

Trabajo Fin de Grado
Ingeniería Electrónica, Robótica y Mecatrónica

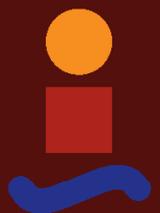
Diseño de aplicación de monitorización de
microrredes renovables domésticas

Autor: Alejandro Castillo Puerto

Tutor: Carolina Albea Sánchez

Dpto. Ingeniería de Sistemas y Automática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2024



Trabajo Fin de Grado
Ingeniería Electrónica, Robótica y Mecatrónica

Diseño de aplicación de monitorización de microrredes renovables domésticas

Autor:

Alejandro Castillo Puerto

Tutor:

Carolina Albea Sánchez

Profesor titular

Dpto. Ingeniería de Sistemas y Automática

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2024

Proyecto Fin de Carrera: Diseño de aplicación de monitorización de microrredes renovables domésticas

Autor: Alejandro Castillo Puerto

Tutor: Carolina Albea Sánchez

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

Presidente:

Vocal/es:

Secretario:

Acuerdan otorgarle la calificación de:

El Secretario del Tribunal

Sevilla, 2024

A mi abuelo Puerto

Agradecimientos

Con este proyecto doy uno de los pasos finales para acabar mis estudios. En todos estos años he podido experimentar una cantidad de emociones que llevo en mi memoria. He vivido el éxito y el orgullo, pero también la ansiedad y la frustración. La mayor enseñanza que me he podido llevar es que ser amigo de la constancia te lleva siempre a buen puerto.

Doy gracias a mis padres que vieron la importancia de educarme, no solo académicamente, si no en valores. Me han apoyado en cada decisión de mi vida y sin ellos la historia sería diferente. Mi padre me dio la inspiración de elegir esta carrera y mi madre me dio la capacidad para estudiarla. A mi hermana Ángela por ser una fuente incondicional de cariño, de risas, bondad y compañía. A mis tías, tíos, primos y demás familiares.

Gracias a mis abuelos, que nos brindaron a toda la familia la oportunidad de vivir con más comodidades y posibilidades que las que ellos tuvieron. A mi abuela Castillo y su alegría eterna. A mi abuelo Puerto que logró proliferar desde lo más humilde a través del esfuerzo, formarse por la vía autodidacta a través de la curiosidad y priorizar a su familia ante todas las cosas. La carrera empezó con él presente y presumo haber mantenido aquel “*Pórtate bien, estudia mucho y disfruta*”. Es el mejor regalo que me pudo hacer.

A mis amigos, confidentes y compañeros de aventuras. A *Koke* y a *Nono* con los que he tenido la suerte de tenerlos de protagonistas en mis anécdotas. Me concedo el lujo de llamarles familia. A mis compañeros de piso *Sema*, *Macarena*, *Manolo* y *David* que hicieron de la convivencia un disfrute y un hogar. En definitiva, a todos aquellos compañeros de instituto, universidad, amigos de la infancia y todo aquél que haya disfrutado un ratito con mi compañía.

A todas aquellas personas que me vieron crecer, a los médicos y profesionales que miraron por mí durante tantos años y que hacen que viva una vida plena y feliz. A Lebrija, mi casa. No puedo haber tenido más suerte de nacer aquí.

Por último, quiero dar las gracias a mi tutora Carolina. Ella me ha hecho descubrir nuevas áreas apasionantes de la ingeniería y se ha adaptado a mis horarios y circunstancias de una forma paciente. Gracias por tu disponibilidad y consejos. Carolina es un ejemplo de algunos otros docentes que llevaron el arte de la docencia a otro nivel priorizando la curiosidad y la cercanía. A todos ellos que lograron inspirarme, este trabajo es vuestro.

Una vez más, gracias a todos. Las páginas de este libro llevan el nombre de cada uno de ustedes.

Alejandro Castillo Puerto

Lebrija, 2024

Resumen

En este proyecto simularemos una microred de suministro eléctrico de origen fotovoltaico aplicado a urbanizaciones, bloques de viviendas y grupos de viviendas en general. Cada hogar dispone de una instalación fotovoltaica acompañada de una batería. El objetivo es implementar una ley de control que se encargue de repartir los excedentes energéticos entre vecinos de forma proporcional a sus gastos particulares. Se estudiará el efecto sobre los estados de carga de las baterías y si la distribución eficiente de energía logra abaratar costes de instalación. Todo ello estará reflejado en la simulación de un software de monitorización especialmente diseñado para uso a nivel usuario envuelta en una interfaz sencilla y estética.

Abstract

This project aims to simulate a PV-sourced microgrid applied to urbanizations, housing blocks and housing groups in general. Each home has got a photovoltaic facility accompanied by a battery. The objective is to implement a control law that is responsible for distributing energy surpluses among neighbors in proportion to their particular expenses. The effect on the state of charge (SoCs) and whether efficient energy distribution manages to reduce installation costs will be studied. All of this will be reflected in the simulation of monitoring software specially designed for user-level use wrapped in a simple and aesthetic interface.

