



UNIVERSIDAD POLITÉCNICA DE MADRID
ESCUELA TÉCNICA DE INGENIERÍA Y DISEÑO INDUSTRIAL
Grado en Ingeniería Eléctrica

TRABAJO DE FIN DE GRADO

Aplicación web para la estimación del coste de una instalación solar conectada a red

Autor: Ionut Cristian Morariu

Tutor: Óscar Perpiñán Lamigueiro
Departamento de Ingeniería Eléctrica,
Electrónica, Automática y Física aplicada

Madrid, 29 de mayo de 2020

Índice general

1. Introducción	1
1.1. Objetivos	1
1.2. Análisis previo de soluciones	2
1.3. Aspectos técnicos	4
1.3.1. Backend	4
1.3.2. Frontend	4
2. Estado del arte	5
2.1. Situación actual de la generación fotovoltaica	5
2.2. Soluciones existentes y sus carencias	6
3. Parte teórica y desarrollo del código	10
3.1. Obtención de datos del usuario	10
3.1.1. Emplazamiento del usuario	10
3.1.2. Área, inclinación, orientación y nivel de suciedad de la superficie de instalación	12
3.2. Obtención de la irradiación global media en el plano horizontal	12
3.3. Naturaleza de la radiación solar	13
3.3.1. Radiación fuera de la atmósfera terrestre	13
3.4. Cálculo de componentes de radiación solar	16
3.5. Radiación en superficies inclinadas	18
3.5.1. Estimación de irradiancia a partir de irradiación diaria	18
3.5.2. Transformación al plano del generador	20
3.5.3. Pérdidas por ángulo de incidencia y suciedad	23
3.6. Cálculo de la energía producida por el generador	26
3.6.1. Definición de un SFCR	26
3.6.2. Configuración de los elementos del sistema	26
3.6.3. Funcionamiento de una célula solar	28
3.6.4. Asociación de dispositivos fotovoltaicos	30
4. Ejemplo práctico de aplicación	43
5. Conclusión	44
Bibliografía	45

Capítulo 1

Introducción

1.1. Objetivos

El objetivo de este proyecto es el desarrollo de una aplicación o página web de fácil acceso para todos los usuarios, con la finalidad de ofrecer una estimación inicial del coste y la posible generación de una instalación fotovoltaica doméstica de conexión a red.

Durante el resto del documento, si fuera necesario, se hará referencia a la aplicación desarrollada en este proyecto en el nombre de SolarCalc.

Para poder llevar a cabo esta estimación, el usuario introducirá una serie de datos acerca de su emplazamiento y edificación en la que desea situar la instalación, y la aplicación hará todos los cálculos necesarios para ofrecer una aproximación lo más cercana al resultado final teniendo en cuenta todas las variables que puedan intervenir.

La aplicación también ofrecerá otros datos de posible interés para el usuario como: el número de paneles que se pueden instalar, la potencia de dichos paneles y la potencia del inversor.

La idea de esta aplicación surge de una conversación que tuve con un conocido cuando estaba planificando la construcción de su nueva vivienda, en la cual quería realizar una instalación fotovoltaica para reducir el gasto en la factura de electricidad.

En su búsqueda no encontró ningún servicio que fuera lo suficientemente sencillo de entender para una persona sin ningún tipo de conocimiento previo acerca de la generación fotovoltaica, que le aportase la posibilidad de poder personalizar los cálculos a su emplazamiento y planos de construcción.

Otro de los puntos clave de la aplicación es que sea de código abierto y gratuita para los usuarios, usando fuentes de información disponibles para cualquier interesado. Todos los pasos y operaciones se podrán obtener, analizar y reutilizar de manera gratuita desde un repositorio de Github ¹.

¹*Github.com*: Plataforma online de almacenamiento de código y documentación de fuentes abiertas.

Los objetivos detallados de esta aplicación son los siguientes:

- Diseñar una interfaz de usuario amigable y sencilla de usar para que la pueda utilizar un gran número de personas sin necesidad de conocimientos sobre energía fotovoltaica.
- Obtención de los datos de irradiación en el emplazamiento indicado por el usuario mediante el uso de API² externas.
- Realizar todos los cálculos necesarios para ofrecer una estimación competente de los siguientes datos:
 - Número de paneles que se pueden instalar.
 - Potencia máxima a instalar.
 - Potencia del inversor.
 - Energía eléctrica producida en un año.

1.2. Análisis previo de soluciones

Antes de comenzar el desarrollo del proyecto, se llevó a cabo una revisión de las soluciones existentes de estimaciones de instalaciones fotovoltaicas existentes en el mercado para decidir si tenía cabida una aplicación como la que se iba a desarrollar.

Algunas de las soluciones encontradas fueron:

1. PVSyst - Photovoltaic Software

El software PVSyst, desarrollado por la empresa suiza con el mismo nombre es quizá el más conocido dentro del ámbito del estudio y la estimación de instalaciones fotovoltaicas. Ofrece una amplia capacidad de personalización de todos los componentes de la instalación.

2. CalculationSolar.com

Es el primer resultado de Google al buscar el término “*calculadora de instalaciones fotovoltaicas*”, por tanto será una de las primeras aplicaciones que una persona que desea realizar una instalación en su vivienda visite.

3. SISIFO

Es una herramienta web diseñada y desarrollada por el Grupo de Sistemas Fotovoltaicos del Instituto de Energía Solar de la Universidad Politécnica de Madrid. Ha sido y es la herramienta interna utilizada por los ingenieros de dicho grupo.

4. PVGIS

Aplicación web desarrollada por el **European Commission Joint Research Center** desde 2001. Su enfoque es asistir en el cálculo y estimación de instalaciones fotovoltaicas, ya sean conectadas a red, de seguimiento o de autoconsumo.

²*Application Programming Interface*: conjunto de funciones y procedimientos que ofrece la posibilidad de un software a interactuar con otro.

5. System Advisor Model

System Advisor Model (SAM), desarrollado por el Laboratorio Nacional de Energías Renovables, perteneciente al Departamento de Energía del gobierno americano, es una software técnico-económico gratuito que ayuda a la toma de decisiones en el amplio campo de las energías renovables. Ofrece un conjunto de soluciones muy completas no solamente relacionadas con la energía fotovoltaica, sino también termosolar, eólica, geotermal o biomasa, entre otras.

6. solaR

solaR es un paquete de código para el entorno de R, desarrollado por Oscar Perpiñán, que a pesar de tener objetivos diferentes a los planteados por la aplicación desarrollada en este proyecto, ha servido como base para validar todos los cálculos que se han realizado en ella.

En el apartado 2.2 se lleva a cabo un desarrollo mas detallado de las características de las soluciones mencionadas así como sus diferencias con la propuesta de este proyecto.

1.3. Aspectos técnicos

Para construir cualquier página web se deben desarrollar dos sistemas diferentes llamados Backend y Frontend. El Backend es la parte de cálculo y tratamiento de peticiones, comúnmente llamado server o servidor. Por otro lado se encuentra el Frontend que es la parte de cara al usuario, la que se encarga de recoger los datos introducidos por este y enviarlos al servidor.

1.3.1. Backend

Para el Backend se ha empleado una tecnología basada en Javascript llamada NodeJS³, con la ayuda de las librerías ExpressJS⁴ y Mongoose⁵.

La base de datos que se ha utilizado para almacenar los datos necesarios ha sido MongoDB.

En el servidor se realizan varias tareas relacionadas con los cálculos necesarios. Algunas de estas tareas son:

- Obtención de los datos de irradiación global media en el plano horizontal para el emplazamiento indicado
- Proceso completo de cálculo para pasar de la irradiación en el plano horizontal al plano inclinado y orientado según los datos introducidos por el usuario
- Proceso de obtención de los datos relacionados con el perfil horario de temperatura en el emplazamiento indicado
- Gestión de las diferentes rutas que constituyen la API.

1.3.2. Frontend

Para el Frontend de la página se han utilizado las tres tecnologías necesarias para poder desarrollar una página web: HTML5, CSS3, Javascript.

Esta parte de la página es la encargada de recoger los datos del usuario y enviarlos al servidor para que se realicen los cálculos. Una vez realizados dichos cálculos, la página mostrará la información relevante al usuario, junto con algunos gráficos adicionales.

Tanto el backend como el frontend están almacenados en un servidor de Linux remoto activo 24/7.

Todas las tareas mencionadas tanto en la parte de Backend como en la parte de Frontend se describirán en detalle en la sección 3, junto con todos los cálculos en los que se ha basado.

³NodeJS: Entorno de ejecución basado en el motor de Chrome llamado V8. <https://nodejs.org/en/>

⁴ExpressJS: Framework web para el entorno de NodeJS. <https://expressjs.com/es/>

⁵Mongoose: Capa intermedia de interacción para las BBDD MongoDB <https://mongoosejs.com/>

Capítulo 2

Estado del arte

2.1. Situación actual de la generación fotovoltaica

Según el informe anual de la UNEF¹ en 2019[1] la evolución de las energías renovables superó incluso las expectativas mas optimistas alcanzando valores proximos a los 100 GW.

Los aspectos que más han influido en estas cifras han sido, entre otros, la reducción drástica del coste de producción de dichas tecnologías. De hecho, la energía fotovoltaica es ya más barata que la generada por plantas de combustibles fósiles en términos de LCOE². Según menciona Bloomber Energy Finance, la fotovoltaica seguirá reduciendo sus costes un 34 % hasta 2030.

Sumado a la reducción del coste, el aumento de compraventa de energía a largo plazo - que sigue en alza desde 2018, alcanzando los 14 GW - es otro de los factores influyentes en el crecimiento del sector. Así lo es también la reducción del precio de la energía en las subastas, alcanzando valores tan bajos como 20\$/MWh.

Concretamente en Europa, el crecimiento anual de la capacidad solar instalada ha sido de un 23 %, con Alemania como líder sumando otros 2,95 GW respecto a la capacidad del año anterior. En segundo y tercer lugar se encuentran Turquía y los Países Bajos.

Bajando un nivel más, nos encontramos con el mercado español, que, según estimaciones del mismo informe de la UNEF, la potencia total instalada experimentó un crecimiento significativo, llegando hasta el valor de 262 MW, sumando la potencia instalada tanto de generación centralizada como la de autoconsumo.

Para este año 2020 se estima se estima que la instalación de energia fotovoltaica alcance el umbral de los 20 GW. Si se cumplen las expectativas, la capacidad podría llegar a alcanzar los 200 GW para 2023.

¹UNEF: Unión Española fotovoltaica

²LCOE: Levelized Cost of energy: Medición del coste medio de generación de energía de una planta a lo largo de su vida útil

Un papel importante lo juegan las autoridades tanto nacionales como a nivel europeo, que apuestan por las fuentes de energías limpias. Para ser exactos, el año 2018 fue uno de los más relevantes en materia de política energética europea desde que se aprobó el tercer paquete de energía en 2009. De las ocho propuestas que se aprobaron, destaca por su importancia para el sector fotovoltaico la directiva 2018/2001, en la que se recoge el derecho básico al autoconsumo, individual o colectivo, al almacenamiento y sobretodo a la venta de excedentes.

