 100
a	1.83	1.86	1.89	1.98	1.99
Rs (ohmios)	0.06	0.02	0.01	0.01	1.17
Rp (ohmios)	419	942	440	1000	270
I0 (A)	7.58E-07	1.88E-06	1.46E-07	9.17E-06	1.88E-12
Ipv (A)	5.61	8.21E+00	6.66	8.98	1.51
a2	1.97	1.74	1.55	1.51	2.00
I02 (A)	1.56E-06	8.84E-06	7.46E-08	4.50E-07	2.36E-12

En cuanto al comportamiento del algoritmo ya se empiezan a apreciar las limitaciones del algoritmo genético con el aumento de la complejidad de un problema. En la Figura 4.9 se ha representado una ejecución cualquiera del algoritmo para una de las células, en este caso se detuvo antes de la ejecución 300 con un error de 0.106 A. Hay que destacar el valor medio de las soluciones de las dos primeras generaciones, superior a 20000 en el primer caso y superior a 5000 en el segundo. Vemos que en este ejemplo se ha logrado una rápida bajada del error, sin embargo, es común en otras ejecuciones permanezca en estos altos valores, como se verá en el siguiente ejemplo.

En la Figura 4.10 se puede ver una ampliación del ejemplo anterior, donde se aprecia la enorme dispersión de las medias de los errores de la población, y el rápido estancamiento de la mejor solución.

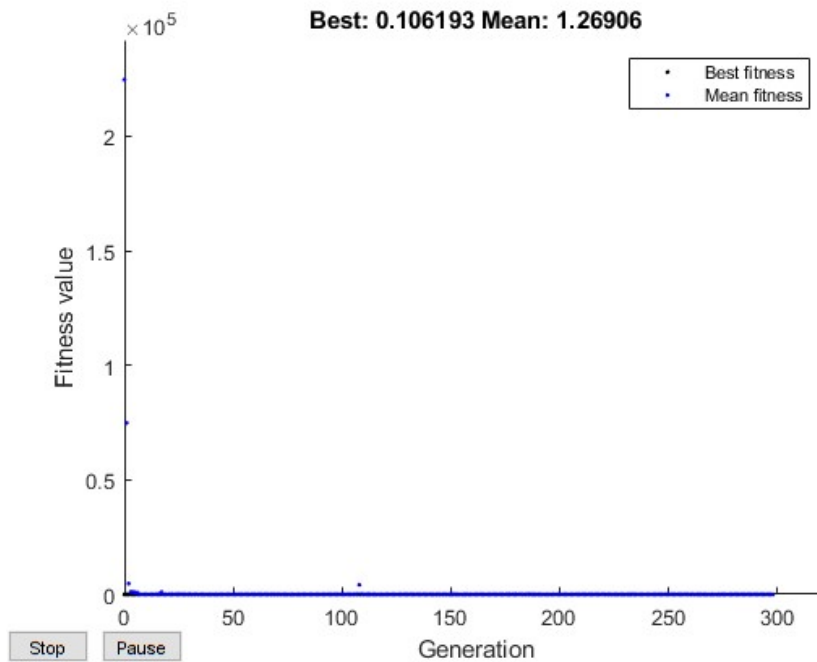


Figura 4.9. Evolución algoritmo GA diodo doble, vista general.

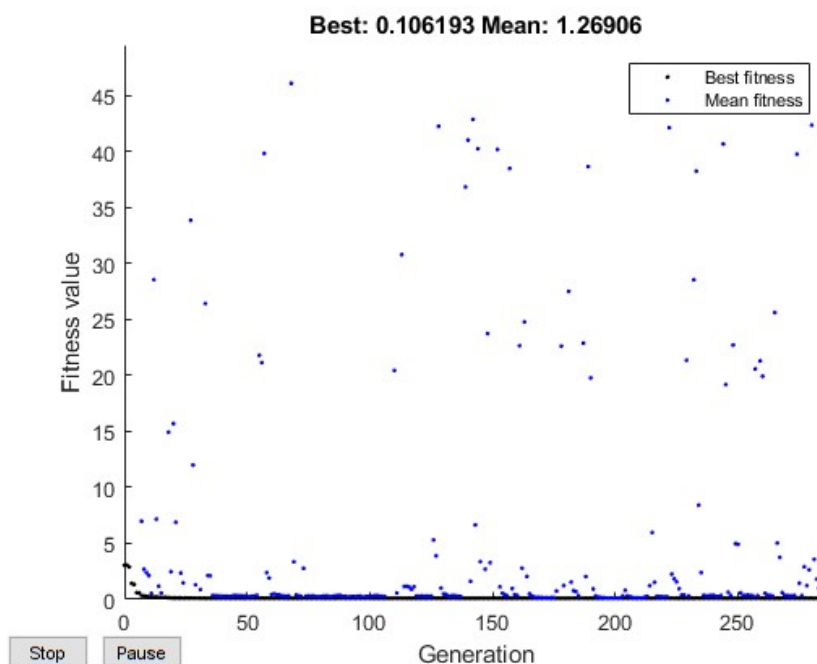


Figura 4.10. Evolución algoritmo GA diodo doble, detalle.

En la Figura 4.11 se ha plasmado el ejemplo de una mala convergencia del algoritmo. Es una de las numerosas ejecuciones del algoritmo para la célula de película fina en la que el GA es incapaz de dar una buena solución, a pesar de detenerse tras casi 1000 generaciones. La razón de la detención en este caso fue la condición de *stall*, por lo que en las últimas 15 generaciones no se mejoró el mínimo alcanzado. Esta figura es una ampliación de la evolución, ya que hay medias muy superiores al ya elevado valor del extremo superior del eje Y, siendo la media inicial de errores  $6 \cdot 10^{15}$  A. Este caso es un claro ejemplo de la dificultad del algoritmo de abandonar el entorno de un mínimo relativo. Es apreciable como la mejor solución permanece cerca de 400000 A durante unas 400 generaciones, pasando entonces a valores cercanos a 10000 A, deteniéndose finalmente cerca de este mínimo tras unas 600 generaciones que mejoran ligeramente la solución.

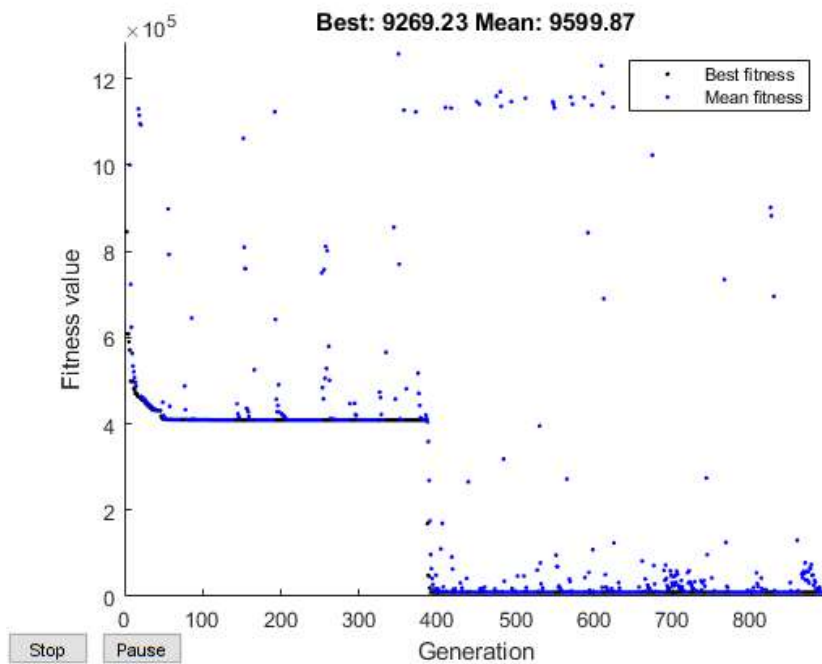


Figura 4.11. Evolución algoritmo GA diodo doble, mala convergencia.

En cuanto a las formas de las curvas I-V obtenidas se puede decir que son las correctas, se ha incluido a modo de ejemplo la de la célula IBS60P 265W en la Figura 4.12.

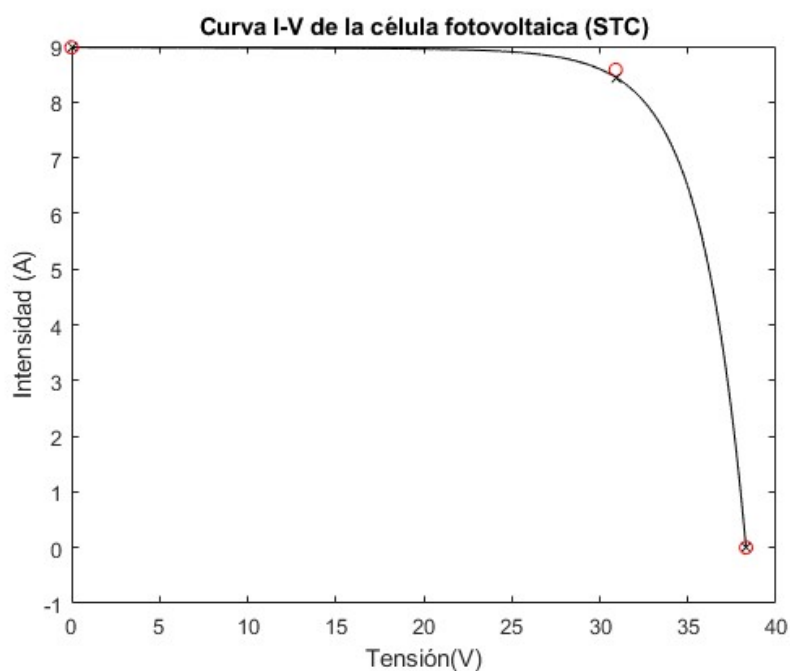


Figura 4.12. Curva I-V de la célula IBS60P 265W. (GA7)

#### 4.4. Conclusiones de la aplicación del algoritmo genético al problema.

Para comparar los diferentes modelos y su comportamiento con cada célula se ha representado en la Figura 4.13 los errores medios en tanto por ciento de todas las células para el modelo de diodo simple de tres parámetros independientes, el de diodo simple de cinco parámetros independientes y el de diodo doble. El eje Y está en

escala logarítmica invertida, por lo que más altura implica menor error.

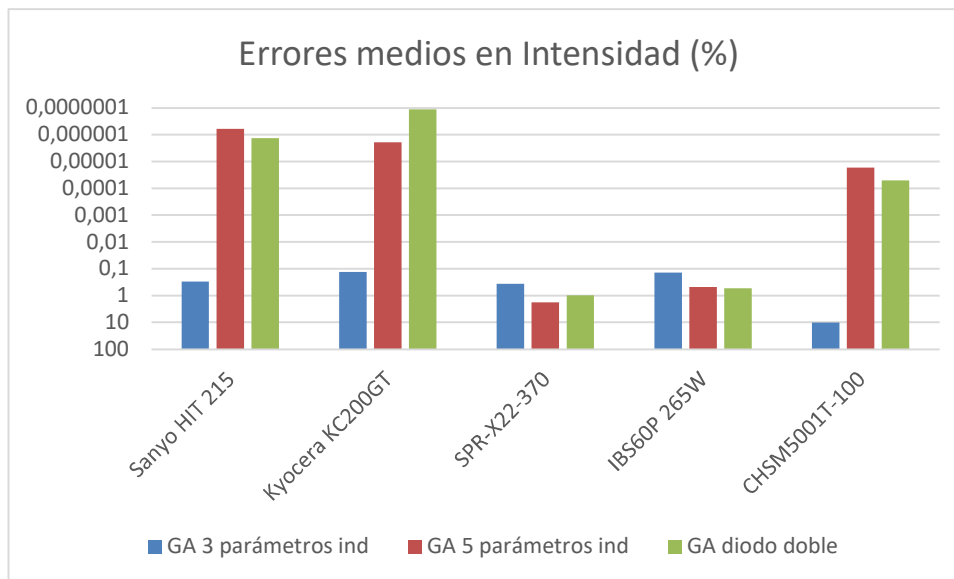


Figura 4.13. Comparación de modelos del GA.

De la gráfica se extrae que el modelo de diodo simple con tres parámetros es menos efectivo en general. Los modelos de diodo simple con cinco parámetros independientes y diodo doble tienen unos errores similares, sin una clara hegemonía de uno sobre otro, lo que no justifica emplear el de diodo doble debido a la mayor complejidad que dificulta la convergencia hacia un error aceptable, como se ha desarrollado anteriormente. Estos dos últimos modelos superan ampliamente al modelo de tres parámetros para la célula Sanyo y para la Kyocera, elegidas a partir del artículo [9] en el que los autores tratan de validar el uso de algoritmos genéticos en la obtención de parámetros del modelo, mientras que en las células SPR-X22-370 y la IBS60P, que han sido tomadas de forma aleatoria, los resultados son igualmente deficientes en los tres modelos.

Para la célula de película fina se obtiene un error aceptable en los modelos de diodo doble y simple de cinco parámetros, sin embargo, este error es el mejor obtenido tras muchas ejecuciones del algoritmo en las que los errores eran de órdenes superiores a miles.

Para terminar, hay que destacar que estos resultados son los mejores obtenidos para una serie de opciones determinadas, es factible que se puedan obtener errores inferiores con otras opciones a pesar de haber intentado optimizar las opciones de la mejor manera posible, ya que el número de opciones y sus combinaciones son elevadas.

## 5 ALGORITMO PATTERN SEARCH (PS)

*No todos los que caminan sin rumbo están perdidos.*

*– J.R.R. Tolkien –*

El algoritmo *pattern search* es un algoritmo de optimización descrito por Hooke y Jeeves en 1961 [10], aunque se atribuye una versión más simple a los físicos Enrico Fermi y Nicholas Metropolis. Como su nombre sugiere se basa en una búsqueda de la solución siguiendo un patrón de exploración en un espacio de tantas dimensiones como variables tenga el problema.

### 5.1. Funcionamiento del algoritmo.

El algoritmo parte de una solución inicial cualquiera en forma de vector que se asigna como punto base, en el que cada componente del vector es una variable del problema a optimizar. Se exploran los puntos alrededor de esta solución inicial y se calcula el valor de la función objetivo variando las componentes del punto inicial uno a uno sumando y restando el tamaño inicial del mallado; por ejemplo, en un problema de tres variables en el que el punto inicial sea  $[3,3,3]$  y el mallado inicial 1, se exploran los puntos  $[2,3,3]$ ,  $[4,3,3]$ ,  $[3,2,3]$ ,  $[3,4,3]$ ,  $[3,3,4]$  y  $[3,3,2]$ .

A continuación, se comparan los valores de la función objetivo en todos los puntos, si la mejor solución se encuentra en alguno de los nuevos puntos éste pasa a ser el nuevo punto base y se procede como antes. Si el mejor punto es el punto base se reduce el tamaño del mallado y se vuelve a iterar. El algoritmo se detiene cuando se alcanza un tamaño de mallado determinado o un límite de iteraciones.

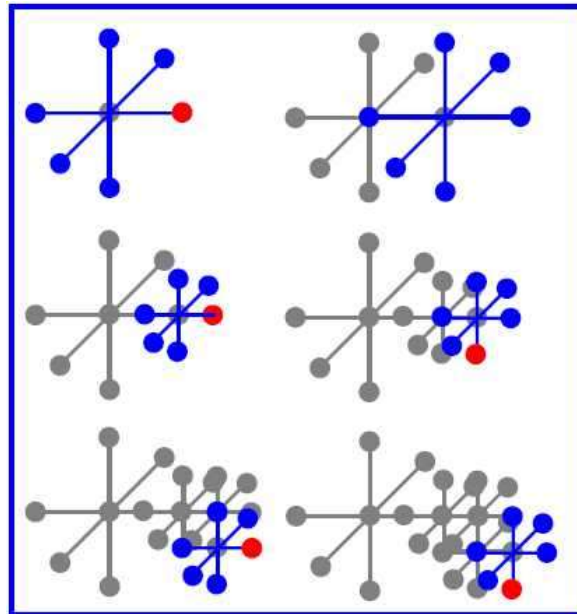


Figura 5.1. Funcionamiento del PS. [11]

En la Figura 5.1 se ha representado gráficamente el comportamiento del algoritmo para un problema de 3 parámetros, se puede ver como en la primera iteración se encuentra una solución mejor (en rojo) mientras que en la segunda no, reduciendo el tamaño del mallado, el cual se mantiene en sucesivas iteraciones al ir mejorando la solución.