Índice

Agradecimientos

Resumen

Abstract

Índice

Índice de Tablas

Índice de Figuras

Índice de Ecuaciones

1	Introducción	1
1.1	<i>Motivación</i>	1
1.2	<i>Estado del arte</i>	1
1.3	<i>Microrredes</i>	4
1.4	<i>Objetivo</i>	5
2	Diseño de la microrred	7
2.1	<i>Entorno de trabajo</i>	7
2.1.1	Software utilizado	7
2.1.2	Hardware utilizado	7
2.1.3	Escenario de aplicación teórica	7
2.2	<i>Modelo inicial</i>	8
2.2.1	La potencia neta	8
2.2.2	La batería	9
2.2.3	El estado de carga	9
2.3	<i>Bloque de control</i>	10
2.4	<i>Esquema de control</i>	12
2.5	<i>Resultados</i>	13
3	Hacia una simulación realista	15
3.1	<i>Instalación fotovoltaica</i>	15
3.1.1	Especificaciones	15
3.1.2	Bloque PV Array	16
3.1.3	Irradiancia	18
3.1.4	Temperatura	21
3.1.5	Potencia generada	23
3.2	<i>Energía consumida</i>	24
3.2.1	Rutinas energéticas	24

3.2.2	Recreación del consumo	25
3.2.3	Potencias netas	28
3.3	<i>Batería</i>	29
3.4	<i>Resultados del modelo aproximado a la realidad</i>	31
3.5	<i>Desconexión de una microrred de la red</i>	34
4	Aplicación de monitoreo	37
4.1	<i>Motivación</i>	37
4.2	<i>Software utilizado</i>	38
4.3	<i>Interfaz de usuario gráfica (GUI)</i>	39
4.3.1	Estado de vivienda	39
4.3.2	Flechas de flujos	40
4.3.3	Panel de información	41
4.3.4	Panel de información	41
4.3.5	Panel de reproducción	41
4.3.6	Detalles gráficos	42
4.4	<i>Comportamiento de la app</i>	42
4.4.1	Parte no editable	42
4.4.2	Comunicación simulación-app	42
4.4.3	Código principal	43
5	Conclusiones	47
	Anexo	49
	Referencias	53

ÍNDICE DE TABLAS

Tabla 1.1 Desglose del consumo de energía primaria en España 2021 [2]	2
Tabla 1.2 Desglose del consumo de energía primaria en España 2022 [2]	2
Tabla 1.3 Regiones europeas con mayores kWh potenciales de media por m ²	3
Tabla 2.1 Especificaciones del computador	7
Tabla 3.3.1 Especificaciones del panel	16
Tabla 3.2 Evolución de la irradiancia en Sevilla (24h)	19
Tabla 3.3 Evolución de la temperatura en Sevilla (24h)	21
Tabla 4.1 Vectores dentro de <i>out</i>	43

ÍNDICE DE FIGURAS

Figura 1.1 Paneles fotovoltaicos domésticos	3
Figura 1.2 Batería de litio Huawei® LUNA2000 10kWh + BMS (PVP 5385€)	4
Figura 2.1 Logotipo de Matlab & Simulink	7
Figura 2.2 Modelo básico de una microrred aislada	8
Figura 2.3 Señal creada para simular la potencia neta	9
Figura 2.4 Interior del bloque de batería	9
Figura 2.5 Evolución del SOC durante 200 segundos	10
Figura 2.6 Esquema de conexionado	10
Figura 2.7 Bloque de control en Simulink del Hogar 1	11
Figura 2.8 Esquema de control de prueba	12
Figura 2.9 Esquema de hogares independientes	13
Figura 2.10 Comparativa de no control vs control distribuido	13
Figura 2.11 Potencias netas del modelo de prueba	14
Figura 3.1 Instalación fotovoltaica propia	15
Figura 3.2 Logotipo de JA Solar	15
Figura 3.3 Panel FV usado	16
Figura 3.4 Bloque PV Array	16
Figura 3.5 Panel de propiedades del PV Array	17
Figura 3.6 Curva característica I-V y P-V	17
Figura 3.7 Tipos de irradiancia	18
Figura 3.8 Sistema de Información Geográfica Fotovoltaica (PVGIS) de la Unión Europea	18
Figura 3.9 Evolución de la irradiancia a lo largo del día	19
Figura 3.10 Filtro de suavizado	20
Figura 3.11 Evolución del filtrado	20
Figura 3.12 Comparativa entre la señal original y la señal filtrada	21
Figura 3.13 Sensor junto con app colocado bajo panel	21
Figura 3.14 Comparativa entre la señal original y la señal filtrada	23