En el panorama español, tras varios años de parálisis debida a la compleja situación política de los últimos años, la energía fotovoltaica volvió a recuperar algo de impulso a final del año 2019. Durante el año 2018 y especialmente el 2019, se han incrementado sustancialmente las instalaciones de fotovoltaica, en gran parte gracias a las expectativas generadas por la eliminación del denominado impuesto al sol y la progresiva suspensión de trabas administrativas todas ellas propiciadas por el nuevo marco legal establecido por la administración pública mediante los Reales Decretos 15/2019 y 244/2019.

Según datos proporcionados por la Red Eléctrica Española ³, la potencia instalada en España ha experimentado un crecimiento de 1,2GW en los últimos 4 años, pasando de 4.6 GW en 2016 a 5.8 GW en 2019.

Se trata del mayor ritmo de crecimiento desde 2008, cuando se instalaron cerca de 2.7 GW de nueva potencia. Es una buena noticia, pero no exenta de dificultades debidas sobre todo a la unidireccionalidad de la red eléctrica de nuestro territorio. Sin ir mas lejos, el operador técnico del sistema, REE ha denegado la instalación de 26,3 GW de nueva potencia debido a la imposibilidad de los nudos para gestionar la energía producida por dicha capacidad.

Esto no significa que dicha energía no se vaya a instalar, sino que habrá que esperar a la nueva planificación energética prevista para el periodo 2021-2026 en la que ya están trabajando tanto la REE como las comunidades autónomas para brindar mas oportunidades para los sistemas de conexión a red.

En definitiva, como podemos observar, las energías renovables, y en especial la fotovoltaica esta dando mucho que hablar y cada vez es una tema más tratado por el público general y por tanto, es un aspecto a tener en cuenta a la hora de construir nuevas edificaciones o mejorar la eficiencia energética de las existentes.

Surge por tanto la necesidad de aplicaciones y soluciones para la estimación de instalaciones fotovoltaicas que sean intuitivas y fáciles de usar para aprovechar el gran interés que está mostrando el público general.

2.2. Soluciones existentes y sus carencias

Como ya se ha mencionado en el apartado 1.2 existen multitud de soluciones para la estimación o simulación de instalaciones fotovoltaicas. Sin embargo, es posible que la aplicación descrita en este proyecto siga teniendo cabida porque ofrece ciertas funciones que, en cierta medida, no existen en las soluciones que se han estudiado.

A continuación se describirán con detalle las soluciones mencionadas anteriormente, junto con sus capacidades y carencias.

³<https://www.ree.es/es/datos/publicaciones/series-estadisticas-nacionales>

1. PVSyst - Photovoltaic Software [2]

El software PVSyst, desarrollado por la empresa suiza con el mismo nombre es quizá el más conocido dentro del ámbito del estudio y la estimación de instalaciones fotovoltaicas. Ofrece una amplia capacidad de personalización de todos los componentes de la instalación.

PVSyst es capaz de calcular con amplio detalle el diseño y dimensionado del sistema, zonas de sombra, envejecimiento del material, almacenamiento de energía, entre otras opciones.

PVSyst se diferencia de la aplicación que se desarrolla en este proyecto en algunos puntos importantes como:

- Es de pago, con una licencia anual de aproximadamente €952 mientras que mi solución es gratuita.
- Es un programa que se debe instalar en un ordenador de Windows (No funciona en Linux u OSX).
- Es poco intuitivo para un usuario con bajos conocimientos de instalaciones fotovoltaicas.

En resumen, PVSyst es un software mucho mas completo y complejo que la solución que yo propongo, y que va enfocada a un público con una base sólida sobre energía fotovoltaica.

2. CalculationSolar.com [3]

Como ya se ha mencionado en la introducción, CalculationSolar.com es el primer resultado que aparece en el motor de búsqueda de Google cuando se introduce el término “*calculadora de instalaciones fotovoltaicas*” y por tanto será una de las primeras opciones que un usuario que está interesado en realizar una instalación fotovoltaica considere.

A diferencia de PVSyst, CalculationSolar.com es una solución en la web y gratuita, por lo que la barrera de acceso es mas baja. Nada mas entrar a la pagina lo primero con lo que nos encontramos es con un formulario sencillo sobre el emplazamiento y la configuración de la instalación. La información que se solicita es sencilla e intuitiva, incluso se ofrece la posibilidad de clicar en un mapa interactivo para determinar las coordenadas.

Lo siguiente es introducir la información acerca de las necesidades de potencia, dado que esta calculadora esta enfocada a las instalaciones de autoconsumo sin conexión a red.

Una vez introducidos estos datos, el programa realiza los cálculos y nos ofrece un resultado bastante detallado del campo fotovoltaico, el regulador de carga, la batería y el inversor que mejor encajaría con nuestro requisitos.

La principal diferencia con la aplicación que yo he desarrollado es requisitos versus limitaciones. Es decir, CalculationSolar.com te permite seleccionar tus requisitos de potencia y te indica el generador que vas a necesita para poder hacer frente a dicha carga. En cambio, mi solución indica la potencia y energía que se puede llegar a generar con las limitaciones arquitectónicas impuestas por el usuario.

3. SISIFO [4]

La solución propuesta por el Grupo de Energías Fotovoltaicas del Instituto de Energía Solar de la Universidad Politécnica de Madrid es también un solución web para el calculo de instalaciones solares tanto de bombeo de agua como conectadas a red.

El sistema de input de información del usuario es muy completo. Paso por paso le permite a éste introducir desde los datos geográficos y meteorológicos hasta hasta los valores de

perdidas en los diferentes cableados del sistema. No obstante, los valores por defecto son muy válidos así que incluso un usuario con poco conocimiento sobre energía podría llegar a obtener unos resultados fiables.

Una vez realizada la simulación, la aplicación aporta multitud de resultados detallados tales como Irradiaciones en el plano horizontal e inclinado, temperaturas y energía producida.

Esta aplicación es la más parecida, salvando las distancias, a la que se desarrolla en este proyecto. Sin embargo, es posible que puede llegar a abrumar en cierta medida a un usuario que desea solamente conocer cual es la potencia o energía que puede llegar a producir en su casa, para saber si le va a resultar rentable la inversión.

4. **PVGIS [5]**

PVGIS es la solución desarrollada por el JRC (Centro de Investigación Conjunta), que forma parte del EU Science Hub. Consiste en una aplicación web que nos permite estimar la energía que puede llegar a producir una instalación fotovoltaica en función de los parámetros introducidos por el usuario.

Tras elegir la localización como primer paso, el programa nos abre la posibilidad de elegir el tipo de instalación, entre conectada a red, con seguimiento, o autónomo.

En este caso, para poder compararlo con la aplicación de este proyecto, se va a realizar la estimación de un sistema conectado a red. La principal diferencia se presenta cuando el programa solicita al usuario la potencia FV pico instalada, así como la tecnología y las pérdidas porcentuales del sistema.

Una vez introducidos estos datos, la aplicación nos ofrece los resultados de producción de energía mensual del sistema.

A diferencia de esto, SolarCalc, solicita al usuario el dato de la superficie disponible, para estimar la potencia máxima y por consiguiente, la energía máxima.

5. **System Advisor Model [6]**

Esta aplicación desarrollada por el Laboratorio de Energías Renovables del Departamento de Energía de los Estados Unidos. Es un conjunto de soluciones enfocadas a facilitar la toma de decisiones relacionadas con el campo de las Energías Renovables. Entre sus funcionalidades se incluyen programas de cálculo de Fotovoltaica, Termosolar, Eólica, Geotermal o Biomasa.

Es una herramienta muy completa a la vez que compleja, con un enfoque centrado en el apartado económico y la rentabilidad. En cuanto al cálculo y la estimación de una instalación fotovoltaica, ofrece una amplia posibilidad de personalización de cada uno de los campos que afecta a dicho cálculo. Cada uno de los parámetros de la radiación, el módulo, el inversor, las sombras e incluso de la inversión y amortización de la instalación son totalmente ajustables.

Con ésta aplicación sucede como con alguna de las mencionadas anteriormente, que puede llevar a confusión a un usuario medio sin conocimientos relacionados con la fotovoltaica. Estos usuarios son el público objetivo de la aplicación de SolarCalc, desarrollada en este proyecto.

6. **solaR [7]**

Este paquete para R, desarrollado por Oscar Perpiñán, permite llevar a cabo estudios tanto del rendimiento de los sistemas fotovoltaicos como de la radiación solar. Incluye una serie de clases, métodos y funciones para calcular aspectos como la geometría solar, radiación solar

incidente sobre un generador, realizar el paso de un generador horizontal a un generador inclinado y orientado y simular el rendimiento de diferentes aplicaciones de la energía fotovoltaica.

A pesar de no ser una aplicación de estimación de instalaciones solares como tal, has sido, junto con el libro [8], la base de contraste y referencia de los cálculos que se han sido necesario llevar a cabo para poder desarrollar la aplicación de SolarCalc.

Capítulo 3

Parte teórica y desarrollo del código

El proceso de cálculo que se va a seguir para la estimación completa de la instalación fotovoltaica conectada a red es el que se detalla en el libro de Óscar Perpiñán, tutor de este trabajo, denominado Energía Solar Fotovoltaica [8]. También se harán menciones a las presentaciones que se encuentran en el mismo enlace que el libro.

A lo largo de este capítulo, se utilizará el término **aplicación** para referirse a todo el conjunto de código relacionado tanto con proceso de obtención de todos los datos necesario como el de mostrar la información relevante al usuario.

3.1. Obtención de datos del usuario

Como se ha mencionado anteriormente, el objetivo principal de la aplicación es de realizar todos los cálculos que se necesitan para estimar una instalación fotovoltaica conectada a red, de la manera mas exacta posible, en un emplazamiento concreto elegido por el usuario. Por tanto, el primer paso que tuve que dar con la aplicación fue la de obtener todos los datos necesarios para realizar dichos cálculos. Estos datos son:

- Emplazamiento del usuario
- Área, inclinación, orientación y nivel de suciedad de la superficie de instalación

3.1.1. Emplazamiento del usuario

Mas adelante, para poder obtener los datos de irradiación global en el plano horizontal, se necesitan los datos de latitud y longitud del sitio del que se quieren obtener dichos datos. Sin embargo, es poco intuitivo pedirle a un usuario que introduzca sus coordenadas, dado que la mayoría desconocen dichos datos.

Por lo tanto, la ruta que he tomado es la de pedirle al usuario su dirección, o una dirección cercana a su localización, y utilizar la API de Google Maps ¹ para convertir dicha dirección en las

¹API Google Maps: Enlace interactivo al que se le pueden enviar los datos de una dirección y devuelve las coordenadas de latitud y longitud de un emplazamiento.

coordenadas de latitud y longitud que necesito para poder extraer la irradiación que he mencionado antes.

Este proceso comienza por recoger los datos de la dirección, municipio y código postal a través del formulario que aparece en la página web.