## 5.2. Pattern search en MATLAB.

Para implementar el algoritmo en MATLAB se ha definido la misma función objetivo que para otros algoritmos, minimizar el error de intensidad en los tres puntos de catálogo (cortocircuito, circuito abierto y máxima potencia). La función objetivo se calcula sustituyendo las coordenadas de un punto en la ecuación de la característica I-V de la célula para cada modelo empleado. Para este algoritmo se han probado los modelos de diodo simple y diodo doble.

### 5.2.1 Modelo de diodo simple.

El modelo tiene cinco parámetros independientes como ya se ha desarrollado en el capítulo de los modelos y los límites tomados son los mismos que para otros algoritmos:

$$a \in [1, 2] \quad R_s \in [0.01, 1.2] \quad R_p \in [50, 1000]$$

$$I_0 \in [1e-12, 1e-5] \quad I_{pv} \in [1, 9]$$

Para este algoritmo es importante la solución inicial, ya que es fácil que alcance un mínimo local, por lo que se ha ejecutado con diez soluciones iniciales diferentes. Para ello se ha dividido la longitud entre los límites de cada parámetro en diez porciones y se ha tomado el primer punto de cada parámetro para la primera ejecución, el segundo punto para la segunda y así sucesivamente. Es evidente que estas soluciones iniciales son muy limitadas y se encuentran alineadas en el espacio vectorial, pero para distribuir soluciones iniciales por un espacio de 5 dimensiones hacen falta numerosas iteraciones.

El tamaño de mallado inicial es 1, por lo que en primer lugar se irá moviendo solo alterando la resistencia en paralelo y la  $I_{pv}$ , al ir reduciéndose se irá explorando el resto de parámetros. El mallado mínimo es  $1e-17$  y el número máximo de iteraciones 500. Los errores se han reflejado en la Tabla 5-1:



Tabla 5-1. Errores PS diodo simple.

	Sanyo HIT 215	Kyocera KC200GT	SPR-X22-370	IBS60P 265W	CHSM5001T- 100
Isc catálogo (A)	5.61	8.21	6.66	8.98	1.499
Impp catálogo (A)	5.13	7.61	6.26	8.58	1.188
Isc modelo (A)	5.78	8.21	7.14	9.00	3.50
Impp modelo (A)	5.13	7.61	6.10	8.31	3.099
loc modelo (A)	-6.04E-11	9.18E-08	6.80E-10	1.23E-11	-4.38E+03
Error medio en intensidad (%)	1.11	6.29E-05	3.45	1.11	1.23E+05
tiempo (s) 10 iter.	35	34	23	22	24

Los errores para este algoritmo son bastante altos, siendo incapaz de dar una solución aceptable en el caso de la célula de película fina y en el resto de casos de alrededor de 2%. Los tiempos son de unos 30 segundos de media. Los parámetros de la célula para estos errores se han recogido en la Tabla 5-2.

Tabla 5-2. Parámetros diodo simple.

	Sanyo HIT 215	Kyocera KC200GT	SPR-X22-370	IBS60P 265W	CHSM5001T- 100
a	1.93	1.56	1.93	1.81	2.00
Rs (ohmios)	0.303	0.056	0.010	0.010	0.164
Rp (ohmios)	832	193	215	998	1000
IO (A)	3.00E-06	2.00E-06	3.00E-06	1.00E-05	1.98E-08
Ipv (A)	5.8	8.2	7.2	9.0	0.0

En la Figura 5.2 se ha representado la evolución del algoritmo. Podemos ver que en este ejemplo la condición de detención que se cumple es la de 500 iteraciones máximas, a pesar de agotar el número de iteraciones y partir de una solución inicial para la función objetivo inferior a 3 Amperios, el error medio alcanzado es de 1.02623 A, un error considerable. El número máximo de iteraciones parece adecuado, ya que existe mejora apreciable hasta pasadas las 200 iteraciones.

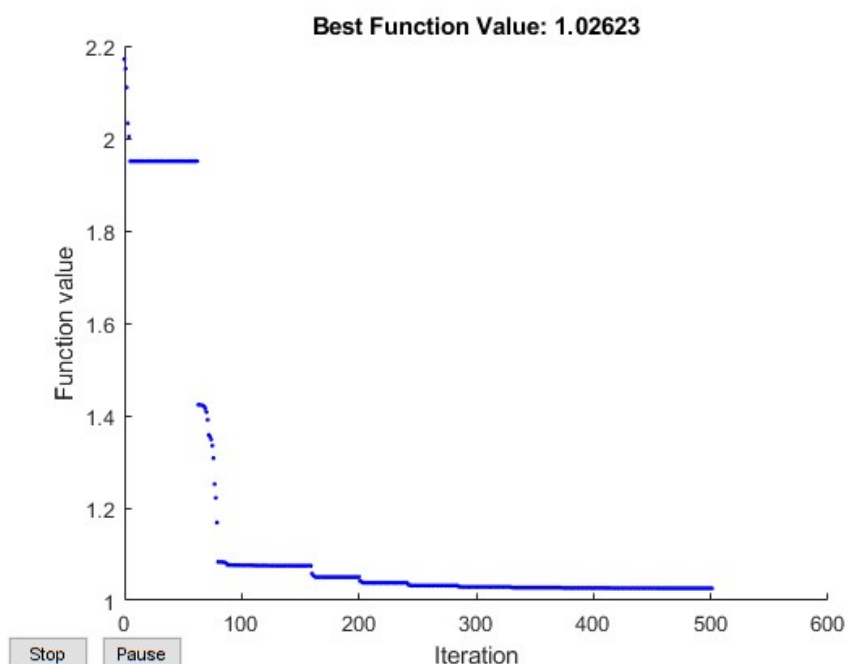


Figura 5.2. Evolución del PS, diodo simple.

A continuación, en la Figura 5.3 y la Figura 5.4 se han representado respectivamente las curvas I-V de la célula con menor error y la de mayor error obviando la de película fina que es inadmisibile. Se puede apreciar en la segunda figura claramente la desviación de la curva, ya que un error de un 3.45% es un valor elevado.

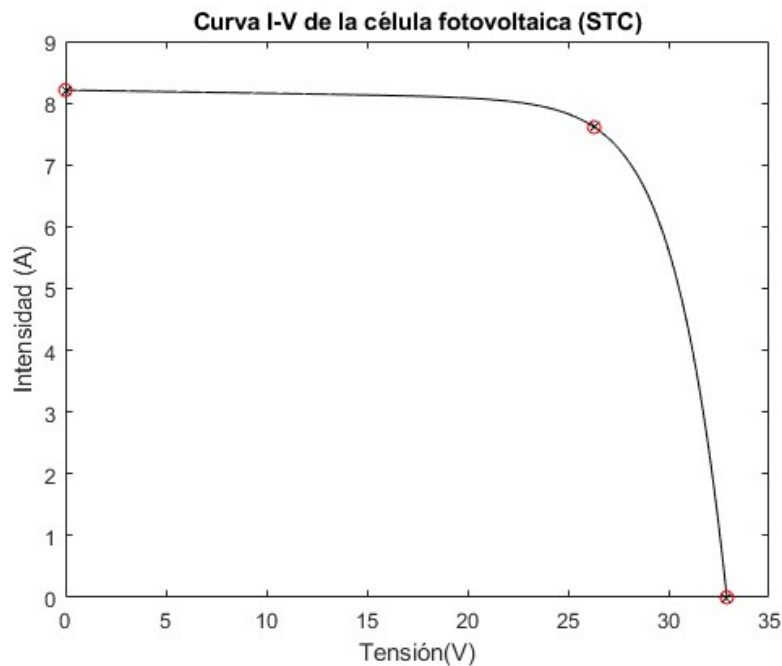


Figura 5.3. Curva I-V de la célula KC200GT, PS diodo simple.

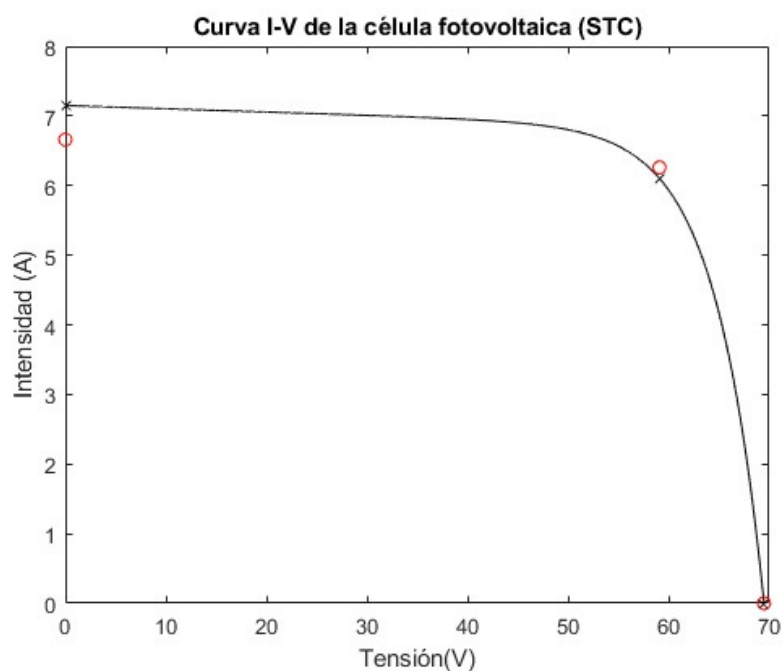


Figura 5.4. Curva I-V de la célula SPR-X22-370, PS diodo simple.

## 5.2.2 Modelo de diodo doble.

Los límites de los siete parámetros independientes de este modelo son los siguientes:

$$\begin{array}{llll}
 a \in [1, 2] & R_S \in [0.01, 1.2] & R_P \in [50, 1000] & \\
 I_0 \in [1e-12, 1e-5] & I_{PV} \in [1, 9] & a_2 \in [1, 2] & I_{02} \in [1e-12, 1e-5]
 \end{array}$$

Se han mantenido los tamaños de mallado inicial y mallado mínimo, y se ha aumentado a 700 el número máximo de iteraciones, ya que es recomendable un mayor número al existir más variables independientes. El algoritmo se ha repetido 10 veces obteniendo soluciones iniciales diferentes de manera análoga al caso anterior. Los errores se han recogido en la Tabla 5-3.

Tabla 5-3. Errores PS diodo doble.

	Sanyo HIT 215	Kyocera KC200GT	SPR-X22-370	IBS60P 265W	CHSM5001T- 100
Isc catálogo (A)	5.61	8.21	6.66	8.98	1.499
Impv catálogo (A)	5.13	7.61	6.26	8.58	1.188
Isc modelo (A)	5.770	8.220	6.925	8.983	2.999
Impv modelo (A)	5.130	7.574	5.961	8.366	1.188
loc modelo (A)	-2.69E-11	3.28E-10	8.94E-10	3.71E-11	-2.05E+04
Error medio en intensidad (%)	1.04	0.18	3.00	0.84	576410.77
tiempo (s) 10 iter.	45	29	40	21	21

Los errores obtenidos son similares a los del modelo de diodo simple con unos tiempos algo superiores, el algoritmo sigue siendo incapaz de alcanzar una solución aceptable para la célula de película fina. Los parámetros del modelo para cada célula con estos errores se han recogido en la Tabla 5-4.

Tabla 5-4. Parámetros PS diodo doble.

	Sanyo HIT 215	Kyocera KC200GT	SPR-X22-370	IBS60P 265W	CHSM5001T -100
a	1.99	1.65	1.91	1.70	2.00
Rs (ohmios)	0.27	0.01	0.01	0.01	0.81
Rp (ohmios)	1000	177	230	438	1000
I0 (A)	2.00E-06	4.00E-06	1.00E-06	2.00E-06	4.63E-08
Ipv (A)	5.77	8.22E+00	6.93	8.98	3.00
a2	1.95	1.90	1.85	1.70	2.00
I02 (A)	2.00E-06	4.00E-06	1.00E-06	2.00E-06	4.63E-08

El criterio de detención predominante es el de mínimo tamaño de mallado, alcanzando raramente las 700 iteraciones.

### 5.3. Conclusiones de la aplicación del algoritmo pattern search al problema.

Con los resultados obtenidos de la aplicación del algoritmo es evidente que la exploración del espacio es deficiente para este problema, tanto en un modelo como en el otro, alcanzando soluciones parciales siempre bastantes alejadas de un error nulo. Esta exploración deficiente es posible que se pueda suplir aumentando el número de soluciones iniciales desde las que se parte y tomando el mejor resultado en cada caso. El coste computacional de esto es elevado, ya que para cubrir el espacio vectorial de forma uniforme el número de soluciones iniciales es alto. Por ejemplo, en el caso del modelo dependiente de cinco parámetros, tomar diez puntos diferentes para cada parámetro resulta en  $10^5$  soluciones iniciales.

En la Figura 5.5 se han graficado los errores medios en intensidad de cada célula para cada modelo con el objetivo de poder visualizar si hay diferencias entre uno y otro. Se puede comprobar que los errores de ambos modelos son parecidos, habiendo una clara diferencia solo en el caso de la célula KC200GT, lo que seguramente se deba al azar.

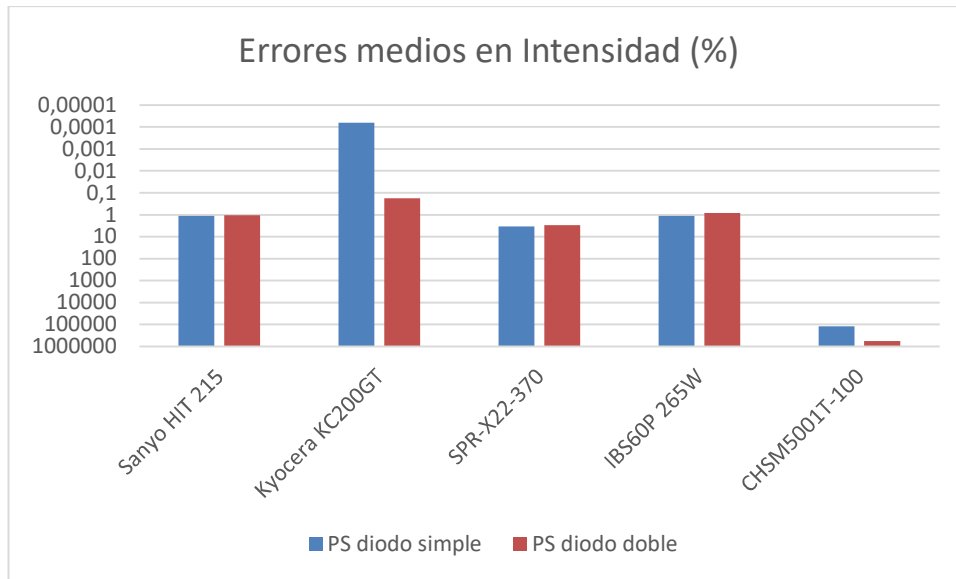


Figura 5.5. Comparación algoritmo PS para los dos modelos.

#### 5.4. Pattern search con solución inicial de GA.

Debido a la gran dependencia que tiene la solución obtenida de la elección del punto base inicial, surge la duda de si es posible mejorar la solución de otro algoritmo tomando como punto inicial su solución final, teniendo así dos algoritmos distintos en serie. Se ha tomado para realizar esta comprobación el algoritmo genético, ya que da una solución bastante buena teniendo aún cierto margen de mejora.

Se ha ejecutado el algoritmo genético y se han tomado los parámetros del modelo de 5 parámetros con todos sus decimales para partir de esta solución como punto base inicial en el *pattern search*. La célula es la SPR-X22-370, la que tenía un mayor error y por ello mejores opciones de mejora, y la solución inicial es:

$a=1.811901252834247$

$R_s=0.010000000001367$

$R_p=9.99998904291371e+02$

$I_0=1.161490145551317e-06$

$I_{pv}=6.660223142044751$

El error para este punto es 0.103753709657511 A, en la Figura 5.6 podemos ver cómo evoluciona este error, que disminuye a 0.103753709657088 A, un cambio mínimo ya que es igual al anterior en los primeros doce decimales. La condición de detención ha sido alcanzar la tolerancia de mallado,  $1e-17$ .