Figura 3.15 Modelo de la instalación fotovoltaica	23
Figura 3.16 Potencia generada a lo largo de un día	24
Figura 3.17 Monitorización de la potencia mediante app	25
Figura 3.18 Combinación de señales	26
Figura 3.19 Potencias combinadas en una gráfica	26
Figura 3.20 Perfil construido por max	27
Figura 3.21 Modelo de Simulink de la demanda de un hogar	27
Figura 3.22 Consumos de cada hogar	28
Figura 3.23 Modelado de Potencia Neta	28
Figura 3.24 Potencias netas	29
Figura 3.25 Bloque de batería	30
Figura 3.26 Implementación completa de la batería	31
Figura 3.27 Modelo realista sin control	31
Figura 3.28 Evolución de los SOC sin control	32
Figura 3.29 Modelo de microrred completo con control distribuido	33
Figura 3.30 SOC con control distribuido	33
Figura 3.31 SOC con control distribuido	34
Figura 3.32 Implementación de la desconexión	35
Figura 3.33 Desconexión del Hogar 1	35
Figura 3.34 Flujos de potencia durante la desconexión	36
Figura 4.1 Aplicación FusionSolar de Huawei	37
Figura 4.2 Entorno de App Designer	38
Figura 4.3 Interfaz gráfica de la aplicación	39
Figura 4.4 Iconos de compartir	40
Figura 4.5 Iconos de batería	40
Figura 4.6 Flechas de flujos	40
Figura 4.7 Panel de consumo neto	41
Figura 4.8 Panel de flujos de energía	41
Figura 4.9 (Dcha. a izq.) Sombra, fondo de cajón y fondo de app	42

ÍNDICE DE ECUACIONES

Ecuación 2-1 Potencia Neta	8
Ecuación 2-2 Modelo matemático del comportamiento de la batería	9
Ecuación 2-3 Algoritmo de control	11
Ecuación 2-4 Señales de control para cada miembro de la microrred	11
Ecuación 4-1 Cálculo de flujos en la microrred	44
Ecuación 4-2 Longitud de <i>out</i>	44
Ecuación 4-3 Ecuación de 2° grado obtenida	44
Ecuación 4-4 Número de hogares en función de la longitud de <i>out</i>	44

1 INTRODUCCIÓN

Una pequeña fracción de la energía del sol basta para alimentar a toda nuestra civilización.

- Elon Musk -

En este capítulo introductorio describiremos el concepto de microrred energética, el uso actual de las energías renovables y el modelo de vivienda de la sociedad moderna.

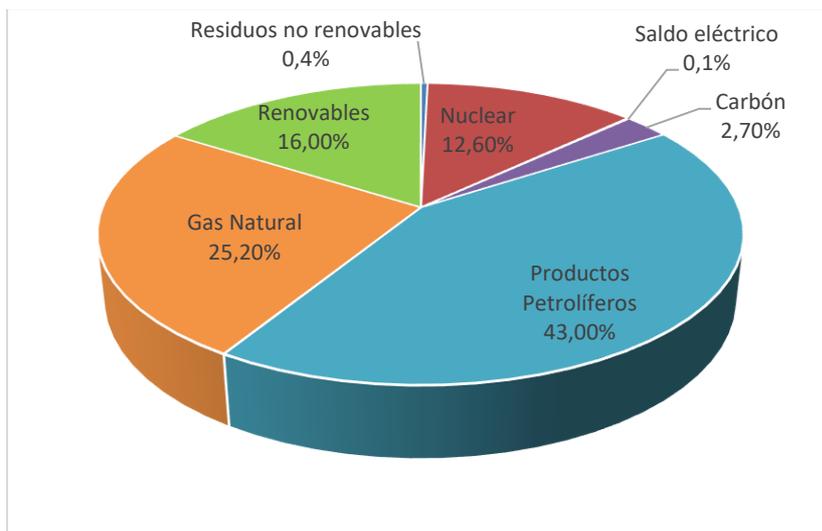
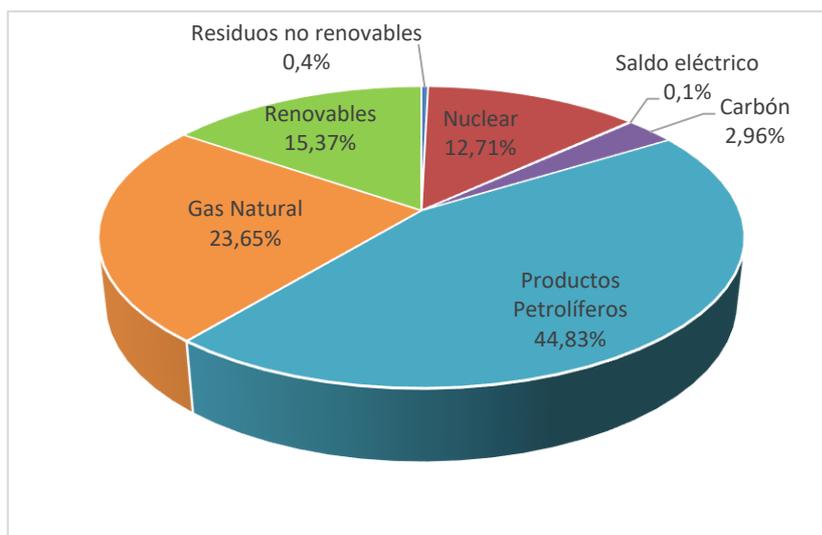
1.1 Motivación