Estos datos son recogidos en el código a través de un nombre único que han recibido:

```
1 const addressInput = document.querySelector('#address');
2 const cityInput = document.querySelector('#city');
3 const postalInput = document.querySelector('#postal');
```

Listing 3.1: Variables correspondientes a los tres campos

Una vez que tenemos estos datos recogidos en las variables, podemos pedir a la API de Google Maps las coordenadas de latitud y longitud de dicho emplazamiento encadenando las tres variables y una clave única de identificación, para obtener un enlace único que se corresponde a dicha localización.

```
1 const getCoordinates = async () => {
2   const address = addressInput.value.split(' ').join('+');
3   const city = cityInput.value;
4   const postal = postalInput.value;
5   const requestURL = `${googleEndpoint}address=${address},${city},${postal}
6     ,spain&key=${googleApiKey}`;
7   const response = await fetch(requestURL);
8   const data = await response.json();
9   const info = {
10     formattedAddress: data.results[0].formatted_address,
11     lat: data.results[0].geometry.location.lat,
12     long: data.results[0].geometry.location.lng
13   };
14
15   return info;
16 };
```

Listing 3.2: Función encargada de solicitar los datos a la API

La función de **getCoordinates** (3.2) recoge el valor de la dirección y reemplaza los espacios con el signo + (formato requerido por la API) y lo concatena con el valor del campo de la ciudad y el código postal. Al final le añade una clave única que identifica la aplicación a la hora de establecer límites de uso y evitar abuso de la API.

Una vez creado este enlace único, el código lanza la petición al servicio y retorna con la información que es recogida y se guarda en dos variables **lat** y **long** para ser utilizadas posteriormente, a la hora de obtener los datos de irradiación global.

3.1.2. Área, inclinación, orientación y nivel de suciedad de la superficie de instalación

Además de la información de latitud y longitud del emplazamiento, el cálculo de la instalación también requiere de información relacionada con el área, la inclinación, la orientación y el nivel de suciedad de la superficie donde se va a realizar la instalación, para poder realizar una estimación lo mas exacta posible.

Estos valores son recogidos directamente de los campos de la pagina web, al igual que los campos anteriores, sin necesitar ningún trato especial:

```
1 const slope = document.querySelector('#slope');
2 const area = document.querySelector('#area');
3 const orientation = document.querySelector('#orientation');
4 const dirtLevel = document.querySelector('#dirt-level');
```

Listing 3.3: Variables correspondientes a los campos indicados

3.2. Obtención de la irradiación global media en el plano horizontal

El primer paso del cálculo de una instalación fotovoltaica es el de conocer la irradiación global media en el plano horizontal para el emplazamiento donde se va a realizar el cálculo. En la sección anterior se obtuvieron las coordenadas de latitud y longitud del emplazamiento introducido por el usuario.

En esta sección, se va a obtener la irradiación para las dichas coordenadas utilizando un servicio de ADRASE ², un proyecto realizado por el CIEMAT. Este servicio ofrece de manera gratuita y de uso libre, datos correspondientes a valores medios mensuales de irradiación global en el plano horizontal para toda la geografía española.

Los datos están disponibles a través de un mapa interactivo y de unos enlaces personalizados que incluyen la valores de latitud y longitud para los que se desea obtener dicha información. En esos enlaces se ofrecen los datos de irradiación global mínima, media y máxima en el plano horizontal.

Con el objetivo de no saturar la pagina de ADRASE y evitar error de funcionamiento de la aplicación debidos a la posibilidad de que la página desaparezca en un futuro, se ha realizado una descarga de los datos de la página, con un intervalo de 0.1 tanto en longitud como en latitud y se han guardado en una base de datos. De esta forma también se reducen los tiempos de cálculo en gran medida al tener un acceso casi instantáneo a los valores medios de irradiación. Una explicación detallada de como se ha realizado dicha descarga se incluye en el Anexo. (** TO DO Anexo con explicación **).

²ADRASE: Acceso a Datos de irradiación Solar en España <http://www.adrase.com/>

3.3. Naturaleza de la radiación solar

3.3.1. Radiación fuera de la atmósfera terrestre

La radiación emitida por el Sol atraviesa el espacio vacío en todas las direcciones sin sufrir pérdidas a causa de la interacción con los medios materiales. Sin embargo, la irradiancia solar, que se define como la densidad de flujo radiante solar, se ve atenuada por el cuadrado de la distancia. Parte de esta irradiancia es interceptada por la Tierra. Dada la relación entre la distancia con el Sol y el tamaño de nuestro planeta, se puede asumir que su valor es constante en toda la superficie exterior de la atmósfera. El valor promedio de esta irradiancia, según varias campañas de medición es de $B_0 = 1367 \frac{W}{m^2}$.

Para calcular la irradiancia incidente en una superficie tangente a la atmósfera en una latitud, debemos tener en cuenta que la distancia entre la Tierra y el Sol varía a lo largo del año, debido a la excentricidad de la órbita elíptica que describe nuestro planeta. La ecuación para calcular dicho valor es:

$$B_0(0) = B_0 \epsilon_0 \cos \theta_{zs} \quad (3.1)$$

Es importante resaltar que el valor de la irradiancia extra-atmosférica o extra-terrestre solo requiere consideraciones atmosféricas. Así, integrando la ecuación 3.1, podemos calcular la irradiación diaria extra-terrestre con la ecuación obtenida:

$$B_{0d}(0) = -\frac{T}{\pi} B_0 \epsilon_0 (\omega_s \sin \phi \sin \delta + \cos \phi \cos \delta \sin \omega_s) \quad (3.2)$$

Es posible demostrar que el promedio mensual de esta irradiación diaria coincide numéricamente con el valor de irradiación diaria correspondiente a los denominados días promedios, días en los que la declinación correspondiente coincide con el promedio mensual. Por tanto, podemos calcular el valor medio mensual de la irradiación diaria extra-atmosférica con el valor de la declinación de uno de los doce días promedio.

Estos doce días promedios son:

Mes	Ene	Feb	Mar	Abr	May	Jun	Jul	Ago	Sep	Oct	Nov	Dic
d_n	17	45	74	105	135	161	199	230	261	292	322	347

Cuadro 3.1: Días promedio

Para calcular esta irradiación diaria, debemos calcular primero los diferentes componentes que forman parte de esta:

- $B_0 = 1367 \frac{W}{m^2}$
- Factor de corrección por excentricidad: $\epsilon_0 = 1 + 0,033 \cdot \cos(\frac{2\pi}{365})$

En código se representa de la siguiente manera:

```
1  const exct = 1 + 0.033 * Math.cos((2 * Math.PI * elem.normalDay) / 365)
```

Listing 3.4: Factor de corrección por excentricidad

- Ecuación de Cooper para la declinación: $\delta = 23,45 \deg \sin\left(\frac{2\pi(d_n+284)}{365}\right)$

En código se representa de la siguiente manera:

```
1  const decl = 23.45 * Math.sin((2 * Math.PI * (elem.normalDay + 284)) / 365)
```

Listing 3.5: Ecuación de Cooper para declinación

- Cenit Solar: $\cos(\theta_{zs}) = \cos(\delta) \cos(\omega) \cos(\phi) + \sin(\delta) \sin(\phi)$

Para el calculo del cenit solar, hay que tener en cuenta que varía en función de la hora del día, así que se deberá calcular un valor para cada una de las 24h del día, con la diferencia de que en el cálculo, las horas irán de -12 a 12 en lugar de 0 a 23 y además, tendremos que convertir las horas a ángulos multiplicando por 15deg. De tal manera, que en código, el calculo del cenit solar, tendrá ésta representación:

```
1  for (let h = -12; h < 12; h++) {
2      const cosZenit =
3          Math.cos(deg2rad(elem.decl)) * Math.cos(deg2rad(h * 15)) *
4          Math.cos(deg2rad(newData.latitude)) +
5          Math.sin(deg2rad(elem.decl)) * Math.sin(deg2rad(newData.latitude))
6      const zenitVal = rad2deg(Math.acos(cosZenit))
7  }
```

Listing 3.6: Cálculo del cenit solar

Aplicando estos cálculos a la ecuación 3.2 podemos calcular el valor de la irradiancia extra-terrestre diaria $B_{0d}(0)$ con la siguiente expresión:

```
1  const B0d0 =
2      -(24 / Math.PI) *
3      B0 *
4      elem.exct *
5      (elem.ws * Math.sin(deg2rad(newData.latitude)) * Math.sin(deg2rad(elem.decl)) +
6      Math.cos(deg2rad(elem.decl)) * Math.cos(deg2rad(newData.latitude)) *
7      Math.sin(elem.ws))
```

Listing 3.7: Ecuación para B0d0

donde:

- B_0 : valor contante de 1367 W/m^2
- exct : excentricidad
- ws : ángulo amanecer
- decl : declinación

Para el cálculo de la irradiancia solar que finalmente incide en una superficie localizada en la corteza terrestre será útil distinguir tres componentes diferenciados, comúnmente denominados:

- Radiación Directa, B : representa la fracción procedente en línea directa del Sol.
- Radiación Difusa, D : representa toda la radiación procedente de todo el cielo, excepto del sol, y por tanto incluye todos los rayos dispersados por la atmósfera. Es una radiación que depende del estado de la atmósfera, y variará en función de las condiciones climatológicas.
- Radiación del albedo, R : es aquella fracción de la radiación procedente de la reflexión con el suelo. Habitualmente, supone una contribución muy pequeña, que en algunos casos, como el de este proyecto, puede ser despreciada.

La suma de las tres componentes constituyen la denominada irradiancia global:

$$G = B + D + R \quad (3.3)$$

Nomeclatura

Es importante distinguir, dentro de las ecuaciones que modelan el comportamiento de la radiación solar, la forma de indicar cada componente de la irradiancia, el instante o el período en el que se recibe.

Es recomendable leer estas expresiones en el orden período, forma, tiempo y lugar utilizando el formato de nomenclatura de la siguiente ecuación

$$Forma_{\text{tiempo, promedio}}(\text{lugar}) \quad (3.4)$$

Para expresar el lugar de incidencia caben las siguientes posibilidades:

- (Orientación, Inclinación) : (β, α)
- (Horizontal) : (0)
- (Superficie perpendicular al vector solar): (n)
- (En el plano del generador): (I)

Por ejemplo, al escribir $B_h(0)$ leeremos irradiación directa (forma) horaria (tiempo) en el plano horizontal (lugar), mientras que $G_{d,m}(I)$ se lee media mensual (período) de la irradiación global (forma) diaria (tiempo) en el plano generador (lugar).

3.4. Cálculo de componentes de radiación solar

Partiendo de los datos obtenidos de la página de ADRASE, de radiación media mensual, debemos calcular las componentes de irradiación directa y difusa en el plano horizontal y después realizar el paso al plano inclinado.

Un esquema resumido del proceso se indica a continuación:

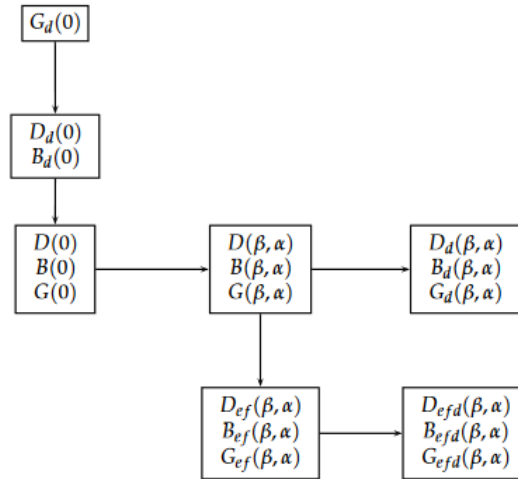


Figura 3.1: Procedimiento de cálculo (Figura 3.3 pág 31 [8])

En la figura 3.1 se muestra el proceso de cálculo que se debe seguir para obtener los valores de radiación efectiva incidente en el plano del generador, que después se utilizarán para calcular la energía que es capaz de entregar este.