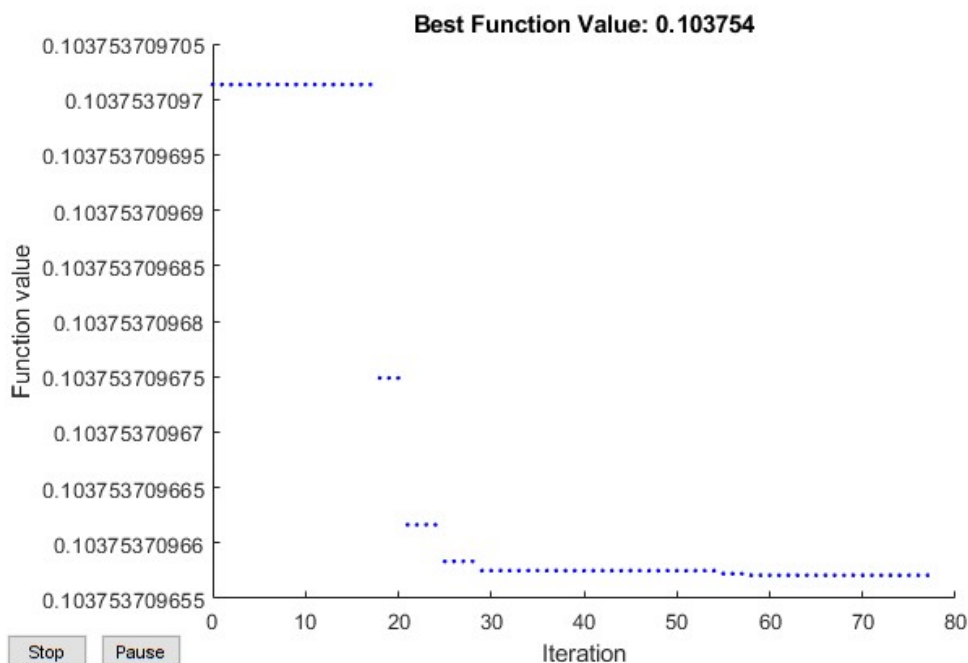


Figura 5.6. Evolución del error del PS al tomar como  $x_0$  la solución del GA, célula SPR-X22-370.

Se ha repetido la prueba con la célula IBS60P-265W, la segunda con mayor error, obteniendo la siguiente solución inicial:

$$a=1.647358725232344$$

$$R_s=0.010000045479541 \text{ ohmios}$$

$$R_p=9.301065933460708e+02 \text{ ohmios}$$

$$I_0=2.519801038161927e-06 \text{ A}$$

$$I_{pv}=8.980138879074788 \text{ A}$$

El error de este punto inicial es 0.044970289419760 A, y su evolución hasta un error de 0.044970262730168 A se ha representado en la Figura 5.7. En este caso sigue siendo un cambio mínimo, apreciándose a partir del decimal número ocho. La condición de detención ha sido la misma que en el caso anterior.

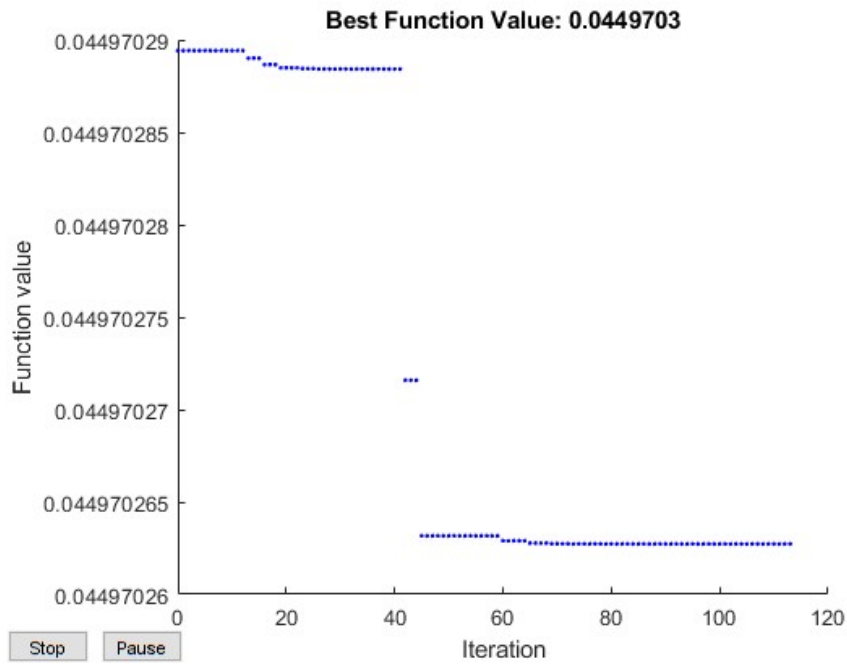


Figura 5.7. Evolución del error del PS al tomar como  $x_0$  la solución del GA, célula IBS60P-265W.

De ambos ejemplos se llega a la conclusión de que sí que es posible mejorar la solución del algoritmo genético y probablemente la de otros algoritmos con el *pattern search*, sin embargo, la disminución del error es muy pequeña. El tiempo empleado en mejorar esta solución es inferior a 2 segundos, ya que se consigue en pocas iteraciones y solo es necesario ejecutar el algoritmo una vez.

# 6 SIMULATED ANNEALING (SA)

---

*Si una pieza de bronce, oro o hierro se rompe en pedazos, el herrero las funde de nuevo.*

*- Sri Gurú Granth Sahib-*

El recocido simulado o *simulated annealing* (SA) es un algoritmo de búsqueda metaheurística utilizado para resolver problemas de optimización global. Está inspirado en el proceso de recocido de cerámicas y aceros, el cual consiste en calentar el material y dejarlo enfriar para permitir una reordenación de sus átomos. Como resultado de esta operación se ablanda el material facilitando su deformación y se disminuye la tensión residual. En el ejemplo de la Figura 6.1 unas piezas de acero están enfriándose a temperatura ambiente en la última etapa del recocido.



Figura 6.1. Recocido de acero. [12]

## 6.1. Descripción del algoritmo.

El algoritmo parte de un punto inicial definido en un espacio vectorial de número de dimensiones igual al número de variables del problema y una temperatura inicial, y genera de forma aleatoria un punto de prueba cuya distancia al punto inicial puede estar definida por la temperatura o la raíz cuadrada de la temperatura, y la dirección en la que se toma el punto es uniformemente aleatoria.

A continuación se evalúa el valor de la función objetivo en el nuevo punto, si es mejor que el anterior se convierte automáticamente en el siguiente punto, si es peor aún puede convertirse en el siguiente con una probabilidad dada por la función de aceptación (5.1), donde  $\Delta$  es la diferencia entre los valores de la función objetivo y  $T$  es la temperatura. Para soluciones peores la probabilidad aumenta si el valor de la nueva función objetivo está cerca del de antes y en las primeras iteraciones en las que la temperatura es mayor, igual que en un recocido real, en el cual es más probable el movimiento de átomos con temperaturas elevadas hacia estados de menor energía.

$$\frac{1}{1 + \exp\left(\frac{\Delta}{T}\right)} \quad (5.1)$$

La temperatura va disminuyendo de forma sistemática, existen varias funciones que pueden definir este enfriamiento, por ejemplo MATLAB ofrece la exponencial (5.2), la rápida (5.3) y la de Boltzmann (5.4). El valor del parámetro de recocido  $k$  es igual al número de iteraciones hasta el recalentamiento.

$$T = T_0 \cdot 0.95^k \quad (5.2)$$

$$T = \frac{T_0}{k} \quad (5.3)$$

$$T = \frac{T_0}{\log(k)} \quad (5.4)$$

Existe un recalentamiento que ocurre tras un número definido de iteraciones, disminuyendo el valor de  $k$  respecto el número de iteraciones resultando en un aumento de la temperatura, lo que simula el recocido en varias etapas que se realiza en ocasiones para obtener mejores propiedades mecánicas.

## 6.2. Aplicación del algoritmo simulated annealing al problema en MATLAB.

El algoritmo se ha probado en los modelos de diodo simple y diodo doble. Se ha definido la misma función objetivo que para los otros algoritmos, minimizar el error de intensidad en los tres puntos de catálogo (cortocircuito, circuito abierto y máxima potencia). El cálculo de la función objetivo se realiza sustituyendo las coordenadas de un punto en la ecuación de la característica I-V de la célula para cada uno de los modelos utilizados.

### 6.2.1 Modelo de diodo simple.

Para este algoritmo en MATLAB es necesario que la distancia entre el límite superior e inferior sea de al menos 0.0002 para hallar la aproximación de la derivada con la diferencia finita en el paso de recalentamiento, por lo que el intervalo utilizado para los otros métodos en  $I_0$  no es suficientemente amplio y se ha tenido que modificar. Los límites son los siguientes:

$$a \in [1, 2] \quad R_s \in [0.01, 1.2] \quad R_p \in [50, 1000]$$

$$I_0 \in [1e-12, 2e-4+1e-12] \quad I_{pv} \in [1, 9]$$

El algoritmo se ha ejecutado diez veces, alterando la solución inicial en cada ejecución para dar una mayor riqueza a la exploración del espacio, aunque los resultados no dependen tanto de esta solución inicial como en el *pattern search*. A diferencia de este último, el SA sí que tiene una cierta aleatoriedad. Las soluciones iniciales se han elegido de forma análoga a como se actuó con el *pattern search*. El valor de la temperatura inicial se ha establecido en 2000 y el máximo de evaluaciones en 1000. La función de enfriamiento elegida ha sido la rápida, y las distancias de los puntos de prueba al inicial están definidas por la raíz cuadrada de la temperatura, aunque con otras funciones y distancias los resultados son similares.

Los errores se han reflejado en la Tabla 6-1.



Tabla 6-1. Errores SA diodo simple.

	Sanyo HIT 215	Kyocera KC200GT	SPR-X22-370	IBS60P 265W	CHSM5001T-100
Isc catálogo (A)	5.61	8.21	6.66	8.98	1.499
Impp catálogo (A)	5.13	7.61	6.26	8.58	1.188
Isc modelo (A)	5.550	8.386	6.612	8.850	1.463
Impp modelo (A)	5.411	7.394	6.291	8.550	1.211
loc modelo (A)	-0.071	-0.052	0.070	0.675	0
Error medio en intensidad	2.68%	1.95%	0.79%	3.24%	1.83%
tiempo (s) 10 iter.	75	83	76	67	64

Los errores son estables, siendo el menor de 0.79% y el mayor 3.24%. Los tiempos son elevados, ya que el número de iteraciones en cada ejecución puede ser de hasta 1000. Los parámetros para estos errores son los de la Tabla 6-2.

Tabla 6-2. Parámetros SA diodo simple.

	Sanyo HIT 215	Kyocera KC200GT	SPR-X22-370	IBS60P 265W	CHSM5001T- 100
a	1.93	1.56	1.93	1.81	2.00
Rs (ohmios)	0.303	0.056	0.010	0.010	0.164
Rp (ohmios)	832	193	215	998	1000
IO (A)	3.00E-06	2.00E-06	3.00E-06	1.00E-05	1.98E-08
Ipv (A)	5.8	8.2	7.2	9.0	0.0

En la Figura 6.2 se ha representado la evolución del valor de la función objetivo en cada iteración en una de las ejecuciones del algoritmo. Se puede apreciar que la mejora es a saltos, con muchas iteraciones en las que no se logra encontrar puntos mejores. A pesar de agotar prácticamente las mil iteraciones la solución dada tiene un error elevado, de 0.325 A.

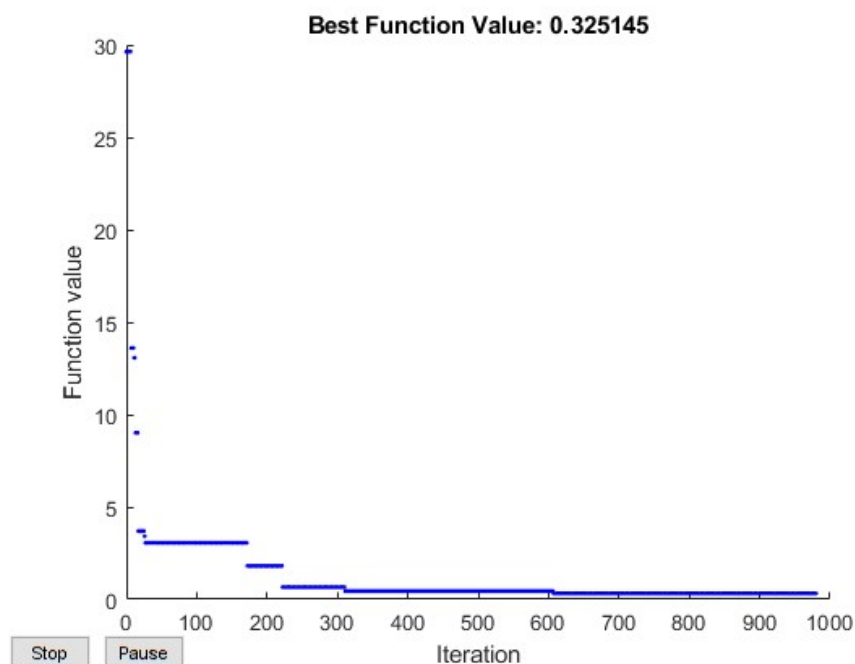


Figura 6.2. Evolución del SA.

A continuación, se ha representado en la Figura 6.3 la curva I-V de la célula con mayor error, el cual es visible

principalmente en el punto de circuito abierto, en el que la intensidad es 0.675 A cuando debería ser nula. También se ha representado el caso más favorable (Figura 6.4), el cual tiene el menor error, aunque la desviación de los puntos es claramente visible.

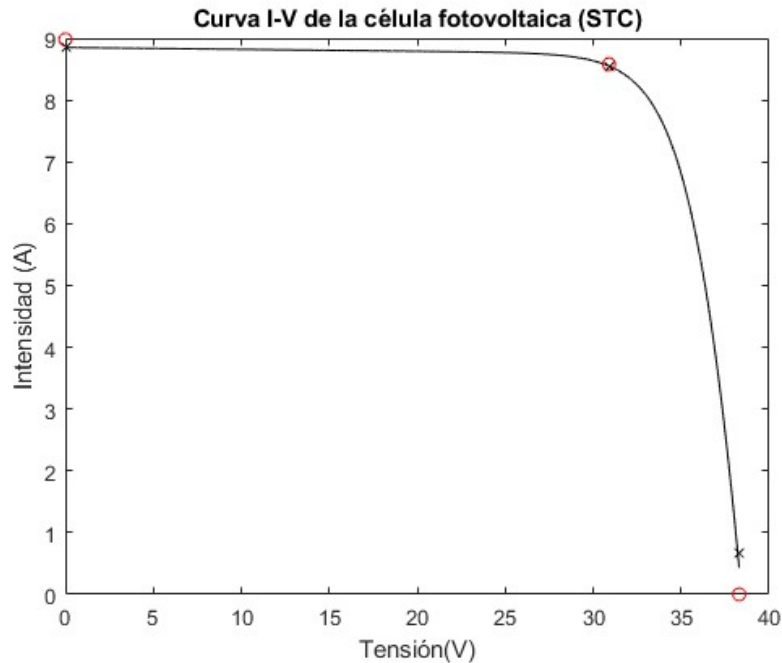


Figura 6.3. Curva I-V IBS60P 265W. (SA5)

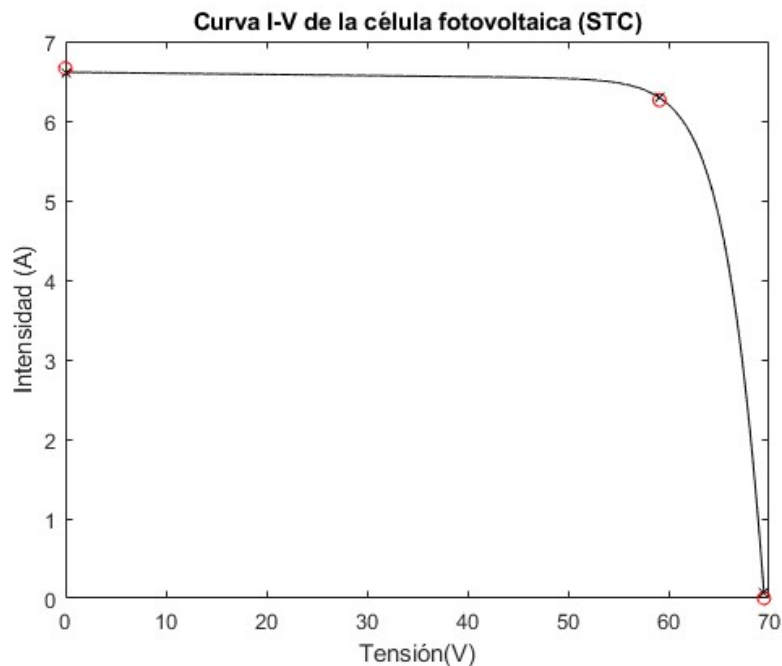


Figura 6.4. Curva I-V SPR-X22-370. (SA5)

## 6.2.2 Modelo de diodo doble.

El problema con el intervalo mínimo de 0.0002 para hallar la aproximación de la derivada con la diferencia finita sigue existiendo, por lo que los límites se han modificado para este modelo también, siendo los siguientes:

$$\begin{array}{llll}
 a \in [1, 2] & R_S \in [0.01, 1.2] & R_P \in [50, 1000] & \\
 I_0 \in [1e-12, 2e-4+1e-12] & I_{PV} \in [1, 9] & a_2 \in [1, 2] & I_{02} \in [1e-12, 2e-4+1e-12]
 \end{array}$$

Las opciones del algoritmo se han mantenido iguales a las utilizadas en el modelo de diodo simple, la temperatura inicial es 2000 y el máximo de evaluaciones es 1000. La función de enfriamiento escogida es la rápida, y las distancias de los puntos de prueba al inicial están definidas por la raíz cuadrada de la temperatura. Los errores se han introducido en la Tabla 6-3.