Las energías renovables han experimentado un crecimiento exponencial a lo largo de los años. La búsqueda de alternativas que sustituyan a fuentes de energía tradicionales como el gas natural o el petróleo se ha convertido en un objetivo de extrema urgencia debido a los recientes conflictos internacionales. Al cambio climático se han sumado múltiples conflictos bélicos que dificultan la llegada de recursos energéticos al resto de países o, como efecto colateral, encarecen la generación de energías en las plantas de cada país. Estos factores han propiciado la instalación masiva de plantas de energía renovable tanto en el ámbito doméstico como en el industrial [1]. Desde el punto de vista doméstico, existen núcleos familiares que no disponen de recursos suficientes para acceder a una instalación fotovoltaica o que habitan entornos urbanos de difícil acceso propiciando el aislamiento de comunidades. La motivación del estudio es buscar soluciones para estas comunidades haciendo que sean energéticamente independientes y abaratando costes de instalación. Se propondrá un modelo energético colaborativo basándonos en el concepto de *microgrid*.

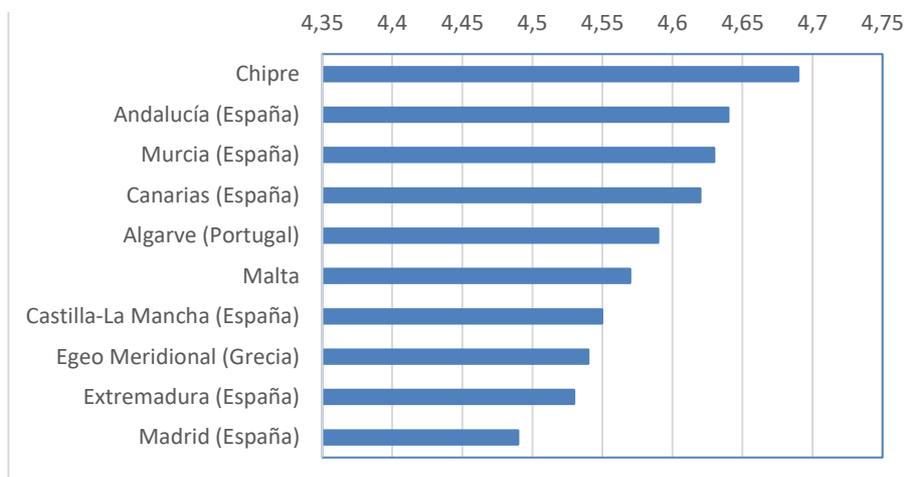
1.2 Estado del arte

España es un país geográficamente extraordinario para el uso de energías renovables. Su calidad de península hace que sea recipiente de vientos de origen oceánico acompañado de fuerzas mareomotrices en la totalidad de sus costas. El clima del país lo hace idóneo para el uso de instalaciones cuyo recurso es el Sol.

Según el último Balance Energético de España 2021 y 2022 publicado por el Ministerio para la Transición Ecológica y el Reto Demográfico [2] estos son los desgloses de consumo de energía primaria tanto en el año 2021 como en el 2022.

Tabla 1.1 Desglose del consumo de energía primaria en España 2021 [2]**Tabla 1.2** Desglose del consumo de energía primaria en España 2022 [2]

Como se puede observar, el consumo de energías provenientes de combustibles fósiles no ha hecho más que aumentar de un año para otro mientras que el uso de energías renovables ha disminuido tímidamente. Este hecho refuerza la idea de apostar por este tipo de energías verdes. Según un informe de *The World Bank* [3], España aglutina el mayor número de regiones con mayor potencial de aprovechamiento de energía solar dentro del top diez europeo.

Tabla 1.3 Regiones europeas con mayores kWh potenciales de media por m²

Con este informe se entiende que la principal baza de países como España para alcanzar la sostenibilidad y alimentar a la mayor cantidad de personas es el Sol. Para aprovechar la energía de nuestra estrella se han desarrollado sistemas termo solares y fotovoltaicos. En este proyecto nos centraremos en los sistemas fotovoltaicos.

La energía solar fotovoltaica es una forma de energía renovable que se obtiene al convertir la luz solar en electricidad. Esto se logra mediante el uso de paneles solares construido a base de dispositivos semiconductores llamados células fotovoltaicas. Estas células son sensibles al efecto de los fotones de luz sobre ellas, lo que mantiene a los electrones en movimiento y produce diferencias de potencial.

**Figura 1.1** Paneles fotovoltaicos domésticos

Según una nota de prensa de la Unión Fotovoltaica Española (UNEF), el autoconsumo fotovoltaico instalado en España creció un 108% respecto a 2021. [4] El número de empresas dedicadas a la instalación de kits de autoconsumo ha aumentado significativamente a lo largo de los últimos años. Sin embargo, a día de hoy, el acceso al autoconsumo e independencia eléctrica no es algo sencillo para todas las familias económicamente hablando.