Los pasos descritos son:

1. Separar la irradiación global diaria en el plano horizontal en sus componentes de irradiación difusa y directa.
2. Convertir la irradiación diaria a un perfil horario de irradiación global, directa y difusa en plano horizontal.
3. Pasar del perfil horario en el plano horizontal a un perfil con la inclinación y orientación indicada por el usuario.
4. Aplicar las pérdidas relacionadas con el ángulo de incidencia y el nivel de suciedad.
5. Volver a convertir el perfil horario a unos valores diarios de irradiación global, difusa y directa.

Para poder calcular la energía producida por un generador fotovoltaico, es necesario conocer la radiación solar que incide sobre la superficie de dicho generador. Para poder predecir la energía producida por éste en un tiempo futuro, el problema que se ha de resolver es estimar la irradiancia³ que recibirá, a partir del comportamiento de la radiación solar en ese lugar.

³**Irradiancia:** densidad de potencia de radiación solar incidente en una superficie. Unidades: $\frac{W}{m^2}$

En 1960, Liu y Jordan [9] expusieron una forma de caracterizar la radiación solar en un lugar, mediante el índice de claridad K_T . Éste índice es la relación entre la radiación global y la extra-terrestre, ambas en el plano horizontal. La expresión de el índice de claridad diario es:

$$K_{Td} = \frac{G_d(0)}{B_{0d}(0)} \quad (3.5)$$

mientras que el índice de claridad mensual es la relación entre las medias mensuales de irradiación diaria:

$$K_{Tm} = \frac{G_{d,m}(0)}{B_{0d,m}(0)} \quad (3.6)$$

En el código, el índice de claridad media se calcula con la siguiente expresión:

```
1  newData.meanValues.forEach((elem, index) => {
2    const Ktd = elem.meanGR / elem.B0d0
3  })
```

Listing 3.8: Índice de claridad diario

Para cada uno de los valores medios mensuales, dividimos el valor de radiación global media entre el $B_{0d}(0)$ calculado anteriormente.

Habiendo calculado el índice de claridad, podemos utilizar su valor para calcular la Fracción de radiación difusa en el plano horizontal ($F_D = \frac{D(0)}{G(0)}$). Para ello, podemos utilizar la ecuación de Page para aproximar el dicho valor conociendo el índice de claridad.

La ecuación que define éste valor es:

$$F_{Dm} = 1 - 1.13K_{Tm} \quad (3.7)$$

En código, la expresión es:

```
1  newData.meanValues.forEach((elem, index) => {
2    const Fd = 1 - 1.13 * elem.Ktd
3  })
```

Listing 3.9: Fracción de difusa

Donde elem.Ktd es el índice de claridad calculado en el paso anterior.

Al conocer, mediante la ecuación de Page el valor de la fracción de radiación difusa, podemos aplicar dicha relación para calcular los valores de radiación directa y difusa en el plano horizontal, a partir de la radiación global. Las dos ecuaciones necesarias para calcular estos valores son:

$$D_d(0) = F_D G_d(0) \quad (3.8)$$

$$B_d(0) = G_d(0) - D_d(0) \quad (3.9)$$

En código, las expresiones toma la siguiente forma:

```

1  newData.meanValues.forEach((elem, index) => {
2      const Dd0 = elem.Fd * elem.meanGR
3      const Bd0 = elem.meanGR - Dd0
4  })

```

Listing 3.10: Radiación directa y difusa

Para cada uno de los doce valores de radiación global media, aplicamos la fracción calculada anteriormente para obtener los dos valores de radiación en el plano horizontal.

3.5. Radiación en superficies inclinadas

Para poder obtener los valores de irradiación global en un plano inclinado a partir de valores de irradiación global en el plano horizontal se describe en la figura 3.1. El proceso de cálculo parte de los valores de irradiación global diaria en el plano horizontal, o en el caso de no tener estos valores disponibles, podemos utilizar la ecuación 3.7 para obtener las respectivas medias mensuales de irradiación difusa y directa en el plano horizontal. A continuación, para poder llevar a cabo las transformaciones al plano inclinado, debemos estimar los valores de irradiancia difusa, directa y global en el plano horizontal. Con estas estimaciones podemos calcular los valores correspondientes al plano inclinado.

Integrando los valores de irradiancia se obtienen las estimaciones de irradiación diaria difusa, directa y global en el plano horizontal. A estos valores les aplicaremos las pérdidas por suciedad, añadiéndole el apellido de “efectiva” a la irradiancia e irradiación. Será esta irradiación incidente efectiva la que se utilizará para el calculo de la energía producida por el generador.

3.5.1. Estimación de irradiancia a partir de irradiación diaria

Partiendo de los valores calculados anteriormente de irradiación diaria difusa, directa y global en el plano horizontal, podemos realizar la transformación al plano inclinado estimando el perfil de irradiancia correspondiente a cada valor de irradiación.

Como la variación solar durante una hora es relativamente baja, podemos suponer que el valor medio de irradiancia durante esa hora coincide numéricamente con irradiación horaria ($\frac{kWh}{m^2}$).

Por otro lado, mediante análisis de series temporales se ha demostrado que la relación entre la irradiancia y la irradiación difusa es equivalente a la existente entre la irradiancia y la irradiación extra-atmosférica.

$$r_D = \frac{D(0)}{D_d(0)} = \frac{B_0(0)}{B_{0d}(0)} \quad (3.10)$$

El factor r_D se puede calcular mediante la siguiente expresión:

$$r_D = \frac{\pi}{T} \cdot \frac{\cos(\omega) - \cos(\omega_s)}{\omega_s \cdot \cos(\omega_s) - \sin(\omega_s)} \quad (3.11)$$

Donde:

- T: Duración del día en horas

- ω_s : ángulo de amanecer, expresado en radianes
- ω : instante central de la hora correspondiente

La implementación en código es la siguiente

```

1  const rd = []
2  for (let h = -12; h < 12; h++) {
3      const hRad = Math.cos(deg2rad(h * 15))
4      const rdval = (Math.PI / 24) * ((hRad - Math.cos(elem.ws)) / (elem.ws *
5          Math.cos(elem.ws) - Math.sin(elem.ws)))
6      rd.push(rdval)
7  }
```

Listing 3.11: Cálculo de rD

Para cada uno de los valores horarios de -12 a 12, se calcula el ángulo horario multiplicando por 15 grados y se introduce en la ecuación, obteniendo 24 valores por cada uno de los meses. El mismo análisis muestra también la relación entre la irradiancia e irradiación global asimilable a una función dependiente de la hora solar.

$$r_G = \frac{G(0)}{G_d(0)} r_D \cdot (a + b(\omega)) \quad (3.12)$$

siendo:

$$a = 0,409 - 0,5016 \cdot \sin(\omega_s + \frac{\pi}{3}) \quad (3.13)$$

$$b = 0,6609 - 0,4767 \cdot \sin(\omega_s + \frac{\pi}{3}) \quad (3.14)$$

donde ω_s es negativa y está expresada en radianes.

En código, las expresiones anteriores se representan de la siguiente forma:

```

1  const rg = []
2  const a = 0.409 - 0.5016 * Math.sin(elem.ws + Math.PI / 3)
3  const b = 0.6609 - 0.4767 * Math.sin(elem.ws + Math.PI / 3)
4  let i = 0
5  for (let h = -12; h < 12; h++) {
6      const hRad = Math.cos(deg2rad(h * 15))
7      const rgval = rd[i] * (a + b * hRad)
8      rg.push(rgval)
9      i++
10 }
```

Listing 3.12: Cálculo de rG

3.5.2. Transformación al plano del generador

Una vez calculados los valores de irradiancia en el plano horizontal, el siguiente paso consiste en estimar los componentes de irradiancia en el plano del generador. La irradiancia directa es calculable mediante criterios geométricos, teniendo en cuenta el ángulo del cenit solar y el ángulo de incidencia en el generador.

Para obtener estos valores en el plano inclinado, lo primero que debemos calcular son los valores horarios de radiación global, directa y difusa en el plano horizontal, aplicando las dos relaciones, r_D y r_G que se han calculado en la sección anterior.

Para ello, emplearemos las siguientes ecuaciones:

$$D_h = r_D \cdot D_d(0) \quad (3.15)$$

$$G_h = r_G \cdot G(0) \quad (3.16)$$

$$G_h = G_h - D_h \quad (3.17)$$

En código, las expresiones serían las siguientes:

```

1  const hourlyValues = []
2  let hour = -12
3  const dawn = rad2deg(elem.ws) / 15
4  elem.dawn = dawn
5  for (let i = 0; i < 24; i++) {
6    const cosHour = Math.cos(deg2rad(hour * 15))
7    const cosWs = Math.cos(elem.ws)
8    if (cosHour > cosWs) {
9      const Dh = elem.rd[i] * elem.Dd0
10     const Gh = elem.rg[i] * elem.meanGR
11     const Bh = Gh - Dh
12     hourlyValues.push({
13       Bh, Dh, Gh, cosHour, cosWs,
14     })
15   } else {
16     hourlyValues.push({
17       Bh: 0, Dh: 0, Gh: 0, cosHour, cosWs,
18     })
19   }
20   hour++
21 }
22 elem.hourlyValues = hourlyValues

```

Listing 3.13: Cálculo de rG

Se puede observar que para las horas en las que el coseno es menor que el coseno del ángulo de amanecer, se toman valores nulos para las radiaciones.

Una vez obtenidos estos valores horarios de irradiancia en el plano horizontal, el siguiente paso consiste en llevarlo al plano del generador.

El planteamiento de ésta transición es el siguiente:

- **Irradiancia Directa** $B(\beta, \alpha)$: ecuación basada en geometría solar (ángulo cenital) y del generador (ángulo de incidencia).

$$B(\beta, \alpha) = B(0) \cdot \frac{\max(0, \cos(\theta_s))}{\cos(\theta_{zs})} \quad (3.18)$$

- **Irradiancia Directa** $D(\beta, \alpha)$: Existen dos modelos del estado de cielo, el isotrópico y el anisotrópico. En este caso, utilizaremos el modelo anisotrópico.

$$D(\beta, \alpha) = D^I(\beta, \alpha) + D^C(\beta, \alpha) \quad (3.19)$$

$$D^I(\beta, \alpha) = D(0) \cdot (1 - k_1) \cdot \frac{1 + \cos(\beta)}{2} \quad (3.20)$$

$$D^C(\beta, \alpha) = D(0) \cdot k_1 \cdot \frac{\max(0, \cos(\theta_s))}{\cos(\theta_{zs})} \quad (3.21)$$

- **Irradiancia de Albedo** $R(\beta, \alpha)$: Se considera nula por tener un porcentaje de aportación muy reducido.