Tabla 6-3. Errores SA diodo doble.

	Sanyo HIT 215	Kyocera KC200GT	SPR-X22-370	IBS60P 265W	CHSM5001T- 100
Isc catálogo (A)	5.61	8.21	6.66	8.98	1.499
Impv catálogo (A)	5.13	7.61	6.26	8.58	1.188
Isc modelo (A)	5.625	7.962	6.750	8.759	1.502
Impv modelo (A)	5.491	7.298	6.317	7.954	1.155
loc modelo (A)	5.38E-01	1.43E-01	4.88E-01	4.37E-01	-9.21E-02
Error medio en intensidad (%)	5.94%	3.08%	3.38%	4.99%	3.59%
tiempo (s) 10 iter.	89	64	66	64	68

En comparación con el modelo de diodo simple los errores son muy superiores, al parecer el hecho de complicar el modelo perjudica el correcto funcionamiento del algoritmo. En cuanto a los tiempos siguen siendo elevados, aunque no se aprecia un incremento respecto al caso anterior. Los parámetros del modelo para los errores obtenidos son los de la Tabla 6-4.

Tabla 6-4. Parámetros SA diodo doble.

	Sanyo HIT 215	Kyocera KC200GT	SPR-X22-370	IBS60P 265W	CHSM5001T -100
a	1.61	1.29	2.00	1.62	1.93
Rs (ohmios)	0.21	0.04	0.02	0.02	0.42
Rp (ohmios)	624	821	907	733	243
I0 (A)	1.00E-12	3.97E-09	7.70E-08	6.77E-08	1.11E-12
Ipv (A)	5.63	7.96E+00	6.75	8.76	1.50
a2	1.07	1.84	1.47	1.99	1.97
I02 (A)	2.47E-11	1.85E-05	2.71E-08	3.02E-05	1.20E-12

Para visualizar la magnitud de los errores se ha representado la curva I-V de la célula Sanyo HIT 215 en la Figura 6.5, donde solo se consiguen puntos similares en el punto de cortocircuito. El punto de circuito abierto tiene una diferencia de más de medio Amperio.

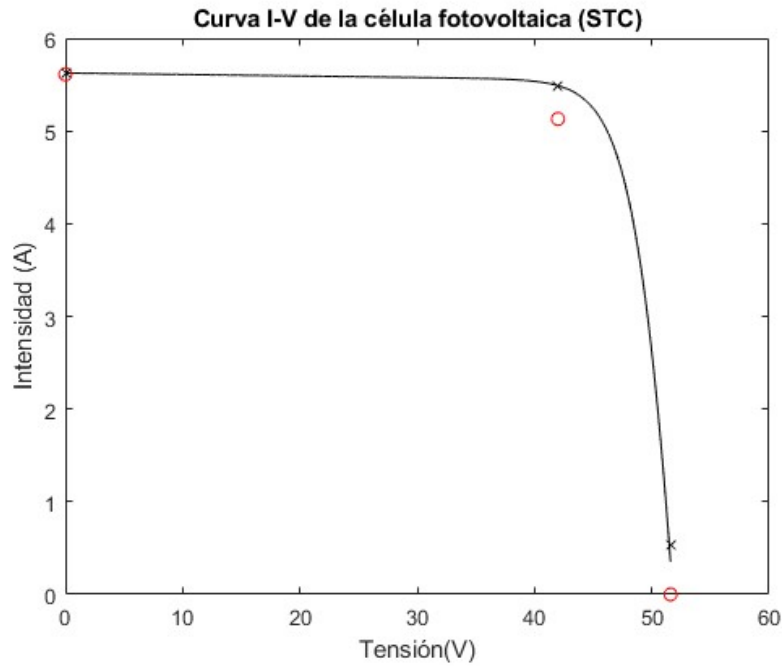


Figura 6.5. Curva I-V de la célula Sanyo HIT 215. (SA7)

### 6.3. Conclusiones de la aplicación del algoritmo SA al problema.

A la vista de los resultados obtenidos, se deduce que no es conveniente emplear el algoritmo de *simulated annealing* a este problema, tanto con el modelo de diodo simple como con el modelo de diodo doble, siendo el último el peor de los dos. Los errores son inaceptablemente altos y los tiempos más elevados que los empleados en otros algoritmos con mejores resultados.

Las soluciones de esta implementación del algoritmo son fruto de unas opciones determinadas, que comparadas con otras opciones probadas dan mejores o similares resultados para este problema en concreto, lo que no excluye que con modificaciones del algoritmo se puedan alcanzar soluciones mejores hipotéticamente.

# 7 COMPARACIÓN DE LOS ALGORITMOS PSO, GA, SA Y PS.

En este capítulo se hace una recopilación de los resultados obtenidos de la aplicación de los algoritmos PSO, GA, SA, y PS para la obtención de la curva I-V a partir de los puntos de cortocircuito, circuito abierto y máxima potencia del catálogo. El objetivo es comparar los algoritmos ya desarrollados en capítulos anteriores en profundidad para determinar cuál es el más apropiado.

A modo de resumen de los resultados de cada algoritmo para cada uno de los cinco paneles solares probados y los modelos de diodo simple y diodo doble se han recogido los errores de intensidad en tanto por ciento en la Tabla 7-1. Además, se ha calculado en la última columna la media aritmética de los errores. Con un primer análisis se observa que los mejores resultados son los obtenidos mediante el algoritmo de enjambre de partículas, con varios errores nulos y la menor media de todas si se toma el modelo de diodo doble.

Tabla 7-1. Errores en % por célula, algoritmo y modelo.

Módulo PV	Sanyo HIT 215	Kyocera KC200GT	SPR-X22- 370	IBS60P 265W	CHSM5001T- 100	Media
GA 1 diodo	6.09E-07	1.94E-06	1.79E+00	4.88E-01	1.70E-05	4.55E-01
GA 2 diodos	1.34E-06	1.13E-07	9.87E-01	5.37E-01	5.04E-05	3.05E-01
PS 1 diodo	1.11E+00	6.29E-05	3.45E+00	1.11E+00	1.23E+05	2.46E+04
PS 2 diodos	1.04E+00	1.80E-01	3.00E+00	8.41E-01	5.76E+05	1.15E+05
PSO 1 diodo	0.00E+00	0.00E+00	1.49E+00	8.02E-01	6.23E-15	4.58E-01
PSO 2 diodos	0.00E+00	0.00E+00	0.00E+00	7.66E-10	0.00E+00	1.53E-10
SA 1 diodo	2.68E+00	1.95E+00	7.93E-01	3.24E+00	1.83E+00	2.10E+00
SA 2 diodos	5.94E+00	3.08E+00	3.38E+00	4.99E+00	3.59E+00	4.20E+00

Estos errores se han representado en la gráfica de la Figura 7.1 en escala logarítmica debido a las grandes diferencias entre errores. Para poder colocar columnas en los errores nulos, estos se han representado como 1e-15. El eje Y está invertido, por lo que las columnas más altas indican menores errores y por tanto, mejores resultados.

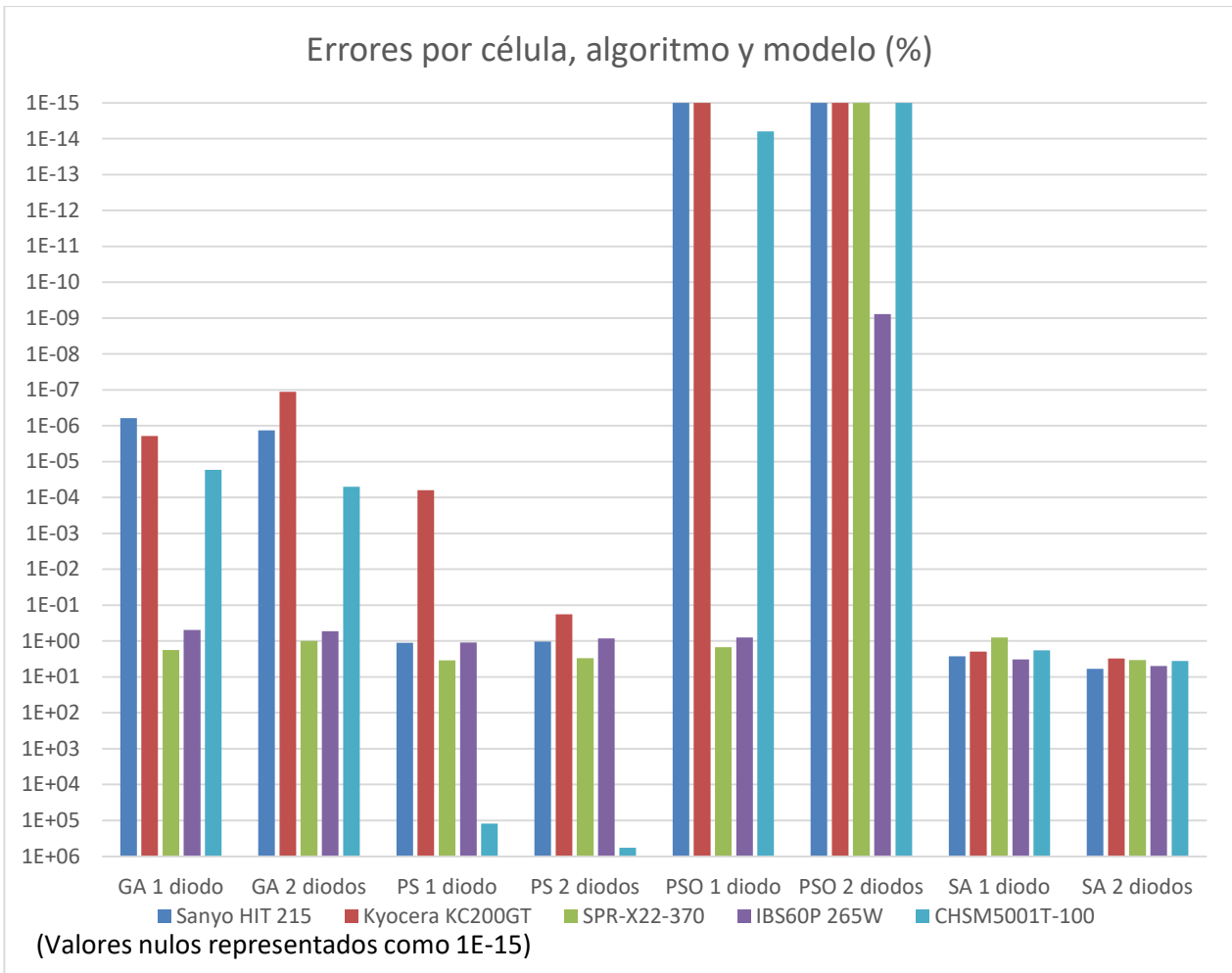


Figura 7.1. Errores en % por célula, algoritmo y modelo.

Como se vio de primeras en la Tabla 7-1, el PSO es claramente superior, con dos ceros para el modelo de diodo simple y cuatro ceros para el modelo de diodo doble. Le sigue el GA, con errores inferiores a 0.00001% en las células de los paneles Sanyo HIT 215 y Kyocera KC200GT para ambos modelos, y un error inferior a 0.0001% para el panel CHSM5001T-100. Los algoritmos PS y SA consiguen unos errores similares, entre el 1% y el 10% en general, con algún error inferior en alguna ocasión.

Es destacable la incapacidad del *pattern search* de alcanzar un error válido en el caso de la célula de película fina para ambos modelos. Además, queda patente sobre todo con el PSO y el GA, que se consiguen mejores resultados en los primeros dos paneles (el Sanyo HIT 215 y Kyocera KC200GT), esto es debido a que son paneles cuyo buen comportamiento ha hecho que sean elegidos para probar los algoritmos en varios estudios. Los otros tres paneles se han escogido de forma aleatoria para probar la reproducibilidad y la repetibilidad de los estudios, ambas pilares del método científico.

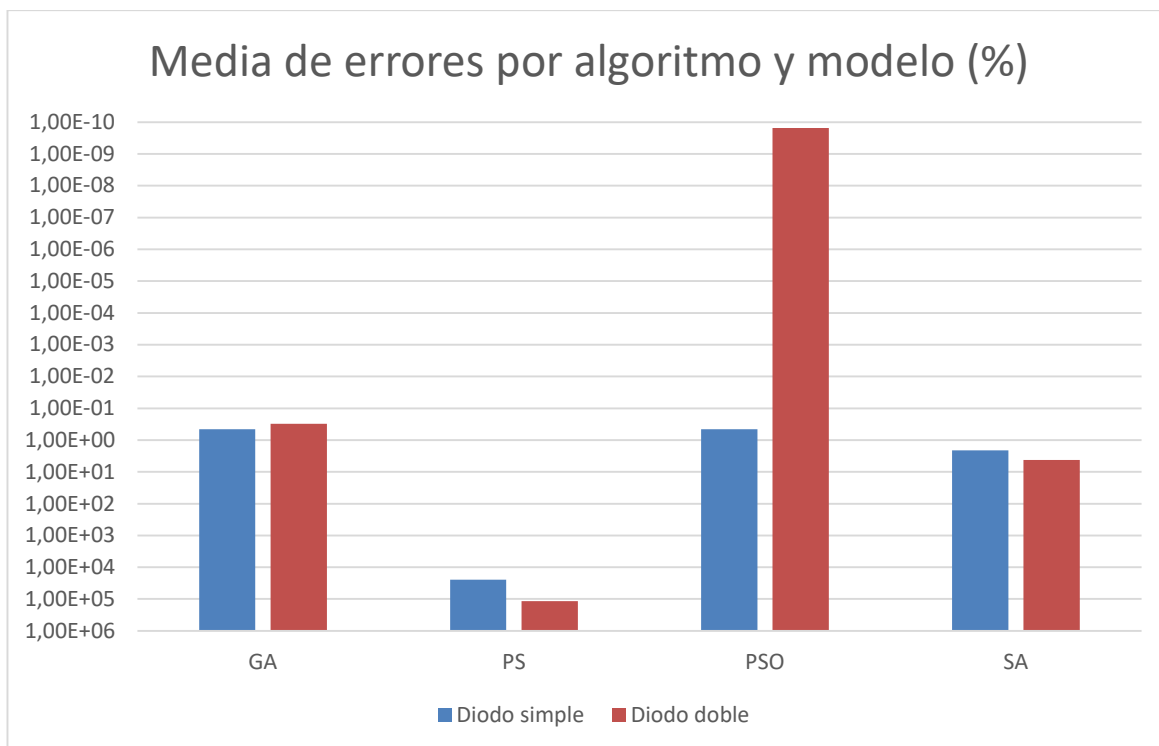


Figura 7.2. Media de errores por algoritmo y modelo.