La energía solar, al igual que cualquier otra fuente de energía verde, es una energía variable que depende de perturbaciones como son las nubes o la propia noche. Sin luz no hay energía y para ello se necesita

complementar la instalación con baterías de litio o plomo que almacenen excedentes de energía para poder usarlos durante estos tramos de no consumo. De otro modo no se podría alcanzar la independencia energética que estaríamos buscando. Esta adición no está exenta de cargos ya que se estima que una batería capaz de alimentar a un hogar en tramos de no consumo oscila entre los 3945€ y los 10612€ [5]. Lamentablemente, y a pesar de las ayudas gubernamentales, no todas las familias se pueden permitir hacer la inversión de una instalación solar con batería incluida que los haga energéticamente independientes.



Figura 1.2 Batería de litio Huawei ® LUNA2000 10kWh + BMS (PVP 5385€)

Por último, debemos tener en cuenta el tipo de vivienda de las familias. Según la institución *Funcas*, España es el país de la Unión Europea donde más gente vive en pisos. El 65,7% de la población reside en este tipo de viviendas, frente a un 34,1% que reside en casas. Esta realidad es mucho más acentuada en ciudades donde un 83,8% vive en pisos [6]. Al disponer de edificios con alturas mayores que su superficie horizontal, se hace imposible disponer de instalaciones que suplan las necesidades de todos los núcleos familiares que residen en estos bloques.

Los datos en países en vías de desarrollo son aún peores ya que estas comunidades se encuentran energéticamente aisladas. En el mundo viven 733 millones de personas sin acceso a electricidad [7]. Otras muchas cocinan con combustibles nocivos para la salud.

1.3 Microrredes

El Grupo de Microrredes del Departamento de Energía de los Estados Unidos define una microrred o *microgrid* como “un grupo de cargas interconectadas, sistemas de almacenamiento y recursos energéticos distribuidos dentro de límites eléctricos definidos que actúan como una entidad única controlable con respecto a la red. Además, tiene la posibilidad de conectarse y desconectarse de la red para permitir que funcione tanto en modo conectado como en modo isla” [8].

Las microrredes pueden operar en modo aislado o conectadas a la red principal, a través del punto de acoplamiento común (PCC). También pueden transicionar entre modos dependiendo de las necesidades del momento.

Las ventajas que ofrecen este tipo de instalaciones son variadas. Por un lado, una comunidad puede disponer de energía si sufre aislamientos debido a cortes de luz por mantenimiento o por desastres naturales. Otra opción es consumir de la red nacional en tramos de bajo costo, pero consumir de las baterías propias cuando los tramos sean de alta demanda y coste. Por otro lado, las microrredes pueden llegar a ser fuentes de ingresos para núcleos familiares al convertirse en generadores de energía, pudiendo vender los excedentes a compañías eléctricas. No hay que olvidar el ahorro en infraestructuras de transporte de energía y la reducción de emisiones de carbono.

Aun así, las microrredes enfrentan varias desventajas en la actualidad. Principalmente existen problemas técnicos debido a la sincronización de los flujos de potencia entre hogares. Los flujos bidireccionales son un problema a la hora de estabilizar la instalación. Hay que tener cuidado también con el manejo de corrientes continuas. Por otro lado, la inversión inicial y, sobre todo, el coste de las baterías hace que muchas familias vean inviable el autoconsumo. La burocracia y las penalizaciones al autoconsumo son un tema a tratar, pero

fuera de la temática de este proyecto.

Este tipo de redes locales han sido utilizadas ampliamente en centros hospitalarios para combatir cortes de luz repentinos. Las salas de quirófano pueden tener garantías ante pérdidas repentinas de energía a mitad de operación, poniendo en riesgo la vida del paciente. Además, supone un ahorro tanto económico como medioambiental.

La planificación de microrredes a nivel doméstico vino casi por obligación debido a grandes desastres naturales. En 2017, el huracán María arrasó con miles de hogares en Norte América dejando a la mayoría de la isla de Puerto Rico sin suministro eléctrico. El municipio de Adjuntas (Puerto Rico) ha implementado una microrred fotovoltaica para evitar incomunicaciones en el futuro [9]. Otros desastres meteorológicos propiciaron el desarrollo de una microrred mucho más avanzada en Montgomery (Maryland, EEUU) por la empresa multinacional Schneider Electric [10]. Todos estos ejemplos se mantienen dentro de un ámbito experimental y no se ha llegado a la estandarización que permita instalar estos modelos energéticos en otros lugares.

1.4 Objetivo

El objetivo principal de este proyecto es desarrollar una aplicación de monitorización para una microrred funcional instalada en una comunidad de viviendas. Las rutinas diarias de cada hogar varían debido a las distintas obligaciones laborales y familiares, lo que resulta en diferentes demandas de energía. La idea central es equipar cada hogar de un bloque, barriada o urbanización con un kit fotovoltaico que incluya baterías. Estas baterías almacenarán los excedentes de energía generados por los sistemas fotovoltaicos y los distribuirán entre los vecinos.