Estos componentes sumados, forman la expresión de la irradiancia global en el plano del generador.

$$G(\beta, \alpha) = D(\beta, \alpha) + B(\beta, \alpha) + R(\beta, \alpha) \quad (3.22)$$

En la ecuación de la irradiancia directa, uno de los elementos que influyen en el cálculo es el ángulo de incidencia, cuya expresión es:

$$\begin{aligned} \cos(\theta_s) = & \text{sign}(\phi) \cdot [\sin(\beta) \cos(\alpha) \cos(\delta) \sin(\phi) - \\ & - \sin(\beta) \cos(\alpha) \cos(\phi) \sin(\delta)] + \\ & + \sin(\beta) \sin(\alpha) \cos(\delta) \sin(\omega) + \\ & + \cos(\beta) \cos(\delta) \cos(\omega) \cos(\phi) + \\ & + \cos(\beta) \sin(\delta) \sin(\phi) \end{aligned} \quad (3.23)$$

Donde:

- β : Inclinación del generador
- α : Orientación del generador
- ϕ : Latitud del emplazamiento
- δ : Declinación
- ω : Hora solar

Traducida a código, la expresión será:

```
1  for (let h = -12; h < 12; h++) {
2      const betaRad = deg2rad(newData.angle)
3      const alphaRad = deg2rad(newData.orientation)
4      const hRad = deg2rad(h * 15)
5      const latRad = deg2rad(newData.latitude)
6      const declRad = deg2rad(elem.decl)
7      const anglIncidenciaVal =
8          Math.sign(newData.latitude) *
9          (Math.sin(betaRad) * Math.cos(alphaRad) * Math.cos(declRad) *
10           Math.cos(hRad) * Math.sin(latRad) - Math.sin(betaRad) *
11           Math.cos(alphaRad) * Math.cos(latRad) * Math.sin(declRad)) +
12           Math.sin(betaRad) * Math.sin(alphaRad) * Math.cos(declRad) *
13           Math.sin(hRad) + Math.cos(betaRad) * Math.cos(declRad) *
14           Math.cos(hRad) * Math.cos(latRad) + Math.cos(betaRad) *
15           Math.sin(declRad) * Math.sin(latRad)
16      const anglIncidencia = rad2deg(Math.acos(anglIncidenciaVal))
17      const anglIncidencia = rad2deg(Math.acos(anglIncidenciaVal))
18      incid.push(anglIncidencia)
19      elem.incid = incid
20  }
```

Listing 3.14: Cálculo del ángulo de incidencia

Teniendo el valor del ángulo de incidencia, podemos proceder a calcular las dos componentes de irradiancia para el plano del generador.

```

1 newData.meanValues.forEach((elem, index) => {
2   for (let i = 0; i < 24; i++) {
3     const zenitRad = deg2rad(elem.zenit[i])
4     const incidRad = deg2rad(elem.incid[i])
5     const betaRad = deg2rad(newData.angle)
6     const numerator = Math.max(0, Math.cos(incidRad))
7     const denominator = Math.cos(zenitRad)
8     const Btilt = elem.hourlyValues[i].Bh * (numerator / denominator)
9     const k1 = elem.hourlyValues[i].Bh / elem.B00[i]
10    const DcTilt = elem.hourlyValues[i].Dh * k1 * (numerator / denominator)
11    const DiTilt = elem.hourlyValues[i].Dh * (1 - k1) * ((1 + Math.cos(betaRad)) / 2)
12    const Dtilt = DcTilt + DiTilt
13    const Gtilt = Btilt + Dtilt
14    elem.hourlyValues[i] = {
15      ...elem.hourlyValues[i],
16      Btilt, Dtilt, DcTilt, DiTilt, Gtilt, cosZenit: Math.cos(zenitRad),
17    }
18  }
19 })

```

Listing 3.15: Cálculo del ángulo de incidencia

Con estos cálculos obtenemos la irradiancia global incidente, es decir, en el plano del generador. El siguiente es añadirle el apellido de “efectiva” aplicándole las pérdidas por ángulo de incidencia y suciedad.

3.5.3. Pérdidas por ángulo de incidencia y suciedad

Al tratarse de sistemas estáticos, la radiación incidente en un módulo fotovoltaico está frecuentemente desviada de la normal a la superficie del módulo. Esta desviación, cuantificada por el ángulo de incidencia [eq. 3.23], es causa de pérdidas por reflexión.

Por otro lado, la suciedad acumulada en la superficie del módulo altera las propiedades angulares del mismo y reduce la capacidad de transmisión del vidrio.

Estos dos fenómenos reducen la irradiancia que es aprovechada por el módulo. Para el caso de la radiación directa, la expresión de irradiancia efectiva es la siguiente:

$$B_{ef}(\beta, \alpha) = B(\beta, \alpha) \cdot \frac{T_{sucio}(0)}{T_{limpio}(0)} \cdot (1 - FT_B(\theta_s)) \quad (3.24)$$

donde $FT_B(\theta_s)$ es el factor de pérdidas angulares para la irradiancia directa, calculable mediante la ecuación:

$$FT_B(\theta_s) = \frac{\exp(-\frac{\cos(\theta_s)}{a_r}) - \exp(-\frac{1}{a_r})}{1 - \exp(-\frac{1}{a_r})} \quad (3.25)$$

Los valores del coeficiente de pérdidas angulares deben ser determinados de forma experimental. En la tabla quedan recogidos algunos valores característicos de un módulo de silicio monocristalino convencional para los diferentes grados de suciedad. Además en la tabla también se recogen los valores de la transmitancia al interior del módulo en incidencia normal respecto a uno limpio

$$\frac{T_{sucio}(0)}{T_{limpio}(0)}.$$

Grado de suciedad	$\frac{T_{sucio}(0)}{T_{limpio}(0)}$	a_r	c_2
Limpio	1	0,17	-0.069
Bajo	0,98	0,20	-0.054
Medio	0,97	0,21	-0.049
Alto	0,92	0,22	-0.023

Cuadro 3.2: Coeficiente de pérdidas angulares y transmitancia relativa en incidencia normal para diferentes niveles de suciedad

El cálculo del coeficiente $FT_B(\theta_s)$, en código, toma este aspecto:

```

1 newData.TDirtTClean = TDirtTClean
2 newData.ar = ar
3 newData.c1 = c1
4 newData.c2 = c2
5 const FTB = (Math.exp(-Math.cos(tiltRad) / ar) -
6 Math.exp(-1 / ar)) / (1 - Math.exp(-1 / ar))

```

Listing 3.16: Cálculo del coeficiente $FT_B(\theta_s)$

Para la componente de isotrópica existe otra expresión que depende del ángulo de inclinación del generador, del coeficiente de pérdidas angulares y de dos coeficientes de ajuste c_1 y c_2 . El primero de ellos toma un valor constante $c_1 = \frac{4}{3\pi}$. El segundo depende linealmente de a_r según se recoge en la tabla. En ésta expresión el ángulo β está en radianes.

$$FT_D \simeq \exp\left[-\frac{1}{a_r} \cdot \left(c_1 \cdot \left(\sin \beta + \frac{\pi - \beta - \sin \beta}{1 + \cos \beta}\right) + c_2 \cdot \left(\sin \beta + \frac{\pi - \beta - \sin \beta}{1 + \cos \beta}\right)^2\right)\right] \quad (3.26)$$

Para estas componentes el cálculo de irradiancia efectiva es similar al de la irradiancia directa. Para la componente difusa circumsolar emplearemos el factor de pérdidas angulares de la irradiancia efectiva:

$$D_{ef}^I(\beta, \alpha) = D^I(\beta, \alpha) \cdot \frac{T_{sucio}(0)}{T_{limpio}(0)} \cdot (1 - FT_D(\beta)) \quad (3.27)$$

$$D_{ef}^C(\beta, \alpha) = D^C(\beta, \alpha) \cdot \frac{T_{sucio}(0)}{T_{limpio}(0)} \cdot (1 - FT_B(\beta)) \quad (3.28)$$

En código las ecuaciones toman esta forma:

```

1      const expFTD =
2          -(1 / ar) *
3          (c1 * (Math.sin(betaRad) + (Math.PI - betaRad - Math.sin(betaRad)) / (1 +
4              Math.cos(betaRad))) +
5              c2 * Math.pow(Math.sin(betaRad) + (Math.PI - betaRad -
6                  Math.sin(betaRad)) / (1 + Math.cos(betaRad)), 2))
7      const FTD = Math.exp(expFTD)
8
9      const Btilt = elem.hourlyValues[i].Btilt * T DirtTClean * (1 - FTB)
10     const DiTilt = elem.hourlyValues[i].DiTilt * T DirtTClean * (1 - FTD)
11     const DcTilt = elem.hourlyValues[i].DcTilt * T DirtTClean * (1 - FTB)

```

Listing 3.17: Cálculo de las componentes efectivas

Antes de pasar a calcular la energía producida por el generador configurado, se va a llevar a cabo una filtración de datos para eliminar los valores horarios entre el atardecer y el amanecer. Para ello, aplicamos este algoritmo a los datos calculados anteriormente

```

1      newData.meanValues.forEach((elem, index) => {
2          const significantValues = []
3          elem.hourlyValues.forEach((value, index) => {
4              if (Math.abs(value.hour) < Math.abs(elem.dawn)) {
5                  significantValues.push(value)
6              }
7          })
8          elem.significantValues = significantValues
9      })

```

Listing 3.18: Selección de los valores clave

Como se observa, los valores horarios menores al amanecer, en valor absoluto, no se introducen en la lista de valores significativos.

3.6. Cálculo de la energía producida por el generador

Una vez calculados los valores de radiación en el plano del generador, el siguiente paso será proceder a estimar la energía que es capaz de producir dicho generador conectado a red.

En el caso de la aplicación desarrollada en el proyecto, el objetivo es estimar la energía máxima que será capaz de producir un sistema fotovoltaico conectado a red.

3.6.1. Definición de un SFCR

Un Sistema Fotovoltaico Conectado a Red (SFCR) es un sistema cuya función es producir energía eléctrica en una condiciones específicas, de tal manera que ésta pueda ser inyectada en la red eléctrica convencional. Para ello, un SFCR se compone del propio generador fotovoltaico, un inversor DC/AC y un conjunto de protecciones eléctricas. Tal y como se muestra en la figura 3.2.

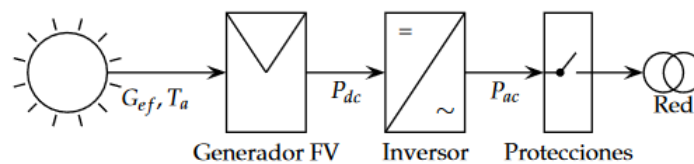


Figura 3.2: Esquema de un SFCR (Figura 6.1 pág 65 [8])

La energía producida por un SFCR es consumida total o parcialmente en el emplazamiento de la instalación, o en sus cercanías. La energía sobrante es inyectada a la red de distribución, y a cambio el propietario es compensado económicamente a través de sistemas de retribución tales como: la retribución con prima o el balance neto.

A diferencia de un sistema autónomo o de bombeo, el diseño de SFCR no necesita considerar un consumo mínimo a satisfacer. Con este mecanismo, el objetivo es que la producción anual del sistema sea la máxima posible sin tomar en consideración los consumos cercanos.

3.6.2. Configuración de los elementos del sistema

A la hora de estimar la energía producida por un SFCR, debemos definir las características nominales de los componentes que forman parte de este, como se muestra en la figura 3.2. Con el objetivo de reducir al máximo la complejidad de la aplicación de cara al usuario final (cuyos conocimientos de fotovoltaica, de media, serán reducidos) se han fijado unos valores standard para el módulo fotovoltaico, la configuración del generador y el inversor.