En la Figura 7.2 se ha representado las medias de los errores de los paneles en una gráfica de características similares a las de la Figura 7.1. Aunque parezca que las conclusiones que se pueden obtener de esta gráfica difieren de las anteriores, esto es debido a que un solo valor fuera de rango distorsiona completamente la media. Esto ocurre por ejemplo para el PS, que tiene un desempeño parecido al SA en cuatro paneles, pero para la célula de película fina el error es superior al 100000%, lo que eleva la media enormemente. Algo similar ocurre en el caso del PSO para un diodo, que parece tener unos resultados similares a los del GA.

En cuanto a los tiempos de ejecución de los algoritmos, en la Tabla 7-2 se han recogido los tiempos de ejecución por algoritmo, modelo y panel de las diez ejecuciones del algoritmo. Estos tiempos deben ser orientativos, ya que la prioridad en el estudio ha sido minimizar el error, por lo que relajando la precisión o el número de ejecuciones se pueden obtener tiempos bastante inferiores. Además, no todos los algoritmos son igual de robustos, por lo que de las diez ejecuciones unos alcanzan más soluciones adecuadas que otros.

Tabla 7-2. Tiempos de ejecución (s).

Módulo PV	Sanyo HIT 215	Kyocera KC200GT	SPR-X22-370	IBS60P 265W	CHSM5001T- 100	Media (segundos)
GA 1 diodo	15	25	17	18	20	19
GA 2 diodos	28	31	36	65	71	46
PS 1 diodo	35	34	23	22	24	28
PS 2 diodos	45	29	40	21	21	31
PSO 1 diodo	47	45	65	47	39	49
PSO 2 diodos	59	65	78	48	43	59
SA 1 diodo	75	83	76	67	64	73
SA 2 diodos	89	64	66	64	68	70

En la Figura 7.3 se ha graficado las medias de los tiempos en segundos de cada algoritmo y modelo, se puede ver que exceptuando el SA, en el resto de casos el modelo de diodo doble tiene un mayor coste computacional, ya que se aumenta de 5 a 7 el número de parámetros. Comparando los dos algoritmos de mejor precisión (PSO y GA), se aprecia que el GA es más rápido, una ventaja que puede hacerlo favorito si una precisión menor es

válida.

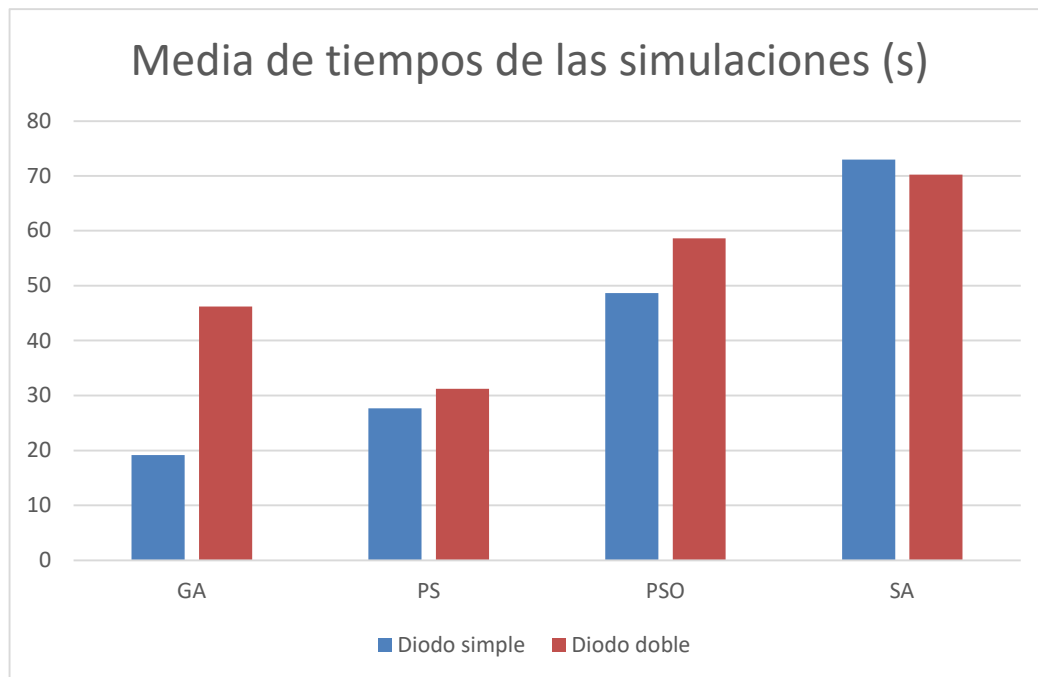


Figura 7.3. Media de tiempos de los algoritmos.

En resumen, para la obtención de un modelo de célula fotovoltaica a partir de los tres puntos del catálogo, el algoritmo idóneo es el de enjambre de partículas empleando el modelo de diodo doble, seguido del modelo de diodo simple para el mismo algoritmo. El algoritmo genético también tiene un buen comportamiento para algunos paneles, sin ser tan efectivo como el PSO, aunque más rápido. Los algoritmos *pattern search* y *simulated annealing* es mejor evitarlos para este problema.



# 8 ESTUDIO DE LOS ALGORITMOS CON DISPONIBILIDAD DE DATOS EXPERIMENTALES

Con solo tres puntos de catálogo se pueden obtener los parámetros del modelo de la célula fotovoltaica deseado de forma satisfactoria con algunos métodos mejor que con otros, y en ocasiones puede ser suficiente. Sin embargo, a veces puede ser necesario que el modelo se ajuste en un número mayor de puntos, para lo que sería necesario disponer de una serie de datos experimentales en los que poder evaluar el ajuste de la curva.

Encontrar puntos distintos de los tres puntos notables que aparecen siempre en el catálogo es complicado, ya que los fabricantes no ponen a disposición del público general los resultados de sus pruebas. Tampoco hay datos experimentales en las editoriales de literatura científica como Elsevier o en las publicaciones de IEEE consultadas. El único artículo que ofrece los datos en los que se basa es uno de T. Easwarakhanthan de 1986 [3]. Ha sido citado en numerosas ocasiones por autores posteriores para demostrar el comportamiento de un determinado algoritmo, algunos de ellos bien entrada la segunda década del siglo presente, lo que evidencia la necesidad de datos más actuales.

## 8.1. Artículo de Easwarakhanthan, datos experimentales.

En el artículo de Easwarakhanthan se testea una célula solar de silicio comercial de 57 mm de R.T.C. France a 33 °C, de la que se obtienen 26 pares de datos intensidad-tensión. El autor utiliza estos datos para encontrar los parámetros del modelo de diodo simple que definan una curva I-V que pase por los puntos con el menor error posible. El algoritmo utilizado es un algoritmo no lineal de mínimos cuadrados basado en el método de Newton modificado con el parámetro de Levenberg.

La función objetivo a minimizar es el sumatorio de la diferencia en cada punto entre la intensidad experimental y la calculada por el modelo para la tensión experimental en dicho punto, cuya expresión es la (6.1), donde  $\theta$  es el vector de parámetros ( $a$ ,  $R_s$ ,  $R_p$ ,  $I_0$ ,  $I_{pv}$ ).

$$S(\theta) = \sum_{i=1}^{26} [I_i - I(V_i, \theta)]^2 \quad (6.1)$$

Para el cálculo de  $I(V_i, \theta)$  el autor emplea la expresión del diodo simple (6.2), en la que para facilitar el cálculo simplifica sustituyendo las intensidades a la derecha de la igualdad por la experimental.

$$I = I_{pv} - I_0 \left[ \exp\left(\frac{q(V + R_s I)}{aKT}\right) - 1 \right] - \frac{V + R_s I}{R_p} \quad (6.2)$$

Los parámetros dados por Easwarakhanthan en el artículo son los de la Tabla 8-1.

Tabla 8-1. Parámetros del modelo de Easwarakhanthan.

a	1.4837
Rs	0.0364
Rp	53.76
l0	3.22E-07
l <sub>pv</sub>	0.7608

En la Tabla 8-2 se han recogido las intensidades y tensiones experimentales en cada uno de los puntos, así como las intensidades calculadas por el autor en cada punto, las cuales resultan sumamente parecidas a las experimentales. Además se han calculado las intensidades para los parámetros de la Tabla 8-1 tomando la aproximación del autor y las intensidades resultantes de hallar el valor real en cada punto de la intensidad sin ninguna aproximación, para lo que se ha usado el comando *fsolve* de MATLAB, dando un error máximo del *fsolve* de  $4.6 \cdot 10^{-7}$ , despreciable comparado con el error en intensidad.

Tabla 8-2. Tensiones e intensidades de Easwarakhanthan y a partir de su modelo.

N.º	V exp	I exp	I calculada por el autor	Error absoluto	I fsolve	Error absoluto	I aproximación	Error absoluto
1	-0.2060	0.7640	0.7641	0.0001	0.7641	0.0001	0.7641	0.0001
2	-0.1290	0.7620	0.7627	0.0007	0.7627	0.0007	0.7627	0.0007
3	-0.0590	0.7605	0.7614	0.0009	0.7614	0.0009	0.7614	0.0009
4	0.0057	0.7605	0.7602	0.0003	0.7602	0.0003	0.7602	0.0003
5	0.0646	0.7600	0.7591	0.0009	0.7591	0.0009	0.7591	0.0009
6	0.1185	0.7590	0.758	0.001	0.7581	0.0009	0.7581	0.0009
7	0.1678	0.7570	0.7571	0.0001	0.7571	0.0001	0.7571	0.0001
8	0.2132	0.7570	0.7561	0.0009	0.7562	0.0008	0.7562	0.0008
9	0.2545	0.7555	0.7551	0.0004	0.7551	0.0004	0.7551	0.0004
10	0.2924	0.7540	0.7537	0.0003	0.7537	0.0003	0.7537	0.0003
11	0.3269	0.7505	0.7514	0.0009	0.7515	0.0010	0.7515	0.0010
12	0.3585	0.7465	0.7473	0.0008	0.7475	0.0010	0.7475	0.0010
13	0.3873	0.7385	0.7401	0.0016	0.7404	0.0019	0.7404	0.0019
14	0.4137	0.7280	0.7274	0.0006	0.7279	0.0001	0.7279	0.0001
15	0.4373	0.7065	0.707	0.0005	0.7079	0.0014	0.7080	0.0015
16	0.4590	0.6755	0.6754	0.0001	0.6769	0.0014	0.6770	0.0015
17	0.4784	0.6320	0.6308	0.0012	0.6334	0.0014	0.6336	0.0016
18	0.4960	0.5730	0.5722	0.0008	0.5758	0.0028	0.5763	0.0033
19	0.5119	0.4990	0.4994	0.0004	0.5046	0.0056	0.5058	0.0068
20	0.5265	0.4130	0.4134	0.0004	0.4200	0.0070	0.4222	0.0092
21	0.5398	0.3165	0.3172	0.0007	0.3252	0.0087	0.3286	0.0121
22	0.5521	0.2120	0.2125	0.0005	0.2215	0.0095	0.2262	0.0142
23	0.5633	0.1035	0.1026	0.0009	0.1135	0.0100	0.1194	0.0159
24	0.5736	-0.0100	-0.0097	0.0003	0.0028	0.0128	0.0116	0.0216
25	0.5833	-0.1230	-0.1241	0.0011	-0.1112	0.0118	-0.1018	0.0212
26	0.5900	-0.2100	-0.2093	0.0007	-0.1952	0.0148	-0.1823	0.0277

Como se puede comprobar, la columna de intensidades calculadas con la aproximación es diferente a la de los datos del artículo, a lo que no se ha encontrado explicación más allá de que los parámetros no tengan suficientes decimales para que los resultados sean los mismos, aunque la diferencia en los puntos de circuito abierto (últimos puntos) es considerable. En la Tabla 8-3 se ha comparado el error medio y el error cuadrático medio de las intensidades dadas por el autor y las calculadas con sus parámetros sin aproximación y con ella. Se concluye que los resultados del autor son mejores, aunque no queda claro por qué son diferentes a los calculados con sus

parámetros. Eliminar la aproximación es beneficioso para reducir el error, además de ser lo estrictamente correcto.

Tabla 8-3. Comparación de errores.

	autor	fsolve	aproximación
MAE	0.00066	0.00372	0.00561
RMSE	0.00075	0.00115	0.00190

## 8.2. Resultados de los algoritmos para el modelo de diodo simple.

Para cuantificar el ajuste de diferentes algoritmos en numerosos puntos de la curva I-V se ha resuelto el problema de minimización con el algoritmo genético (GA), el algoritmo de enjambre de partículas (PSO), el *pattern search* (PS) y el *simulated annealing* (SA).

La expresión a minimizar es la (6.1), en la que se aproxima el punto con las condiciones de testeo para poder evaluar rápidamente el valor de la función objetivo, algo fundamental en este tipo de algoritmos. Una vez se tienen los parámetros del modelo para la mejor solución del algoritmo se calculan los puntos con *fsolve*, hallando el valor de I para cada valor de V del experimento, y se compara con el valor de I del experimento. El mayor valor del error dado por *fsolve* es varios órdenes de magnitud menor que la diferencia entre la intensidad experimental y la calculada del punto con menor error.

Los parámetros de cada algoritmo son los siguientes:

- **GA**

Límites:

$a \in [1, 2]$                        $R_S \in [0.01, 1.2]$                        $R_P \in [10, 1000]$

$I_0 \in [1e-12, 1e-5]$        $I_{PV} \in [0.3, 9]$

Tamaño de población: 50

Individuos resultados de cruce: 35

Individuos resultados de mutación: 12

Élites que sobreviven entre iteraciones: 3

Iteraciones máximas: 1000

Número de iteraciones sin cambio en la función objetivo para la detención del algoritmo: 15

Función de selección: *remainder*

Función de cruce: heurística

- **PSO**

Límites:

$a \in [1, 2]$                        $R_S \in [0.01, 1.2]$                        $R_P \in [10, 1000]$

$I_0 \in [1e-12, 1e-5]$        $I_{PV} \in [0.3, 9]$

Función de creación: uniforme

Rango del factor de inercia: [0.1, 1.1]

Constante de atracción  $\gamma_1$ : 1.49

Constante de atracción  $\gamma_2$ : 1.49

Miembros del enjambre: 100

Iteraciones máximas: 1000

Número de iteraciones sin cambio en la función objetivo para la detención del algoritmo: 20

- **PS**

Límites:

$a \in [1, 2]$                        $R_S \in [0.01, 1.2]$                        $R_P \in [10, 1000]$

$I_0 \in [1e-12, 1e-5]$        $I_{PV} \in [0.3, 9]$

Mallado inicial: 1

Mallado máximo:  $1e-17$

Iteraciones máximas: 500

Factor de expansión del mallado: 2

Factor de disminución del mallado: 0.5

- **SA**

Límites:

$a \in [1, 2]$                        $R_S \in [0.01, 1.2]$                        $R_P \in [10, 1000]$

$I_0 \in [1e-12, 2e-4+1e-12]$        $I_{PV} \in [0.3, 9]$

Temperatura inicial: 1000

Función de enfriamiento: rápida

Función de distancia al punto anterior: cuadrática

Número máximo de evaluaciones: 1000

Los algoritmos se han ejecutado diez veces, tomando en cada caso la mejor solución de todas las ejecuciones. Los algoritmos GA y PSO no necesitan solución inicial, por lo que todas las ejecuciones dependen solo de la aleatoriedad, mientras que para los algoritmos SA y PS se han tomado soluciones iniciales diferentes para cada ejecución.

En la Tabla 8-4 se recogen los parámetros del modelo para la mejor solución obtenida.

Tabla 8-4. Parámetros modelo diodo simple.

	SA	GA	PSO	PS
a	1.903	1.489	1.503	1.604
Rs	0.010	0.036	0.035	0.032
Rp	697	57	61	109
I0	7.9E-06	3.5E-07	4.0E-07	1.0E-06
Ipv	0.719	0.761	0.761	0.760

Para ver la efectividad de los algoritmos se han colocado en una tabla los 26 puntos experimentales y se han calculado las intensidades del modelo para todas las tensiones experimentales, comparándola punto a punto con las intensidades experimentales, y hallando el error absoluto entre ambas. Los resultados son los de la Tabla 8-5.

Tabla 8-5. Errores por punto modelo diodo simple.