El reto consiste en diseñar un algoritmo de control que distribuya proporcionalmente los excedentes de cada hogar a aquellos vecinos que más lo necesiten. Por ejemplo, los hogares con baterías más cargadas donarán potencia a aquellos con un estado de carga bajo. Además, si una familia se encuentra de vacaciones, podrá donar los excedentes de su hogar a otras casas que demanden potencia. El objetivo final es homogeneizar el estado de carga de todas las baterías, prolongando su vida útil y potencialmente reduciendo los costes de instalación.

Para visualizar estos flujos de energía entre vecinos, se diseñará una aplicación intuitiva e ilustrativa que permita a los usuarios comprender fácilmente cómo se comparte la energía dentro de la comunidad.

2 DISEÑO DE LA MICRORRED

El diseño de una microrred simple requiere de varias etapas en las que poco a poco vamos enriqueciendo la instalación. El desarrollo incluye el estudio de la potencia neta que maneja una instalación fotovoltaica doméstica, el estudio de las rutinas y gastos energéticos de un hogar, el desarrollo del algoritmo de control, el comportamiento de una batería y la valoración de resultados. Se simplifica la microrred alejándonos de la implementación de electrónica de potencia porque repartimos el peso del proyecto entre la simulación de la microrred y la simulación de la aplicación.

2.1 Entorno de trabajo

2.1.1 Software utilizado

El software utilizado en todo el proyecto ha sido *Matlab R2023b* desarrollado por *MathWorks*. Específicamente usaremos su entorno de programación *Simulink*. En este entorno se pueden crear modelos basados en bloques de una forma mucho más visual y a muy alto nivel. Es útil para la simulación de sistemas a través del análisis de sucesos (modelo de caja negra). Su lenguaje de programación es propio, aunque resulta ser una modificación de *C* adaptada al cómputo matemático.



Figura 2.1 Logotipo de Matlab & Simulink

Para el proyecto que deseamos desarrollar, vamos a utilizar un *Toolbox* adicional a *Simulink* llamado *Simscape Electrical*. Esta herramienta nos permite modelar y simular sistemas electrónicos, mecatrónicos y eléctricos de una forma sencilla acercándonos a comportamientos similares a sistemas reales.

2.1.2 Hardware utilizado

Para poder realizar todas las simulaciones, manejar multitud de datos simultáneos y poder desarrollar una app que muestre animaciones, se ha utilizado un computador con las siguientes especificaciones:

Tabla 2.1 Especificaciones del computador

Parámetro	Valor
CPU	AMD Ryzen 5 5600G
Frec. CPU	3.9 – 4.4 GHz
GPU	Radeon Graphics 7
RAM	16 GB

2.1.3 Escenario de aplicación teórica

El modelo de Simulink implementará una red de microrredes conformada por cuatro hogares denominados:

“hogar 1”, “hogar 2”, “hogar 3” y “hogar 4”. Estos hogares forman parte de una urbanización y se establecen flujos de potencia bidireccionales entre todos ellos. Por ejemplo: El “hogar 1” puede compartir flujos de potencia con el 2, 3 y 4. Estos 3 a su vez lo pueden hacer con el 1.

Se estudiará un ciclo de 24 horas observando las transferencias de flujos de potencia dependiendo de los gastos energéticos y necesidades de cada hogar. Se supone que, al ser una urbanización, los niveles de radiación solares son los mismos para todas las instalaciones.

El algoritmo de control que se ha desarrollado ha sido un control distribuido. La elección de este algoritmo se basa en el aprovechamiento de dos propiedades muy atractivas: La robustez ante conexiones/desconexiones y la escalabilidad debido a que el sistema funciona bajo el principio “Pensar globalmente, actuar localmente”. Esta desconexión no supone un problema ya que la propiedad del algoritmo de control de robustez permite redistribuir los flujos de energía entre los hogares que siguen conectados a la red de microrredes. Además, la propiedad de escalabilidad permite que pueda extenderse a un número mayor de hogares.

2.2 Modelo inicial

Para poder escalar e interconectar varios hogares, hay que tener claro cuáles son los elementos claves con los que vamos a trabajar. En esencia estamos inyectando una potencia neta a una batería haciendo que esta se cargue o se descargue. La unidad más básica de una microrred aislada independiente se resume en este modelo de tres etapas.

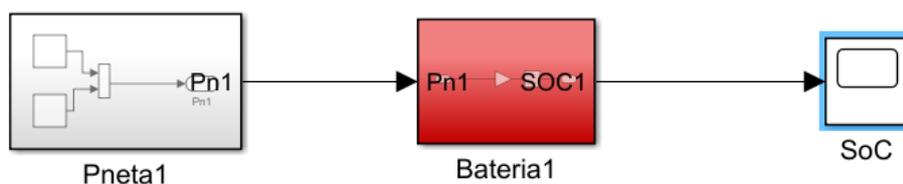


Figura 2.2 Modelo básico de una microrred aislada

2.2.1 La potencia neta

En este proyecto definimos la potencia neta como:

$$\text{Potencia Neta} = \text{Potencia Demandada} - \text{Potencia Generada}$$

Ecuación 2-1 Potencia Neta

La potencia demandada es aquella que el hogar necesita para alimentar todos los elementos eléctricos. La potencia generada es la producida, por ejemplo, por una instalación fotovoltaica. Si la potencia neta es positiva significa que el hogar está gastando más de lo que genera y si es negativa, tiene excedentes.