Estos valores se muestran en las siguientes tablas:

- **Configuración del módulo** El módulo elegido es uno disponible comercialmente, fabricado por JINKO SOLAR, modelo JKM320PP ⁴

G^*	$1000 \frac{W}{m^2}$
V_{oc}^*	46,4 V
I_{sc}^*	9,05 A
V_{mpp}^*	37,4 V
I_{mpp}^*	8,56 A
N_{cs}	12
N_{cp}	6
TONC	45 °
T_c	25 °
V_t	0,025V
m	1,3
Área modulo	$1,957 \times 0,992m^2$
Potencia Nominal	320 Wp

Cuadro 3.3: Configuración módulo standard

- **Inversor:**

k_0	0,01
k_1	0,025
k_2	0,05
Potencia	40 kW
V_{min}	420 V
V_{max}	750

Cuadro 3.4: Configuración del inversor

- **Generador** La configuración del generador será de **12** módulos en serie y **11** módulos en paralelo.

⁴Módulos solar JINKO SOLAR, modelo JKM320PP <https://autosolar.es/pdf/Ficha-Tecnica-Jinko-Solar-305-320W.pdf>

3.6.3. Funcionamiento de una célula solar

Para poder llevar a cabo la estimación de la energía producida por el generador, debemos conocer como influye factores como la radiación o la temperatura en una célula solar y en los valores de tensión y corriente que se alcanzan en dichas condiciones.

Existen dos puntos clave que definen una célula solar, la corriente de cortocircuito (I_{sc}) y la tensión de circuito abierto (V_{oc}). Estos dos parámetros suelen estar disponibles en la información asociada a una célula y se pueden relacionar mediante la siguiente ecuación:

$$I = I_{sc} \cdot \left[1 - \exp \left(\frac{e \cdot (V_{oc} - V)}{m \cdot k \cdot T_c} \right) \right] \quad (3.29)$$

Punto de máxima potencia

Superpuesta a la curva de corriente-tensión, la figura 3.3 incluye la relación entre la potencia y la tensión. Como se puede observar, existe un punto donde dicha potencia es máxima, antes de volver a experimentar una bajada. Las coordenadas de dicho punto vienen definidas por la condición $\frac{dP}{dV} = 0$. En dicho punto, la potencia entregada por la célula es máxima y será considerada como potencia nominal: $P_{mpp} = V_{mpp} \cdot I_{mpp}$. Las unidades de dicha potencia son los vatios pico (Wp), indicando así que se trata de la máxima potencia alcanzable.

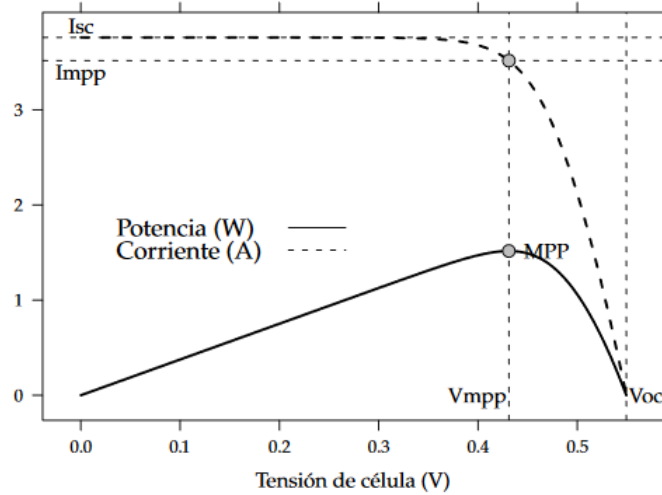


Figura 3.3: Curvas corriente-tensión (línea discontinua) y potencia-tensión (línea continua) de una célula solar ($T_a = 20^\circ\text{C}$ y $G = 800 \frac{\text{W}}{\text{m}^2}$) (Figura 4.6 pág 49) [8])

Como la célula funciona en corriente continua, su potencia es $P = V \cdot I$ y por tanto:

$$\begin{aligned} \frac{d(I \cdot V)}{dV} &= V \cdot \frac{dI}{dV} + I \cdot \frac{dV}{dV} \\ dP &= V + I \end{aligned} \quad (3.30)$$

En el punto de máxima potencia se cumplirá:

$$\frac{dI}{dV} = -\frac{I_{mpp}}{V_{mpp}} \quad (3.31)$$

Factor de forma y eficiencia

Como se puede observar en la figura 3.3, el área encerrada por el rectángulo $I_{mpp} \cdot V_{mpp}$ es inferior al definido por $I_{sc} \cdot V_{oc}$. La relación entre estas dos áreas, se conoce como factor de forma, y toma la siguiente expresión:

$$FF = \frac{I_{mpp} \cdot V_{mpp}}{I_{sc} \cdot V_{oc}} \quad (3.32)$$

Este factor toma valores entre 0,7 y 0,8 variando poco de unas células a otras. Conociendo los valores de I_{sc} y V_{oc} es posible calcular la potencia en el punto de máxima potencia, dado que $P_{mpp} = FF \cdot I_{sc} \cdot V_{oc}$.

Por otra parte, la calidad de una célula se puede cuantificar según la ecuación de la eficiencia de conversión:

$$\eta = \frac{I_{mpp} \cdot V_{mpp}}{P_L} \quad (3.33)$$

donde P_L representa la potencia luminosa que incide en la célula.

Influencia de la temperatura y la radiación

Es importante tener en cuenta, a la hora de entender el funcionamiento de una célula solar, la influencia de los dos principales factores externos: la temperatura ambiente y la iluminación incidente o radiación.

El aumento de la temperatura tiene una influencia notable en la tensión de circuito abierto de la célula según el valor dV_{oc}/dT_c . donde T_c es la temperatura de la célula, dependiente de la temperatura ambiente y la radiación incidente. La forma de calcular ésta temperatura depende de las características constructivas del módulo que encapsula a la célula. Si el fabricante no nos ofrece información sobre este valor, para células de silicio cristalino se común tomar el siguiente valor:

$$\frac{dV_{oc}}{dT_c} = -2,3 \frac{mV}{^{\circ}C} \quad (3.34)$$

Cálculo del punto de máxima potencia

Anteriormente, se ha mostrado la relación entre la temperatura y la irradiancia con la tensión de *circuito abierto* V_{oc} y la corriente de *cortocircuito* I_{sc} . Además, los fabricantes suelen incluir estos dos valores en el punto de máxima potencia en las condiciones estándar de medida.

Por tanto, para poder estimar el comportamiento de la célula en otras condiciones, es necesario trasladar estos parámetros a los valores de temperatura e irradiancia en dicha situación.

El procedimiento de cálculo consistirá en comenzar asumiendo que el factor de forma es constante, para calcular los valores de I_{sc} y V_{oc} en las condiciones de temperatura y radiación del emplazamiento. Posteriormente realizaremos los cálculos para tomar en cuenta la variación del factor de forma.

3.6.4. Asociación de dispositivos fotovoltaicos

Por defecto, las características eléctricas de una sola célula no son suficientes para alimentar una carga convencional. Por lo tanto, es necesario realizar agrupaciones en serie y en paralelo de varias de estas células en lo que conocemos como módulo fotovoltaico para entregar de esta manera los valores de tensión y corriente adecuados para la carga.

Además, un módulo fotovoltaico protege físicamente de la intemperie y aísla eléctricamente al conjunto de células, otorgando la rigidez mecánica necesaria.

Modelado de un módulo

Conociendo el comportamiento de una célula solar independiente, podemos trasladarlo al módulo teniendo en cuenta la configuración de éste, sobre todo el numero de células en serie y en paralelo que lo componen.

Para modelar el funcionamiento de un módulo podemos realizar las siguientes suposiciones:

- Los efectos de la resistencia en paralelo son despreciables.
- La resistencia serie es independiente de las condiciones de operación.
- La corriente fotogenerada (I_L) es igual a la corriente de cortocircuito.
- En cualquier condición de operación: $\exp(\frac{V+I_s}{V_t}) > 1$

En un módulo compuesto por N_{cs} células en serie y N_{cp} ramas en paralelo, y suponiendo que todas las células del módulo tienen características idénticas, la tensión de dicho módulo es $V_m = N_{cs} \cdot V_c$ y la corriente del módulo es $I_m = N_{cp} \cdot I_c$ siendo I_c y V_c la corriente y la tensión de una célula, respectivamente.

Dadas estas suposiciones, la curva de la corriente de módulo toma la siguiente expresión:

$$I_m = I_{sc} \cdot \left(1 - \exp\left(\frac{V_m - V_{oc} + I_m \cdot R_s}{V_t}\right) \right) \quad (3.35)$$

Supondremos que la corriente de cortocircuito depende exclusivamente y de forma lineal de la irradiancia:

$$I_{sc} = G_{ef} \cdot \frac{I_{sc}^*}{G_{stc}} \quad (3.36)$$

y la tensión de circuito abierto depende exclusivamente de la temperatura de célula, y decrece linealmente con ella:

$$V_{oc}(T_c) = V_{oc}^* + (T_c - T_c^*) \cdot \frac{dV_{oc}}{dT_c} \quad (3.37)$$

Como ya hemos mencionado anteriormente, si no hay información específica del por parte del fabricante, para módulos de silicio cristalino es habitual emplear el valor:

$$\frac{dV_{oc}}{dT_c} = -2,3 \frac{mV}{\text{célula}^\circ\text{C}} \quad (3.38)$$

Comportamiento térmico del módulo

En las ecuaciones anteriores se hace referencia a las condiciones estándar de medida. En estas condiciones, la temperatura de la célula es de 25°C. Sin embargo, la temperatura de operación de la célula es diferente y depende del balance de potencias del módulo.

El módulo recibe potencia luminosa, absorbiendo la fracción de ésta que no se refleja al exterior. De dicha fracción parte es transformada en energía eléctrica mientras que el resto se entrega en forma de calor al entorno.

Para simplificar podemos asumir que el incremento de la temperatura de la célula respecto a la ambiente depende linealmente de la irradiancia incidente en ésta. El coeficiente de proporcionalidad depende de muchos factores, tales como el modo de instalación del módulo, la velocidad del viento, la humedad del ambiente o las características constructivas del laminado que cubre las células.

Todos estos factores quedan recogidos en un valor único representado por la temperatura de operación nominal de la célula (NOCT o TONC), definida como aquella que alcanza una célula cuando su módulo trabaja en las siguientes condiciones:

- Irradiancia: $G = 800 \frac{W}{m^2}$.
- Espectro: el correspondiente $AM = 1,5$.
- Incidencia normal.
- Temperatura ambiente: $T_a = 20^\circ C$.
- Velocidad del viento: $v_v = 1 \frac{m}{s}$.

$$T_c = T_a + \frac{G_{ef} \cdot (TONC - 20)}{800} \quad (3.39)$$

Como se puede observar, en la temperatura de la célula influyen tanto la temperatura ambiente como la radiación global eficaz. Al trabajar con valores horarios, es necesario conocer el valor de la temperatura ambiente en cada hora del día.

El procedimiento para calcular los valores de temperatura ambiente horarios para cada uno de los meses en el emplazamiento indicado por el usuario no es trivial pues normalmente no suele haber un registro histórico de temperaturas. Para ello vamos a utilizar el proceso descrito en el documento [10].