Datos experimentales			SA		GA		PSO		PS	
N.º	V exp	I exp	I modelo	Error	I modelo	Error	I modelo	Error	I modelo	Error
1	-0.2057	0.7640	0.8538	0.0898	0.7629	0.0011	0.7622	0.0018	0.7607	0.0033
2	-0.1291	0.7620	0.8537	0.0917	0.7620	0.0000	0.7616	0.0004	0.7606	0.0014
3	-0.0588	0.7605	0.8536	0.0931	0.7612	0.0007	0.7610	0.0005	0.7606	0.0001
4	0.0057	0.7605	0.8535	0.0930	0.7604	0.0001	0.7605	0.0000	0.7605	0.0000
5	0.0646	0.7600	0.8534	0.0934	0.7597	0.0003	0.7600	0.0000	0.7604	0.0004
6	0.1185	0.7590	0.8532	0.0942	0.7590	0.0000	0.7595	0.0005	0.7603	0.0013
7	0.1678	0.7570	0.8529	0.0959	0.7583	0.0013	0.7590	0.0020	0.7600	0.0030
8	0.2132	0.7570	0.8521	0.0951	0.7576	0.0006	0.7584	0.0014	0.7595	0.0025
9	0.2545	0.7555	0.8507	0.0952	0.7566	0.0011	0.7575	0.0020	0.7584	0.0029
10	0.2924	0.7540	0.8478	0.0938	0.7551	0.0011	0.7559	0.0019	0.7563	0.0023
11	0.3269	0.7505	0.8428	0.0923	0.7524	0.0019	0.7529	0.0024	0.7524	0.0019
12	0.3585	0.7465	0.8344	0.0879	0.7475	0.0010	0.7477	0.0012	0.7457	0.0008
13	0.3873	0.7385	0.8209	0.0824	0.7391	0.0006	0.7386	0.0001	0.7349	0.0036
14	0.4137	0.7280	0.8004	0.0724	0.7248	0.0032	0.7236	0.0044	0.7180	0.0100
15	0.4373	0.7065	0.7717	0.0652	0.7028	0.0037	0.7011	0.0054	0.6937	0.0128
16	0.4590	0.6755	0.7324	0.0569	0.6701	0.0054	0.6680	0.0075	0.6597	0.0158
17	0.4784	0.6320	0.6827	0.0507	0.6255	0.0065	0.6235	0.0085	0.6154	0.0166
18	0.4960	0.5730	0.6218	0.0488	0.5675	0.0055	0.5662	0.0068	0.5596	0.0134
19	0.5119	0.4990	0.5506	0.0516	0.4967	0.0023	0.4964	0.0026	0.4923	0.0067
20	0.5265	0.4130	0.4689	0.0559	0.4130	0.0000	0.4138	0.0008	0.4130	0.0000
21	0.5398	0.3165	0.3789	0.0624	0.3189	0.0024	0.3207	0.0042	0.3230	0.0065
22	0.5521	0.2120	0.2810	0.0690	0.2153	0.0033	0.2176	0.0056	0.2223	0.0103
23	0.5633	0.1035	0.1785	0.0750	0.1065	0.0030	0.1087	0.0052	0.1141	0.0106
24	0.5736	-0.0100	0.0724	0.0824	-0.0061	0.0039	-0.0048	0.0052	-0.0006	0.0094
25	0.5833	-0.1230	-0.0383	0.0847	-0.1231	0.0001	-0.1236	0.0006	-0.1230	0.0000
26	0.5900	-0.2100	-0.1210	0.0890	-0.2100	0.0000	-0.2124	0.0024	-0.2161	0.0061

A primera vista se aprecia que los resultados de unos algoritmos son mejores que los de otros, para tener un solo valor como comparación se ha calculado la media de los errores. También se ha calculado el error cuadrático medio (RMSE). Ambas magnitudes para cada uno de los algoritmos se han recogido en la Tabla 8-6. Con los resultados se puede concluir que el mejor algoritmo en este caso es el GA, seguido de cerca por el PSO. Los menos efectivos son el PS y el SA, siendo este último muy inferior a los otros tres.

Tabla 8-6. MAE y RMSE de los algoritmos para el modelo de diodo simple.

	SA	GA	PSO	PS
MAE	0.0793	0.0019	0.0028	0.0054
RMSE	0.0809	0.0026	0.0038	0.0075

En la Figura 8.1, la Figura 8.2, la Figura 8.3 y la Figura 8.4 se han representado respectivamente las curvas resultantes de los algoritmos GA, PSO, PS y SA en negro. Los puntos experimentales por los que debe pasar se han dibujado con círculos rojos, mientras que los puntos del modelo para las mismas tensiones se han marcado con cruces negras. Se puede ver claramente la correlación entre los errores medios y las desviaciones de las curvas, siendo el algoritmo SA el menos adecuado.

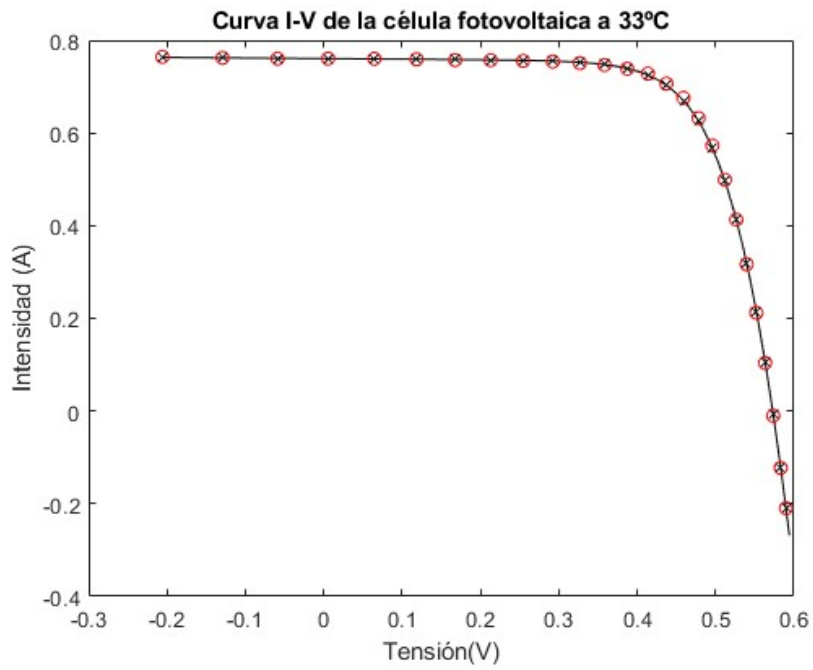


Figura 8.1. Curva I-V GA

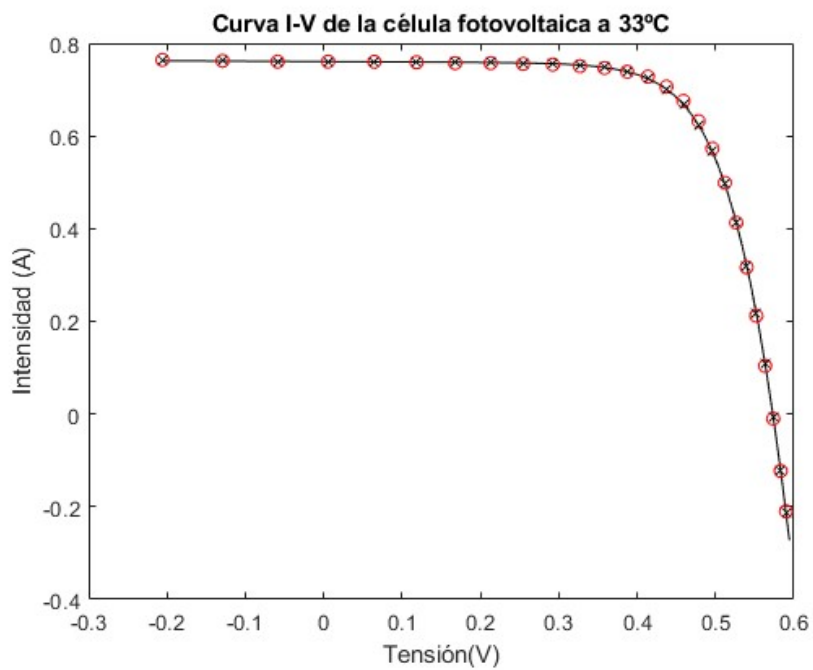


Figura 8.2. Curva I-V PSO

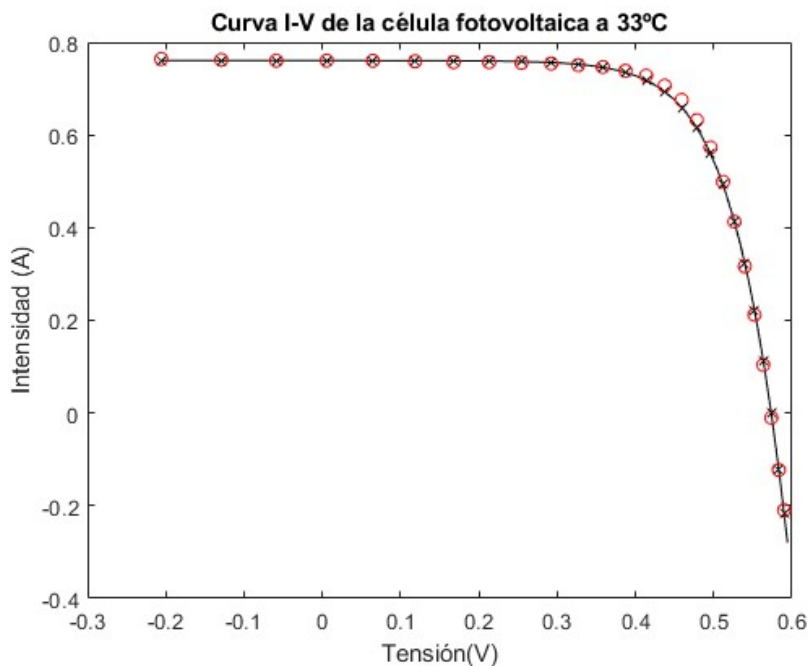


Figura 8.3. Curva I-V PS

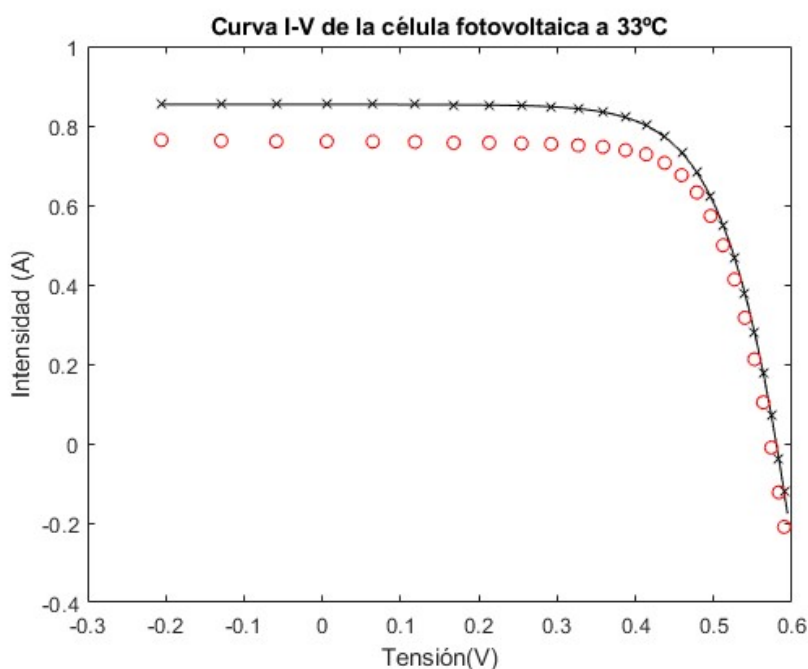


Figura 8.4. Curva I-V SA.

### 8.3. Resultados de los algoritmos para el modelo de diodo doble.

Es interesante examinar si el modelo de diodo doble es capaz de superar los resultados obtenidos con el modelo de diodo simple, para ello se han repetido las pruebas realizadas con este nuevo modelo. Se ha alterado únicamente el número de iteraciones en algunos algoritmos y se han añadido los dos nuevos parámetros relativos al segundo diodo con los límites de  $a_2$  e  $I_{02}$  iguales en cada caso a los de  $a$  e  $I_0$ . En la Tabla 8-7 se recogen los parámetros de la mejor solución de diez ejecuciones para cada algoritmo.

Tabla 8-7. Parámetros modelo diodo doble.

	SA	GA	PSO	PS
a	1.763	1.505	1.996	1.616
Rs	0.014	0.034	0.032	0.030
Rp	797	83	65	80
I0	2.1E-06	3.7E-07	5.7E-10	1.0E-06
Ipv	0.699	0.761	0.760	0.761
a2	1.384	1.968	1.569	1.961
I02	1.0E-12	1.3E-06	7.3E-07	1.0E-06

Se han calculado, como en el caso anterior, las intensidades del modelo punto a punto y se han comparado con las experimentales calculando el valor absoluto de la diferencia entre ambas, lo que constituye el error. Los datos se han recogido en la Tabla 8-8.

Tabla 8-8. Errores modelo diodo doble.

Datos experimentales			SA		GA		PSO		PS	
N.º	V exp	I exp	I modelo	Error	I modelo	Error	I modelo	Error	I modelo	Error
1	-0.2057	0.7640	0.5772	0.1868	0.7629	0.0011	0.7628	0.0012	0.7607	0.0033
2	-0.1291	0.7620	0.5771	0.1849	0.7620	0.0000	0.7620	0.0000	0.7606	0.0014
3	-0.0588	0.7605	0.5771	0.1834	0.7612	0.0007	0.7612	0.0007	0.7606	0.0001
4	0.0057	0.7605	0.5770	0.1835	0.7604	0.0001	0.7605	0.0000	0.7605	0.0000
5	0.0646	0.7600	0.5769	0.1831	0.7597	0.0003	0.7598	0.0002	0.7604	0.0004
6	0.1185	0.7590	0.5768	0.1822	0.7590	0.0000	0.7592	0.0002	0.7602	0.0012
7	0.1678	0.7570	0.5765	0.1805	0.7583	0.0013	0.7585	0.0015	0.7600	0.0030
8	0.2132	0.7570	0.5761	0.1809	0.7576	0.0006	0.7578	0.0008	0.7594	0.0024
9	0.2545	0.7555	0.5750	0.1805	0.7567	0.0012	0.7568	0.0013	0.7582	0.0027
10	0.2924	0.7540	0.5730	0.1810	0.7552	0.0012	0.7552	0.0012	0.7560	0.0020
11	0.3269	0.7505	0.5692	0.1813	0.7526	0.0021	0.7524	0.0019	0.7519	0.0014
12	0.3585	0.7465	0.5625	0.1840	0.7478	0.0013	0.7473	0.0008	0.7450	0.0015
13	0.3873	0.7385	0.5512	0.1873	0.7395	0.0010	0.7385	0.0000	0.7339	0.0046
14	0.4137	0.7280	0.5334	0.1946	0.7255	0.0025	0.7239	0.0041	0.7167	0.0113
15	0.4373	0.7065	0.5075	0.1990	0.7037	0.0028	0.7016	0.0049	0.6923	0.0142
16	0.4590	0.6755	0.4707	0.2048	0.6710	0.0045	0.6686	0.0069	0.6582	0.0173
17	0.4784	0.6320	0.4226	0.2094	0.6264	0.0056	0.6240	0.0080	0.6140	0.0180
18	0.4960	0.5730	0.3618	0.2112	0.5682	0.0048	0.5663	0.0067	0.5585	0.0145
19	0.5119	0.4990	0.2888	0.2102	0.4971	0.0019	0.4959	0.0031	0.4918	0.0072
20	0.5265	0.4130	0.2031	0.2099	0.4130	0.0000	0.4127	0.0003	0.4130	0.0000
21	0.5398	0.3165	0.1068	0.2097	0.3185	0.0020	0.3191	0.0026	0.3236	0.0071
22	0.5521	0.2120	0.0002	0.2118	0.2147	0.0027	0.2159	0.0039	0.2232	0.0112
23	0.5633	0.1035	-0.1129	0.2164	0.1058	0.0023	0.1072	0.0037	0.1151	0.0116
24	0.5736	-0.0100	-0.2314	0.2214	-0.0067	0.0033	-0.0055	0.0045	0.0001	0.0101
25	0.5833	-0.1230	-0.3561	0.2331	-0.1234	0.0004	-0.1230	0.0000	-0.1230	0.0000
26	0.5900	-0.2100	-0.4499	0.2399	-0.2100	0.0000	-0.2105	0.0005	-0.2170	0.0070

Para cada algoritmo se ha calculado el error absoluto medio (MAE) y el error cuadrático medio (RMSE) y se han recopilado en la Tabla 8-9, de esta manera se puede comparar fácilmente la aptitud de los algoritmos.