En primera instancia, y como prueba, generamos una señal sinusoidal multiplicada por un bloque de número aleatorio para darle el aspecto de señal con ruido. Esta señal simula la potencia neta.

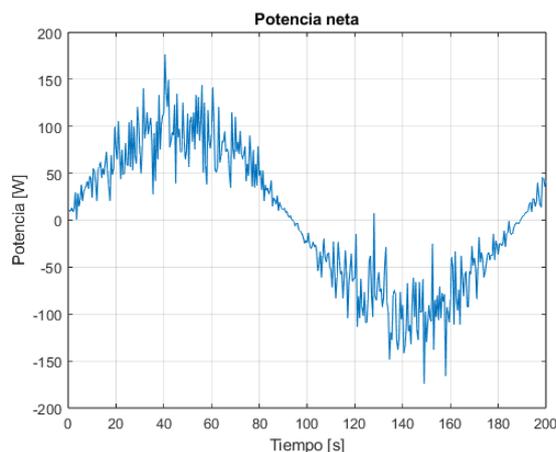


Figura 2.3 Señal creada para simular la potencia neta

2.2.2 La batería

Este bloque implementa un modelo matemático del comportamiento de la batería. Este está dictado por la ecuación siguiente:

$$\frac{dSOC}{dt} = - \frac{\eta}{C_{max}} \cdot P_N$$

Ecuación 2-2 Modelo matemático del comportamiento de la batería

Esta ecuación dice que la tasa de cambio del estado de carga de la batería (SOC) es proporcional a la potencia neta aplicada (P_N), ajustada por la eficiencia de la batería (η) y normalizada por su capacidad máxima en unidades de energía (C_{max}).

- La derivada del estado de carga respecto del tiempo ($dSOC/dt$) indica como cambia este con el tiempo, es decir, la velocidad con la que carga o se descarga la batería.
- El signo negativo (-) indica que cuando baja el estado de carga de la batería (variación negativa), la potencia neta es positiva. La relación es inversamente proporcional por lo que el signo negativo se encarga de invertir la dirección de cambio.
- El término η/C_{max} actúa como un factor de escala que ajusta la potencia aplicada en base al efecto real del estado de carga. Se tiene en cuenta la eficiencia de la conversión de energía y su capacidad total. En este modelo de prueba se establece un rendimiento del 80% y una capacidad máxima no realista pero ilustrativa de 17600Wh. Al hacer el cálculo, este factor de escala tendrá un valor de $-4,5455 \cdot 10^{-5}$.

Sabiendo que lo que recibimos es potencia neta y que queremos sacar el estado de carga, es trivial despejar este integrando la expresión del factor de escala con signo negativo por la potencia neta.

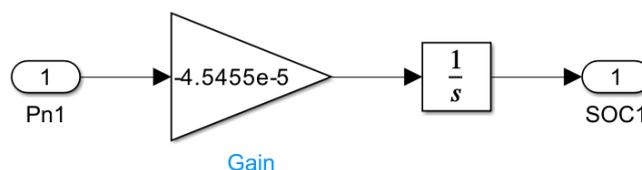


Figura 2.4 Interior del bloque de batería

2.2.3 El estado de carga

Una vez que el bloque de batería procesa la potencia neta, se muestran la evolución de la carga y descarga de

esta. En Simulink, el SOC se puede visualizar en tiempo real durante la simulación utilizando un *Scope*. El bloque de batería envía valores de 0 a 1.

Con el fin de aumentar la vida útil de las baterías, estas nunca deberían vaciarse o llenarse completamente. Por lo general, el estado de carga estará comprendido entre 0.1 y 0.9 o incluso entre 0.2 y 0.8.

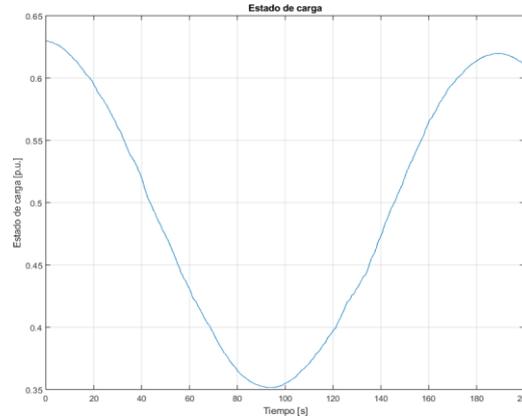


Figura 2.5 Evolución del SOC durante 200 segundos

2.3 Bloque de control

El bloque de control es aquel que gestiona la cantidad de potencia que se transfiere a cada hogar miembro de la microrred. Este control será distribuido, particularmente para lograr que las baterías ajusten sus SOC's en función de la diferencia entre su propio estado y el de sus vecinos.