En dicho documento se describe un proceso para desarrollar un perfil horario de temperaturas partiendo de una temperatura máxima y mínima.

Los pasos a seguir son:

1. **Obtención de la temperatura mínima y máxima:** Lo primero que necesitamos para poder realizar el perfil de temperatura es conocer los valores de temperatura mínima y máxima para cada uno de los meses. Para ello utilizaremos la API que nos proporciona AEMET a través de su plataforma de OPENDATA ⁵. Para ello buscamos en su base de datos la estación climatológica más cercana a las coordenadas del usuario y solicitamos la información necesaria, en este caso las temperaturas mensuales de un año entero.

⁵AEMET OPENDATA: Base de datos abierta de información climatológica, entre otras <https://opendata.aemet.es>

2. **Cálculo del ángulo de amanecer:** Una vez obtenidos los valores de temperatura mensuales, el siguiente paso es calcular el ángulo de amanecer para cada uno de los 12 días normales, como se ha hecho en pasos anteriores mediante la declinación. Para ello empleamos las dos ecuaciones:

$$\delta = 23,45 \cdot \sin\left(\frac{2\pi \cdot (d_N + 284)}{365}\right) \quad (3.40)$$

$$\begin{aligned} \cos(\omega_s) &= -\tan(\delta) \cdot \tan(lat) \\ \omega_s &= \arccos(\omega_s) \end{aligned} \quad (3.41)$$

donde:

- d_N : Día normal correspondiente 3.3.1
- lat : Latitud del emplazamiento

3. **Calcular los valores de T_m y T_r :** En el documento mencionado, el primer paso para llegar al perfil de temperaturas, una vez obtenidos los valores de T_{max} y T_{min} es calcular los dos valores intermedios:

$$T_m = \frac{T_{max} + T_{min}}{2} \quad (3.42)$$

$$T_r = \frac{T_{max} - T_{min}}{2} \quad (3.43)$$

4. **Calcular la matriz de T_a :** Una vez obtenidos los dos valores intermedios de T_m y T_r , el siguiente paso es calcular la matriz de valores de temperatura ambiente T_a para cada hora. Para ello el documento nos indica que debemos calcular tres valores que denomina a_1 , a_2 , a_3 . Utilizaremos estos tres valores para calcular T_1, T_2, T_3 , eligiendo uno u otro en función del ángulo solar. Las expresiones de estos tres valores se indican a continuación:

$$a_1 = \frac{12\pi \cdot (\omega_s - \omega)}{21\pi + 12 \cdot \omega_s} \quad (3.44)$$

$$a_2 = \frac{\pi \cdot (3\pi - 12\omega)}{3\pi - 12\omega} \quad (3.45)$$

$$a_3 = \frac{\pi \cdot (24\pi + 12 \cdot (\omega_s - \omega))}{21 \cdot \pi + 12 \cdot \omega_s} \quad (3.46)$$

donde ω es el ángulo solar y ω_s es el ángulo amanecer.

Con estos tres valores podemos calcular la otra terna de T_1, T_2 y T_3 con las siguientes ecuaciones:

$$T_1 = T_m - T_r \cdot \cos(a_1) \quad (3.47)$$

$$T_2 = T_m + T_r \cdot \cos(a_2) \quad (3.48)$$

$$T_3 = T_m - T_r \cdot \cos(a_3) \quad (3.49)$$

5. **Selección del valor clave:** en la matriz de temperatura ambiente introduciremos uno de los tres valores en función de si el ángulo solar es menor al del amanecer, si está entre en amanecer y $\frac{\pi}{4}$ o si es menor que $\frac{\pi}{4}$.

En código, todo este proceso toma la siguiente forma:

```

1  const wp = Math.PI / 4;
2  const Tm = (Tmax + Tmin) / 2;
3  const Tr = (Tmax - Tmin) / 2;
4  const Ta = [];
5  const decl = 23.45 * Math.sin((2 * Math.PI * (normalDays[index] + 284)) / 365);
6  const cosWs = -Math.tan(deg2rad(decl)) * Math.tan(deg2rad(latitude));
7  const ws = -Math.acos(cosWs);
8  for (let h = -12; h < 12; h++) {
9      const w = Math.cos(deg2rad(h * 15));
10     const a1 = (Math.PI * 12 * (ws - w)) / (21 * Math.PI + 12 * ws);
11     const a2 = (Math.PI * (3 * Math.PI - 12 * w)) / (3 * Math.PI - 12 * ws);
12     const a3 = (Math.PI * (24 * Math.PI + 12 * (ws - w))) / (21 * Math.PI + 12 * ws);
13     const T1 = Tm - Tr * Math.cos(a1);
14     const T2 = Tm + Tr * Math.cos(a2);
15     const T3 = Tm - Tr * Math.cos(a3);
16     if (w <= ws) {
17         Ta.push(T1);
18     } else if (w > ws && w <= wp) {
19         Ta.push(T2);
20     } else if (w > wp) {
21         Ta.push(T3);
22     }
23 }

```

Listing 3.19: Cálculo de la temperatura ambiente

Habiendo obtenido la matriz de temperatura ambiente, el siguiente paso es utilizar estos valores para calcular la temperatura de la célula T_c con la ecuación 3.39 necesaria para calcular la tensión de circuito abierto V_{oc} .

En código sería:

```

1  const cellTempProfiles = tempProfiles.map(({ hourlyTa, Tmax, Tmin }, indexA) => {
2    const meanValue = radiationData.meanValues[indexA];
3    const TCProfile = hourlyTa.map((temp, indexB) => {
4      const Gef = meanValue.hourlyValues[indexB].Gtilt * 1000;
5      const Tc = temp + (Gef * (moduleData.TONC - 20)) / 800;
6      return Tc;
7    });
8    return { TCProfile, month: index + 1, Tmax, Tmin };
9  });

```

Listing 3.20: Cálculo de la temperatura de célula

Como se observa en el código, para cada uno de los valores horarios utilizamos la G_{ef} en el plano inclinado (en el código aparece como G_{tilt}) y la temperatura ambiente calculada anteriormente (en el código es $temp$) para calcular la T_c .

Conociendo ya los valores horarios de temperatura de la célula, procedemos a calcular la tensión de circuito abierto V_{oc} utilizando la expresión 3.50 que, como se puede observar es la misma que 3.37 pero añadiendo el número de células en serie para convertirlo a tensión del módulo

$$V_{oc}(T_c) = V_{oc}^* + (T_c - T_c^*) \cdot \frac{dV_{oc}}{dT_c} \cdot N_{cs} \quad (3.50)$$

que en código quedaría de la siguiente forma:

```

1  const VocProfile = cellTempProfile.map(({ TCProfile }, index) => {
2    const VocArray = TCProfile.map((temp) => {
3      const Voc = moduleData.VocStar + (temp - moduleData.Tc)
4        * (-2.3 / 1000) * moduleData.Ncs;
5      return Voc;
6    });
7    return VocArray;
8  });
9

```

Listing 3.21: Cálculo de la tensión de circuito abierto

Por último, podemos calcular el valor de la corriente de cortocircuito I_{sc} que depende linealmente del valor de irradiancia según se indica en la expresión 3.36 que en código resulta de la siguiente forma:

```

1  const IscProfile = radiationData.meanValues.map(({ hourlyValues }) => {
2    const IscArray = hourlyValues.map(({ Gtilt }) => {
3      const IscValue = Gtilt * 1000 * (moduleData.IscStar / moduleData.Gstar);
4      return IscValue;
5    });
6    return IscArray;
7  });

```

Listing 3.22: Cálculo de la corriente de cortocircuito

Factor de forma variable

Los cálculos anteriores se han realizado suponiendo que el factor de forma FF es constante, por tanto, de la expresión 3.32, si suponemos que FF es constante, se podrían extraer los valores de tensión y corriente en el punto de máxima potencia ya que si

$$FF = FF^* \quad (3.51)$$

entonces

$$\frac{I_{mpp} \cdot V_{vmpp}}{I_{sc} \cdot V_{oc}} = \frac{I_{mpp}^* \cdot V_{vmpp}^*}{I_{sc}^* \cdot V_{oc}^*} \quad (3.52)$$

pudiendo así obtener los valores de I_{mpp} y V_{vmpp} .

Estos cálculos se realizaron como una primera aproximación que sirvió para comprobar si el proceso de cálculo de la radiación eficaz incidente eran correcto.

Sin embargo, hacer esta suposición nos aleja en cierta manera de ofrecer una estimación lo más próxima a la realidad posible.

Por tanto, para intentar aproximarnos lo máximo posible a una estimación acertada, eliminaremos dicha suposición y tendremos en cuenta la variación del factor de forma. Para ello, utilizaremos el siguiente proceso:

1. **Cálculo de la tensión térmica, V_t , a temperatura de la célula:** Calcularemos el valor de V_t a 25 ° C con la expresión:

$$V_{tn} = \frac{V_t \cdot (273 + 25)}{300} \quad (3.53)$$

```

1  const Vtn = (moduleData.Vt * (273 + 25)) / 300;

```

Listing 3.23: Cálculo de V_{tn}

2. **Cálculo de R_s^*** : El segundo paso consiste en calcular el valor de resistencia en serie con los valores STC:

$$R_s^* = \frac{\frac{V_{oc}^*}{N_{cs}} - \frac{V_{mpp}^*}{N_{cs}} + m \cdot V_{tn} \cdot \ln \left(1 - \frac{I_{mpp}^*}{I_{sc}^*} \right)}{\frac{I_{mpp}^*}{N_{cp}}} \quad (3.54)$$

En código quedaría:

```
1  const RsStar =
2    (moduleData.VocStar / moduleData.Ncs -
3     moduleData.VmppStar / moduleData.Ncs +
4     moduleData.m * Vtn * Math.log(1 - moduleData.ImpStar / moduleData.IscStar)) /
5    (moduleData.ImpStar / moduleData.Ncp);
```

Listing 3.24: Cálculo de R_s^*

3. **Cálculo de r_s** : Utilizando el valor R_s^* calculado en el paso anterior junto con los valores de V_{oc} y I_{sc} calculados anteriormente podemos calcular r_s que se utilizará más adelante en proceso. La ecuación que lo define viene dada por

$$r_s = R_s^* \cdot \left(\frac{N_{cs}}{N_{cp}} \cdot \frac{I_{sc}}{V_{oc}} \right) \quad (3.55)$$

y el equivalente en código es

```
1  const rs = VocProfile.map((VocArray, arrIndex) => {
2    return VocArray.map((VocValue, valIndex) => {
3      return RsStar * ((moduleData.Ncs / moduleData.Ncp)
4        * (IscProfile[arrIndex][valIndex] / VocValue));
5    });
6  });
```

Listing 3.25: Cálculo de r_s

Como se puede observar en la línea 4 del código, el valor de I_{sc} viene acompañado por dos índices entre corchete, esto es porque en todo momento estamos trabajando con valores horarios de cada día promedio correspondiente a cada uno de los 12 meses del año, por tanto, estamos trabajando una una matriz de 12x24, de ahí el uso de los dos índices.