Tabla 8-9. MAE y RMSE de los algoritmos para el modelo de diodo doble.

	SA	GA	PSO	PS
MAE	0.1981	0.0017	0.0023	0.0059
RMSE	0.1989	0.0023	0.0033	0.0082

Los resultados son similares a los obtenidos para el modelo de diodo simple, siendo el algoritmo con menores errores absolutos el genético, seguido de cerca por el algoritmo de enjambre de partículas. En tercer lugar, se encuentra el *pattern search* y el de mayor error es el *simulated annealing*, con una gran diferencia respecto a los demás.

Con los parámetros del modelo ya calculados se han trazado las curvas I-V con *fsolve* en cada caso, así como los puntos en los que se calcula el error a la tensión experimental, con las intensidades experimentales marcados con círculos rojos y las intensidades del modelo con cruces negras. Estas curvas son las de la Figura 8.5, la Figura 8.6, la Figura 8.7 y la Figura 8.8 para los algoritmos GA, PSO, PS y SA respectivamente. En la última figura se aprecia que el error medio del *simulated annealing*, de 0.1981 A, es excesivo, quedando la curva muy por debajo de por donde debería pasar.

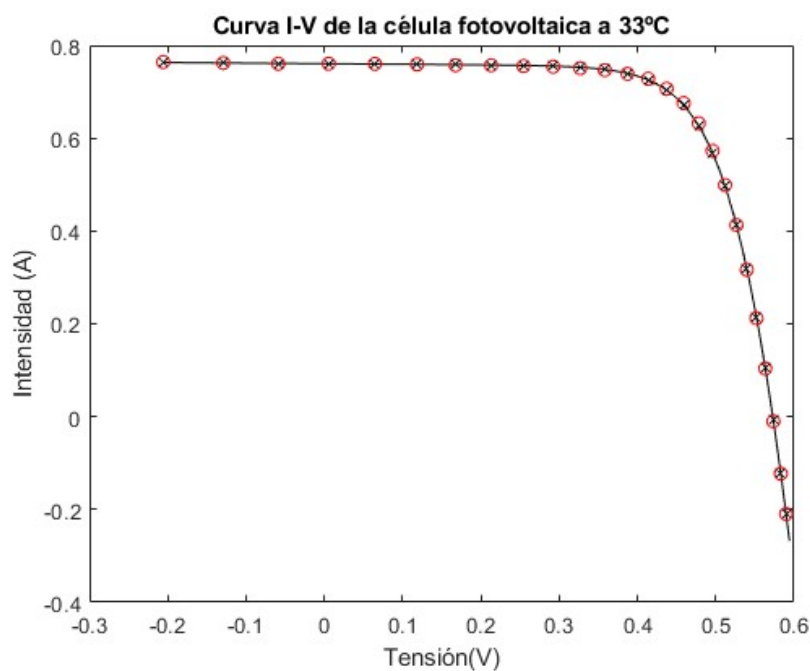


Figura 8.5. Curva I-V GA diodo doble.

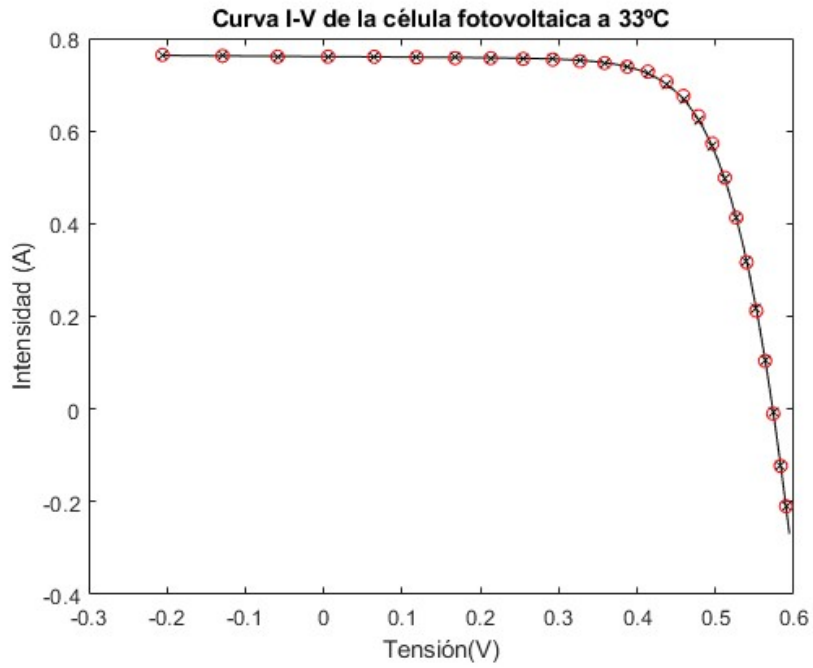


Figura 8.6. Curva I-V PSO diodo doble.

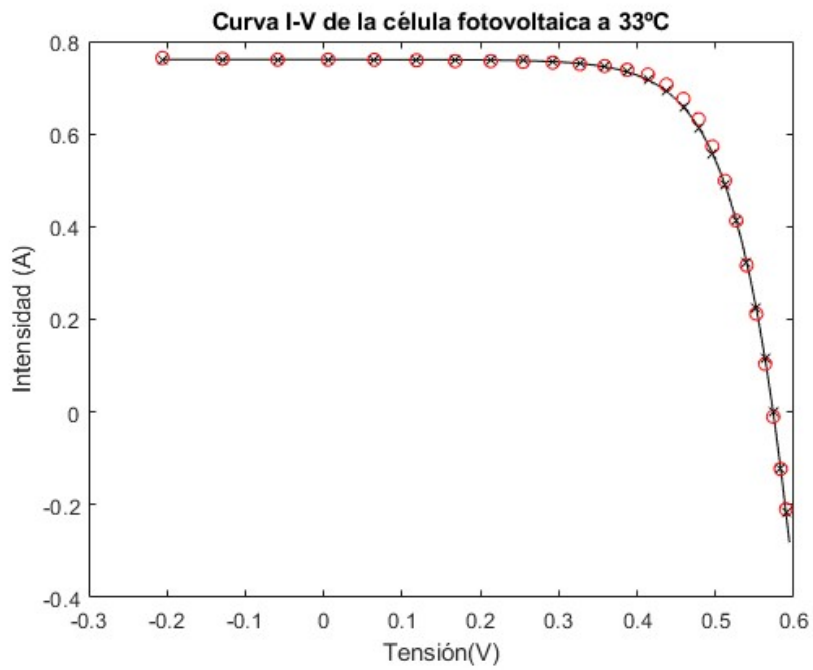


Figura 8.7. Curva I-V PS diodo doble.

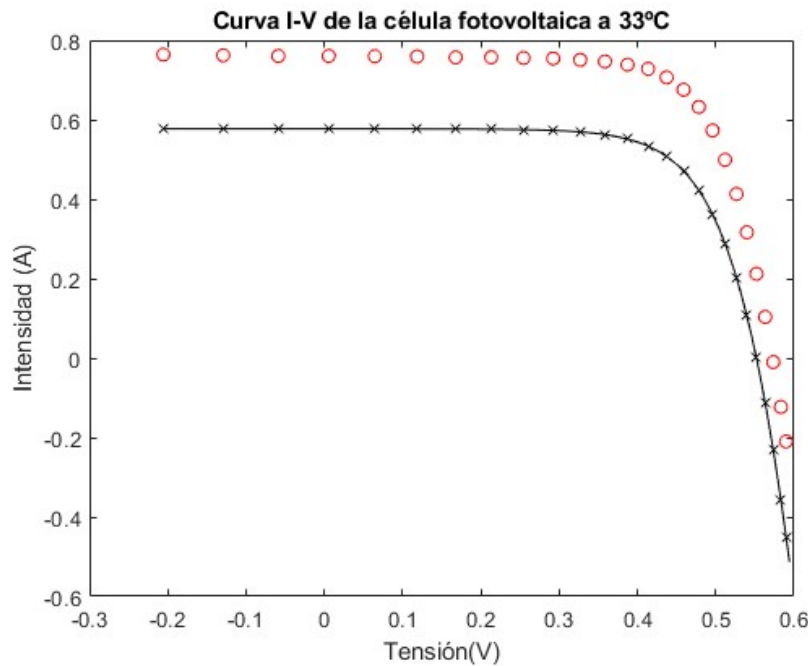


Figura 8.8. Curva I-V SA diodo doble.

#### 8.4. Conclusiones y observaciones del estudio.

Del estudio se extrae la conclusión de que hay un algoritmo claramente menos efectivo para el problema en cuestión, el *simulated annealing*, y los otros tres consiguen resultados aceptables, siendo el más adecuado el genético, seguido del de enjambre de partículas y en tercer lugar el *pattern search*. Estos resultados se han obtenido probando diferentes opciones y sus combinaciones para cada algoritmo, buscando siempre la mejor solución posible. Esto no significa que no sea factible encontrar errores menores con otras opciones como algunos autores desarrollan en sus artículos, sin embargo, en este estudio no se han alcanzado tales errores, ya que en general es algo que depende en cierta medida de la aleatoriedad y las opciones elegidas, las cuales no siempre se incluyen en los artículos.

Además de comparar los diferentes algoritmos, es interesante estudiar si un modelo es mejor que otro, así que se han comparado los errores absolutos medios en la Figura 8.9 y los errores cuadráticos medios en la Figura 8.10, para los modelos de diodo simple y diodo doble y para cada algoritmo. Para evitar que los errores del SA eclipsen a los de los otros algoritmos se han representado en escala logarítmica y se ha invertido la gráfica, por lo que una barra más alta representa un error menor.

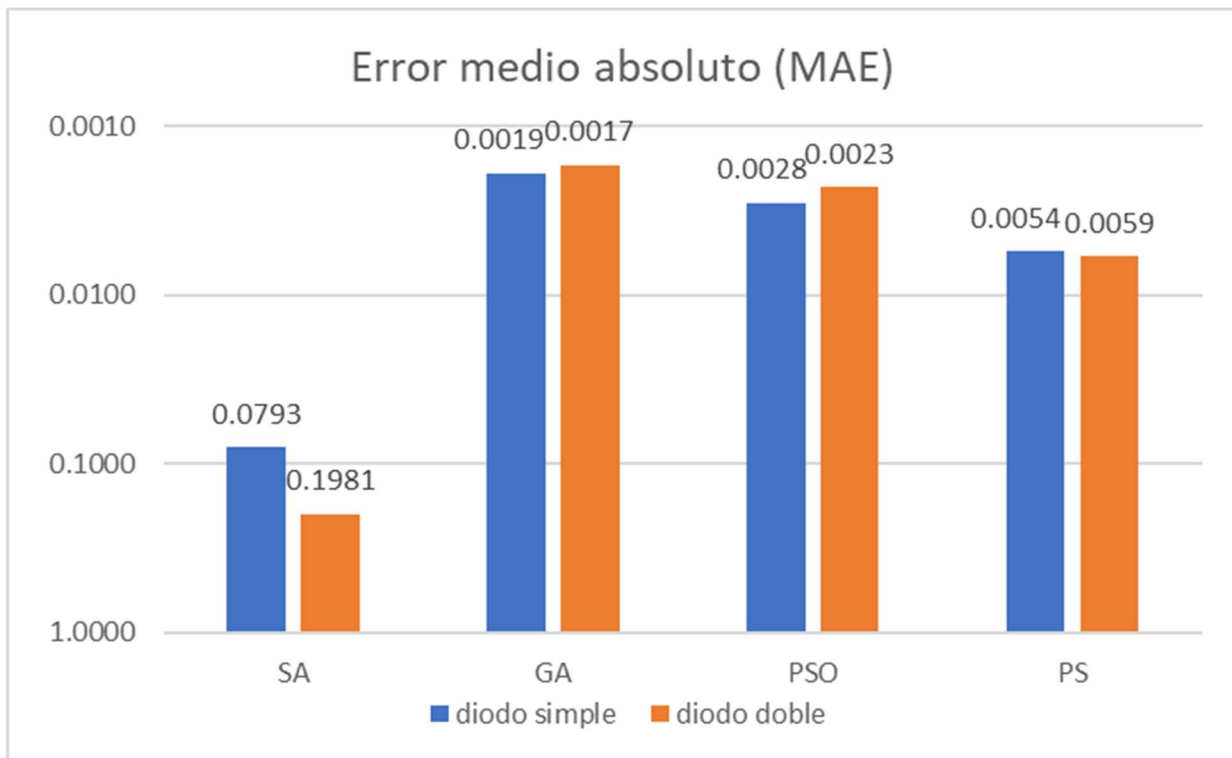


Figura 8.9. Comparativa MAE.

Con la comparativa del error medio absoluto se puede asegurar que los resultados de diodo simple y doble son similares, si obviamos los del SA, siendo ligeramente mejor el modelo de diodo simple en el PS y el modelo de diodo doble para el GA y el PSO.

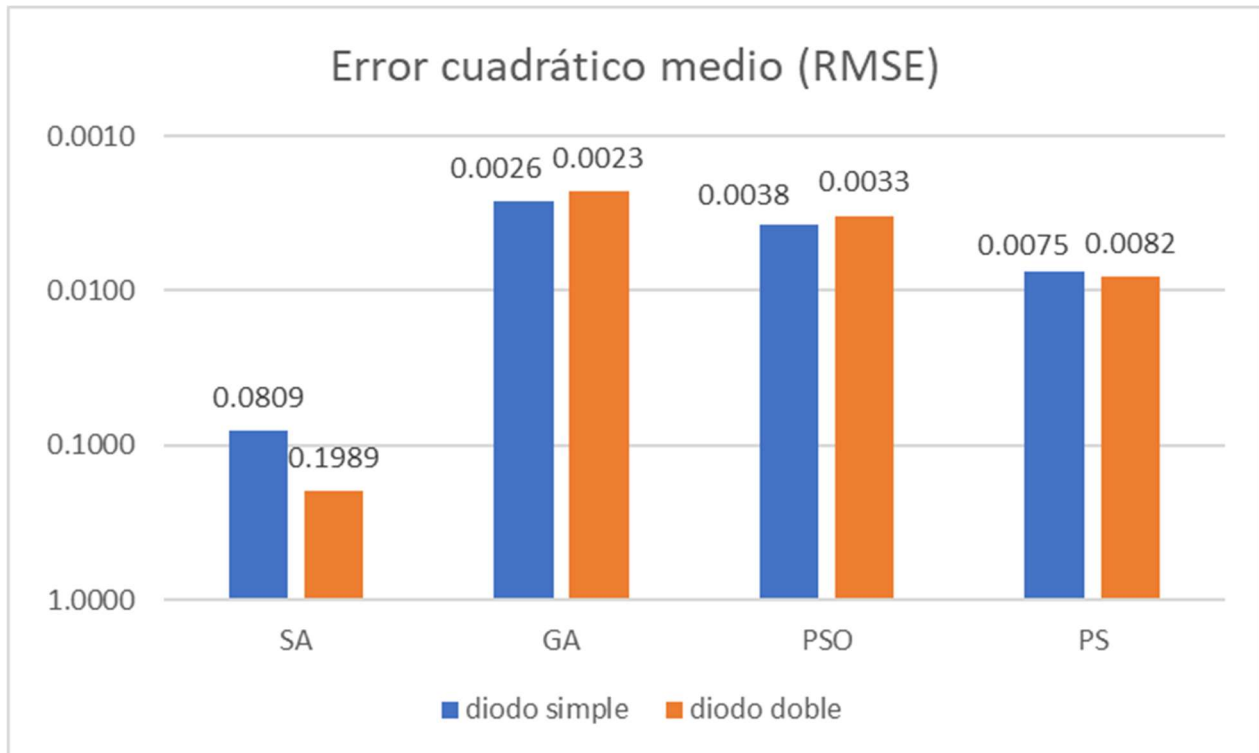


Figura 8.10. Comparativa RMSE.