4. **Cálculo de k_{oc}** : A continuación, utilizando los valores de temperatura ambiente obtenidos con anterioridad junto con la tensión de circuito abierto, calcularemos k_{oc} mediante la expresión:

$$k_{oc} = \frac{V_{oc} / N_{cs}}{m \cdot V_t \cdot \frac{T_c + 273}{300}} \quad (3.56)$$

En código tendría este forma:

```

1  const koc = VocProfile.map((VocArray, arrIndex) => {
2    return VocArray.map((VocValue, valIndex) => {
3      return VocValue / moduleData.Ncs / (moduleData.m * ((moduleData.Vt *
4        (cellTempProfile[arrIndex].TCProfile[valIndex] + 273)) / 300));
5    });
6  });

```

Listing 3.26: Cálculo de k_{oc}

Con éstos cálculos previos, éste método propone localizar el punto de máxima potencia de forma aproximada mediante las ecuaciones:

$$i_{mpp} = 1 - \frac{D_M}{k_{oc}} \quad (3.57)$$

$$v_{mpp} = 1 - \frac{\ln(k_{oc}/D_M)}{k_{oc}} - r_s \cdot i_{mpp} \quad (3.58)$$

donde:

$$D_M = D_{M0} + 2 \cdot r_s \cdot D_{M0}^2 \quad (3.59)$$

$$D_{M0} = \frac{k_{oc} - 1}{k_{oc} - \ln k_{oc}} \quad (3.60)$$

En código el proceso a seguir para llegar a los dos valores de i_{mpp} y v_{mpp} es el siguiente:

```

1  const DMO = koc.map((kocArray, arrIndex) => {
2    return kocArray.map((kocValue, valIndex) => {
3      return (kocValue - 1) / (kocValue - Math.log(kocValue));
4    });
5  });

```

Listing 3.27: Cálculo de D_{M0}

```

1  const DM = DMO.map((DMOArray, arrIndex) => {
2    return DMOArray.map((DMOValue, valIndex) => {
3      return DMOValue + 2 * rs[arrIndex][valIndex] * Math.pow(DMOValue, 2);
4    });
5  });

```

Listing 3.28: Cálculo de D_M

```

1  const impp = DM.map((DMArray, arrIndex) => {
2    return DMArray.map((DMValue, valIndex) => {
3      return 1 - DMValue / koc[arrIndex][valIndex];
4    });
5  });

```

Listing 3.29: Cálculo de i_{mpp}

```

1  const vmpp = impp.map((imppArray, arrIndex) => {
2    return imppArray.map((imppValue, valIndex) => {
3      return 1 - Math.log(koc[arrIndex][valIndex] /
4        DM[arrIndex][valIndex]) / koc[arrIndex][valIndex] -
5        rs[arrIndex][valIndex] * imppValue;
6    });
7  });

```

Listing 3.30: Cálculo de v_{mpp}

Por último, multiplicando los valores de i_{mpp} y v_{mpp} por I_{sc} y V_{oc} respectivamente, obtendremos los valores de I_{mpp} y V_{mpp} que serán los que utilizaremos para calcular la potencia entregada por el generador en el punto de máxima potencia.

```

1  const Vmpp = vmpp.map((vmppArray, arrIndex) => {
2    return vmppArray.map((vmppValue, valIndex) => {
3      return vmppValue * VocProfile[arrIndex][valIndex];
4    });
5  });
6  const Impp = impp.map((imppArray, arrIndex) => {
7    return imppArray.map((imppValue, valIndex) => {
8      return imppValue * IscProfile[arrIndex][valIndex];
9    });
10 });
11

```

Listing 3.31: Cálculo de V_{mpp} y I_{mpp}

Una vez calculados los valores de I_{mpp} y V_{mpp} podemos pasar a obtener los valores horarios de la potencia en el MPP, simplemente multiplicando los valores de corriente y tensión: $P_{mpp} = I_{mpp} \cdot V_{mpp}$.


```

1  const calculatePmpp = (ImppProfile, VmppProfile) => {
2    const PmppProfile = ImppProfile.map((ImppArray, dayIndex) => {
3      const PmppArray = ImppArray.map((ImppValue, hourIndex) => {
4        return ImppValue * VmppProfile[dayIndex][hourIndex];
5      });
6      return PmppArray;
7    });
8    return PmppProfile;
9  };
10

```

Listing 3.32: Cálculo de P_{mpp}

Cálculo final de potencias y energías

La potencia que obtenemos es la de un solo módulo. Para conocer la potencia que va a ser capaz de entregar el generador, debemos tener en cuenta su configuración de módulos en serie y en paralelo, y multiplicar estos valores de la siguiente manera:

```

1  const calculateGeneratorPower = (PmppProfile) => {
2    const PdcProfile = PmppProfile.map((PmppArray) => {
3      return PmppArray.map((PmppValue) => {
4        return PmppValue * generatordata.seriesModules * generatordata.parallelModules;
5      });
6    });
7    return PdcProfile;
8  };
9

```

Listing 3.33: Cálculo de la potencia del generador

Con este paso obtenemos la potencia horaria entregada por el generador fotovoltaico. El siguiente paso será pasar esa potencia a través del inversor y calcular la potencia a la salida de este. Para ello debemos llevar a cabo el siguiente desarrollo.

Lo primero, establecemos las expresiones de las potencias normalizadas Siendo P_{inv} la potencia nominal del inversor:

$$p_i = \frac{P_{DC}}{P_{inv}} \quad (3.61)$$

$$p_o = \frac{P_{AC}}{P_{inv}} \quad (3.62)$$

Para calcular la estos dos valores, utilizaremos un modelo empleado en inversores fotovoltaicos:

$$\eta_{inv} = \frac{p_o}{p_o + k_0 + k_1 p_o + k_2 p_o^2} \quad (3.63)$$

donde: k_0 , k_1 y k_2 son tres coeficientes ofrecidos por el fabricante del inversor.

Por otro lado, por definición de rendimiento, tenemos:

$$\eta_{inv} = \frac{p_o}{p_i} \quad (3.64)$$

Si despejamos las dos ecuaciones anteriores:

$$p_i = p_o + k_0 + k_1 p_o + k_2 p_o^2 \quad (3.65)$$

A partir de la expresión anterior obtenemos una ecuación de segundo grado con p_o como incognita:

$$k_2 p_o^2 + (k_1 + 1)p_o + (k_0 - p_i) = 0 \quad (3.66)$$

Y por último, si conocemos la p_o podemos calcular la potencia en corriente alterna que entrega el generador:

$$P_{AC} = p_o \cdot p_{inv} \quad (3.67)$$

En código, el desarrollo quedaría de la siguiente manera:

1. **Cálculo de p_i :** A partir de la potencia calculada anteriormente y conociendo la potencia del inversor, podemos calcular la p_i de esta manera:

```

1 const calculatePiProfile = (PdcProfile) => {
2   return PdcProfile.map((PdcArray) => {
3     return PdcArray.map((PdcValue) => {
4       return PdcValue / inverterData.power;
5     });
6   });
7 };

```

Listing 3.34: Cálculo de p_i

2. **Cálculo de p_o :** Utilizando p_i podemos despejar la ecuación de segundo grado y calcular el valor de p_o eligiendo la solución que cumpla que $p_o > 0$:

```

1 const calculatePoProfile = (PiProfile) => {
2   return PiProfile.map((PiArray) => {
3     return PiArray.map((PiValue) => {
4       if (PiValue <= 0) {
5         return 0;
6       }

```

```

7      const a = inverterData.k2;
8      const b = inverterData.k1 + 1;
9      const c = inverterData.k0 - PiValue;
10     const s1 = (-b + Math.sqrt(b * b - 4 * a * c)) / (2 * a);
11     const s2 = (-b - Math.sqrt(b * b - 4 * a * c)) / (2 * a);
12     return s1 > 0 ? s1 : s2;
13   });
14 });
15 };

```

Listing 3.35: Cálculo de p_o

3. **Cálculo de P_{inv} :** Habiendo calculado la p_o , podemos despejar la potencia generada por el generador (P_{inv}) de la ecuación 3.62:

```

1 const calculatePacProfile = (PoProfile) => {
2   return PoProfile.map((PoArray) => {
3     return PoArray.map((PoValue) => {
4       return PoValue * inverterData.power;
5     });
6   });
7 };

```

Listing 3.36: Cálculo de P_{AC}

Finalmente, conociendo las potencias en corriente continua y en alterna, podemos calcular la energía mensual y anual que es capaz de producir el generador configurado. Para ello, en el caso del balance mensual, sumaremos los 24 valores horario que hemos obtenido, y los multiplicaremos por 30.

Por otro lado, para calcular la energía anual, simplemente sumaremos los 12 valores de energía mensual que hemos calculado.

```

1 const calculateMonthlyEac = (PacProfile) => {
2   return PacProfile.map((PacArray) => {
3     return (
4       30 * PacArray.reduce((total, currentValue) => {
5         return total + (currentValue > 0 ? currentValue : 0);
6         EacCell.textContent = MonthlyEac[index];
7       }, 0))));

```

Listing 3.37: Cálculo de la energía mensual en alterna

```
1 const calculateMonthlyEdc = (PdcProfile) => {  
2   return PdcProfile.map((PdcArray) => {  
3     return (  
4       30 * PdcArray.reduce((total, currentValue) => {  
5         return total + (currentValue > 0 ? currentValue : 0);  
6       }, 0)  
7     });  
8   });  
9 };
```

Listing 3.38: Cálculo de la energía mensual en continua

```
1 const calculateAnnualEnergy = (MonthlyEnergy) => {  
2   return MonthlyEnergy.reduce((total, currentValue) => {  
3     return total + (currentValue > 0 ? currentValue : 0);  
4   }, 0);  
5 };
```

Listing 3.39: Cálculo de la energía anual

Capítulo 4

Ejemplo práctico de aplicación

Capítulo 5

Conclusión

Bibliografía

- [1] Unión Española Fotovoltaica. 2019. Informe anual 2019: El sector fotovoltaico impulsor de la transición energética. https://unef.es/wp-content/uploads/dlm_uploads/2019/09/memoria_unef_2019-web.pdf.
- [2] PVSyst. PVSyst SA. <https://www.pvsyst.com/download-pvsyst/>.
- [3] CalculationSolar. 2014. <http://www.calculationsolar.com/>.
- [4] SISIFO.Instituto de Energía Solar. <https://www.sisifo.info/es/default>.
- [5] PVGIS. European Commission, Joint Research Centre. https://re.jrc.ec.europa.eu/pvg_tools/es/tools.html.
- [6] System Advisor Model. NREL, US Department of Energy <https://sam.nrel.gov/>.
- [7] Oscar Perpiñán (2012). solaR: Solar Radiation and Photovoltaic Systems with R, Journal of Statistical Software, 50(9), 1-32. <http://www.jstatsoft.org/v50/i09/>.
- [8] Perpiñán, O. 2018. Energía Solar Fotovoltaica. <http://oscarperpinan.github.io/esf/>.
- [9] B. Y. H. Liu y R. C. Jordan. "The interrelationship and characteristic distribution of direct,diffuse, and total solar radiation". *Solar Energy* 4 (1960), págs. 1-19
- [10] Estimating average daytime and daily temperature profiles within Europe. Thomas A.Huldm Marcel Šúri, Ewan D.Dunlop, Fabio Micale 2006 <https://www.sciencedirect.com/science/article/abs/pii/S1364815205001593>