En cuanto a la comparativa del error cuadrático medio, las conclusiones son similares. El RMSE respecto al MAE penaliza en mayor medida la presencia de errores grandes al elevarlos al cuadrado, por lo que no parece que haya diferencias significativas en la distribución de los errores. Es interesante ver qué forma tienen estos

errores, para lo que se han trazado las gráficas con los errores en cada punto para los algoritmos PSO y PS en la Figura 8.11 y la Figura 8.12 respectivamente, para uno de ellos es ligeramente mejor un modelo y para el otro algoritmo es mejor el otro modelo.

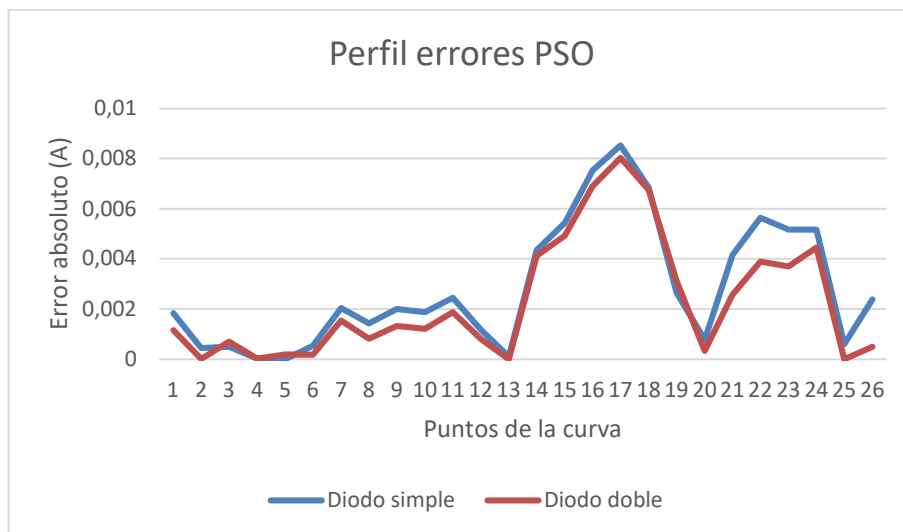


Figura 8.11. Perfil de errores, PSO.

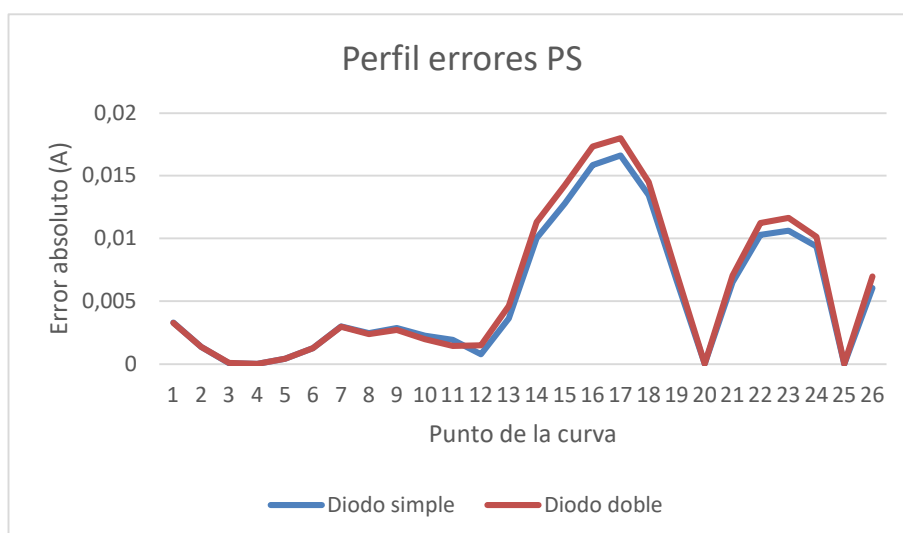


Figura 8.12. Perfil de errores, PS.

Es llamativo el parecido de los dos perfiles, además del perfil del algoritmo genético que es muy similar a estos y no se ha incluido. Esto se debe a que los errores alcanzados son debidos principalmente al ruido de las medidas tomadas en el experimento original, por lo que difícilmente se pueden mejorar ya los resultados, aunque algunos autores defienden que sí.

En la Figura 8.13 se han representado los primeros ocho puntos experimentales en rojo, y los mismos puntos para la solución dada por el algoritmo genético en negro, además de la curva para esta solución también en negro. Se ha escogido esta zona de la curva por ser más horizontal, en la que se aprecia claramente el mayor ruido en los puntos 1 y 7, que se corresponden con los puntos con mayor error de los 8 primeros en las curvas de perfiles de errores. Para las soluciones de los algoritmos PS y GA la pendiente de la curva hace que pase por los puntos 3, 4 y 5; mientras que para la solución del algoritmo PSO una ligera modificación de la pendiente hace que pase por los puntos 4, 5 y 6; con un mayor error en el punto 3. Queda patente que en vista de la Figura 8.13 es imposible que la curva dada por un modelo cualquiera de la célula fotovoltaica, cuya zona de fuente de intensidad es lineal, pase por los puntos 1-5.

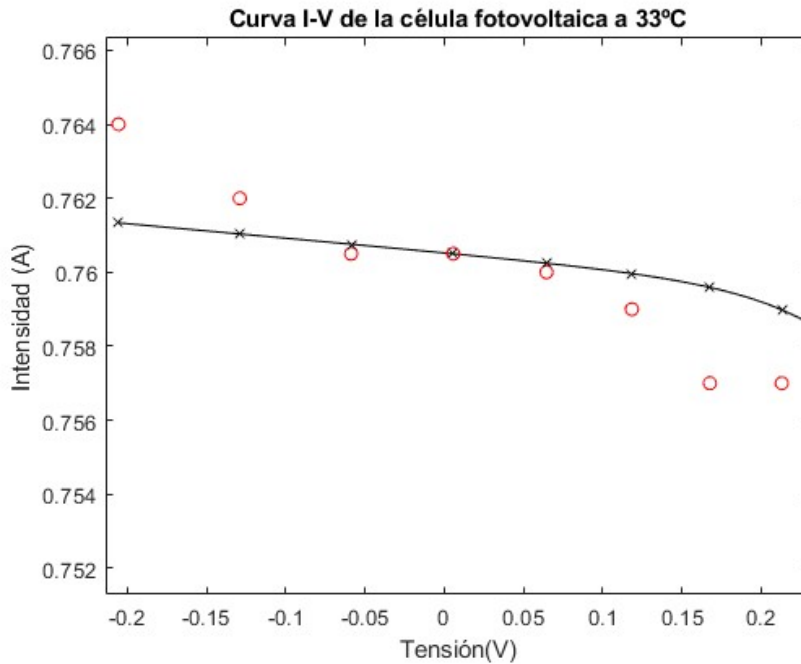


Figura 8.13. Ruido en los datos experimentales, GA diodo simple.

En cuanto a los tiempos de ejecución, la tónica general es similar a la de las pruebas con especificaciones técnicas. En este caso sorprende la disminución del tiempo de computación al aumentar la complejidad del modelo añadiendo un segundo diodo. Esto puede deberse a que el número de iteraciones que necesita para converger es menor, aunque cada iteración lleve más tiempo.

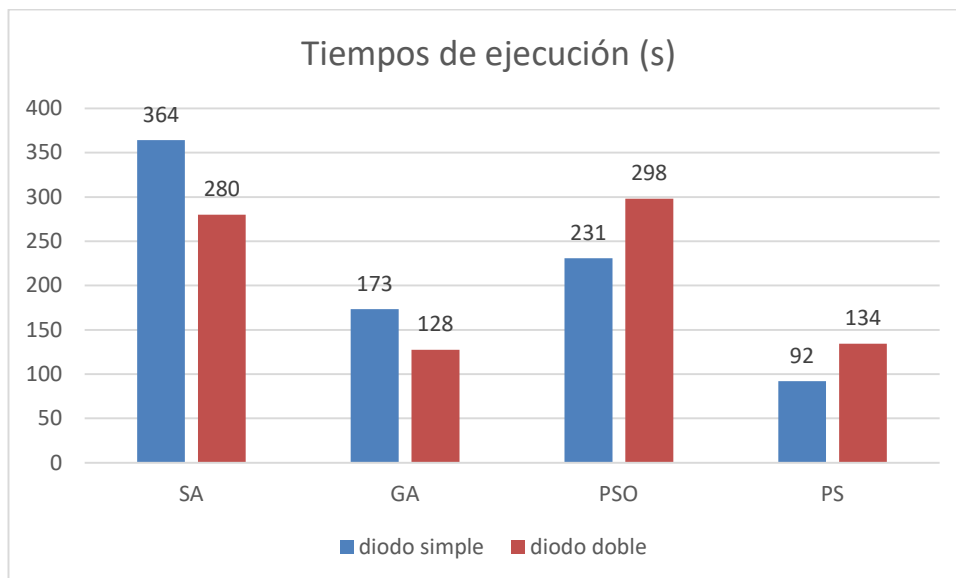


Figura 8.14. Tiempos de ejecución.

Como se ha visto en este capítulo, en el caso de tener numerosos puntos de la curva I-V, el modelo de la célula a considerar es indiferente si lo único importante es el orden de magnitud de los errores, aunque conviene probar ambos si se desean minimizar al máximo.

Por otra parte, los algoritmos a utilizar preferentemente son el GA y el PSO, siendo el PS aceptable también, y se debe evitar la utilización del SA según los resultados obtenidos. Además, si se requieren errores mínimos, la prioridad debe ser disponer de unos datos experimentales con un error lo más reducido posible, ya que el comportamiento de los tres algoritmos aprobados (GA, PSO y PS) es excelente para los datos disponibles.

# 9 CONCLUSIONES DEL ESTUDIO EN AMBOS ESCENARIOS

---

Con los resultados obtenidos de la aplicación de los cuatro algoritmos (GA, PSO, SA y PS) a los dos problemas propuestos (obtención de los parámetros a partir de puntos de catálogo y a partir de datos experimentales) y con dos modelos de célula fotovoltaica (de diodo simple y de diodo doble) se pueden extraer una serie de conclusiones:

1. Los algoritmos PSO y GA son muy superiores a los algoritmos SA y PS para ambos problemas.
2. En el caso del problema de optimización a partir de datos experimentales el GA resultó ligeramente superior al PSO. Sin embargo, hay que destacar que el ruido de los datos es elevado, y no parece justo determinar con total certeza que el GA sea más adecuado en el caso de disponer de datos experimentales.
3. En el caso del problema de optimización a partir de datos de catálogo el PSO resulta muy superior al GA, con muchos errores nulos. A diferencia del otro problema, en este caso sí que hay mayores evidencias para afirmar que es el más adecuado. Esto es debido fundamentalmente a que se dispone de datos de catálogo de muchos modelos, mientras que solo se ha podido estudiar los datos experimentales de un modelo.
4. Para los algoritmos con mejores soluciones (GA y PSO) el modelo de diodo doble consigue unos resultados con menos error por norma general que el modelo de diodo simple. Esta diferencia es pequeña, excepto para el PSO con datos de catálogo, que se obtienen más errores nulos.
5. El uso del algoritmo PS utilizando como solución inicial la obtenida por otro algoritmo mejora la solución, pero la reducción del error es mínima.
6. Simplificar el modelo de diodo simple para pasar de cinco parámetros independientes a tres dificulta obtener soluciones adecuadas, además no se reduce el tiempo de computación ya que requiere el cálculo de los dos parámetros eliminados a partir de los otros tres. Esto se ha demostrado en el caso del algoritmo genético para el problema a partir de datos de catálogo.

Existen numerosos algoritmos que se pueden emplear para este problema además de los cuatro estudiados en este trabajo, algunos de ellos son: *artificial bee colony* (ABC), *differential evolution* (DE), *Joint Approximate Diagonalization of Eigen matrices algorithm* (JADE), *biogeography-based optimisation* (BBO), *innovative global harmony search* (IGHS), *bird mating optimisation* (BMO), *flower pollination algorithm* y *teaching-learning based optimisation* (TLBO). Afirmer con rotundidad que uno es superior al resto es una tarea ardua debido al elevado número de métodos. Otro problema es que al tratarse de métodos metaheurísticos cuya solución obtenida es aleatoria, no se puede demostrar o refutar que las excelentes soluciones que ofrece un autor para apoyar el algoritmo que ha empleado son el resultado accidental de infinidad de ejecuciones, o que incluso se hayan obtenido por otros medios.

Con el objetivo de minimizar el factor aleatoriedad de estos algoritmos, y enfrentarlos en igualdad de condiciones, se ha intentado encontrar siempre la mejor combinación posible de las infinitas opciones con las que se pueden ejecutar los algoritmos en cada caso, para después tomar la mejor solución de diez ejecuciones. Por ello, es hipotéticamente posible obtener errores menores en cualquiera de los casos, debido a la naturaleza

aleatoria de los algoritmos, o a que se ha encontrado una combinación de opciones mejor que la definida en dicho caso.

Para futuras investigaciones sobre el mejor modelo y el mejor algoritmo de obtención de los parámetros de dicho modelo, es fundamental disponer de mayor variedad de datos experimentales, ya que los únicos disponibles al público y los cuales se siguen utilizando actualmente para el testeo de nuevos algoritmos son los del artículo de Easwarakhanthan de 1986.



## REFERENCIAS

- [1] A. R. Jordehi, «Parameter estimation of solar photovoltaic (PV) cells: A review, *Renewable and Sustainable Energy Reviews*,» vol. 61, pp. 354-371, 2016.
- [2] A. Jordehi, «*Renewable and Sustainable Energy Reviews*,» *Elsevier*, n° 61, pp. 354-371, 2016.
- [3] J. B. I. B. & C. B. T. EASWARAKHANTHAN, «Nonlinear Minimization Algorithm for Determining the Solar Cell Parameters with Microcomputers.,» *International Journal of Solar Energy*, vol. 4, n° 1, pp. 1-12, 1986.
- [4] G. J. Villalva MG, «Comprehensive approach to modeling and simulation of photovoltaic arrays.,» *Power Electron IEEE Trans*, vol. 1, n° 24, pp. 198-208, 2009.
- [5] J. Kennedy y R. Eberhart, «Particle Swarm Optimization,» de *Proceedings of IEEE International Conference on Neural Networks IV.*, 1995.
- [6] «Big sur magazine,» [En línea]. Available: <http://www.viatgesbigsur.com/viajes/la-gran-migracion-un-espectaculo-natural-unico/>.
- [7] H. & M. Y. Grandis, «Particle Swarm Optimization (PSO) for Magnetotelluric (MT) 1D Inversion Modeling.,» de *IOP Conference Series: Earth and Environmental Science*. 62. 012033. 10.1088/1755-1315/62/1/012033, 2017.
- [8] Á. Abril. [En línea]. Available: <http://dineroclub.net/2019/02/06/que-son-los-algoritmos-geneticos/>.
- [9] M. & M. M. & M. T. M. I. Ismail (M. S. Ismail), «Characterization of PV panel and Global Optimization of Its Model Parameters using Genetic Algorithm.,» *Energy Conversion and Management.*, n° 73, pp. 10-25, 2013.
- [10] R. a. J. T. A. Hooke, «` Direct Search` Solution of Numerical and Statistical Problems,» *Association for Computing Machinery*, vol. 8, n° 2, p. 212–229, 1961.
- [11] «Dakota reference manual,» [En línea]. Available: [https://dakota.sandia.gov/sites/default/files/docs/6.0/html-ref/method-coliny\\_pattern\\_search.html](https://dakota.sandia.gov/sites/default/files/docs/6.0/html-ref/method-coliny_pattern_search.html).
- [12] «Thai Tohken Thermo Co. LTD,» [En línea]. Available: <https://thaitohken.co.th/en/our-service/normalizing-and-annealing/>.

# GLOSARIO

---

a	Factor de idealidad del diodo
$a_2$	Factor de idealidad del diodo 2
GA	Algoritmo genético
$I_0$	Corriente de saturación del diodo
$I_{02}$	Corriente de saturación del diodo 2
$I_D$	Intensidad en el diodo
$I_{OC}$	Intensidad de circuito abierto
$I_{MPP}$	Intensidad en el punto de máxima potencia
$I_{PV}$	Intensidad de la fuente
$I_{SC}$	Intensidad de cortocircuito
K	Constante de Boltzmann
PS	<i>Pattern Search</i>
PSO	<i>Particle Swarm Optimization</i>
q	Carga del electrón
$R_s$	Resistencia en serie del modelo
$R_p$	Resistencia en paralelo del modelo
SA	<i>Simulated Annealing</i>
T	Temperatura
$V_D$	Tensión en el diodo