El conexionado de la microrred, y por tanto la distribución para su control, fue propuesto en un principio con una configuración en anillo donde un hogar compartía energía únicamente con sus dos vecinos y estos vecinos lo hacían a su vez con otros vecinos. Así se hacía sucesivamente hasta el último hogar que compartía con el penúltimo vecino y con el primero cerrando la cadena. No obstante, este algoritmo se descartó debido a que el sistema controlado tenía infinitos mínimos locales, diferentes al valor de consenso de la red. Esto se debe, a la entrada de perturbación de cada sistema que son las potencias netas. Por tanto, este trabajo se realizó bajo la hipótesis que todos los nodos se conectan con todos.

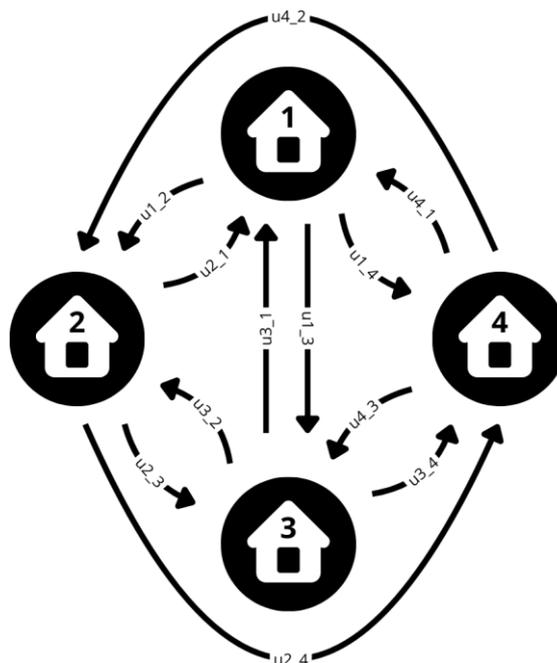


Figura 2.6 Esquema de conexionado

En el esquema de conexionado de la figura 2.6, las flechas muestran la dirección de los flujos de potencia acompañados de su anotación. La anotación de cada flecha es de la forma u_{i-j} siendo i el número de hogar de origen y j el número del hogar de destino. La u es la letra con la que se representa la señal de control.

El bloque de control se implementará en un bloque de *Simulink* llamado *MATLAB Function* que se explica a continuación:

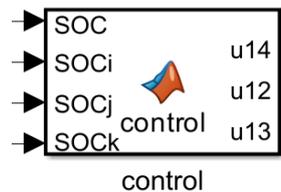


Figura 2.7 Bloque de control en Simulink del Hogar 1

Las entradas de la función para el caso del hogar 1 son:

- SOC: Es el estado de carga de la propia batería.
- SOCi, SOCj, SOck: Son los estados de carga de los tres miembros restantes de la microrred.
- Kc: Esta es una constante de control que determina cuánto influyen las diferencias de SOC en los ajustes de carga. No aparece visualmente como entrada ya que se introduce directamente en la consola de Matlab.

Las salidas de la función para el caso del hogar 1 son:

- u12, u13, u14: Son las señales de control que determinan como ajustar la potencia compartida entre el miembro actual y cada uno de los otros miembros. Cada señal indica como de lejos está con su vecina y el producto se multiplica por una constante K_c . Esta constante hace referencia a la velocidad de convergencia hacia el consenso. Cuanto mayor es esa constante, más rápido responde el sistema. El primer número de la nomenclatura se refiere al número del miembro y el segundo se refiere al miembro destinatario.

$$u_{14} = -K_c \cdot (SOC - SOC_i) ;$$

$$u_{12} = -K_c \cdot (SOC - SOC_j) ;$$

$$u_{13} = -K_c \cdot (SOC - SOCK) ;$$

Ecuación 2-3 Señales de control para cada miembro de la microrred

La ley de control total es aquella que logra un balance de potencias a lo largo de la simulación. El valor de u que vuelve a la batería en adición a la potencia neta proveniente de las placas fotovoltaicas queda de la siguiente forma:

$$u_{total} = -K_c \cdot ((SOC - SOC_i) + (SOC - SOC_j) + (SOC - SOCK))$$

Ecuación 2-4 Ley de control

2.4 Esquema de control

El esquema de control unirá el contenido visto en el apartado 2.2 y 2.3 de este capítulo de tal forma que podamos ver resultados simplificados en condiciones ideales, pero funcional. El propósito del modelo de Simulink es igualar los SOC's de todas las baterías de la microrred, promoviendo una gestión eficiente de la energía y asegurando que todos los hogares participen equitativamente en el almacenamiento y uso de la energía. Este control fomenta la eficiencia del sistema y la adaptación dinámica de este según los cambios en los patrones de consumo de energía de cada miembro de la microrred.

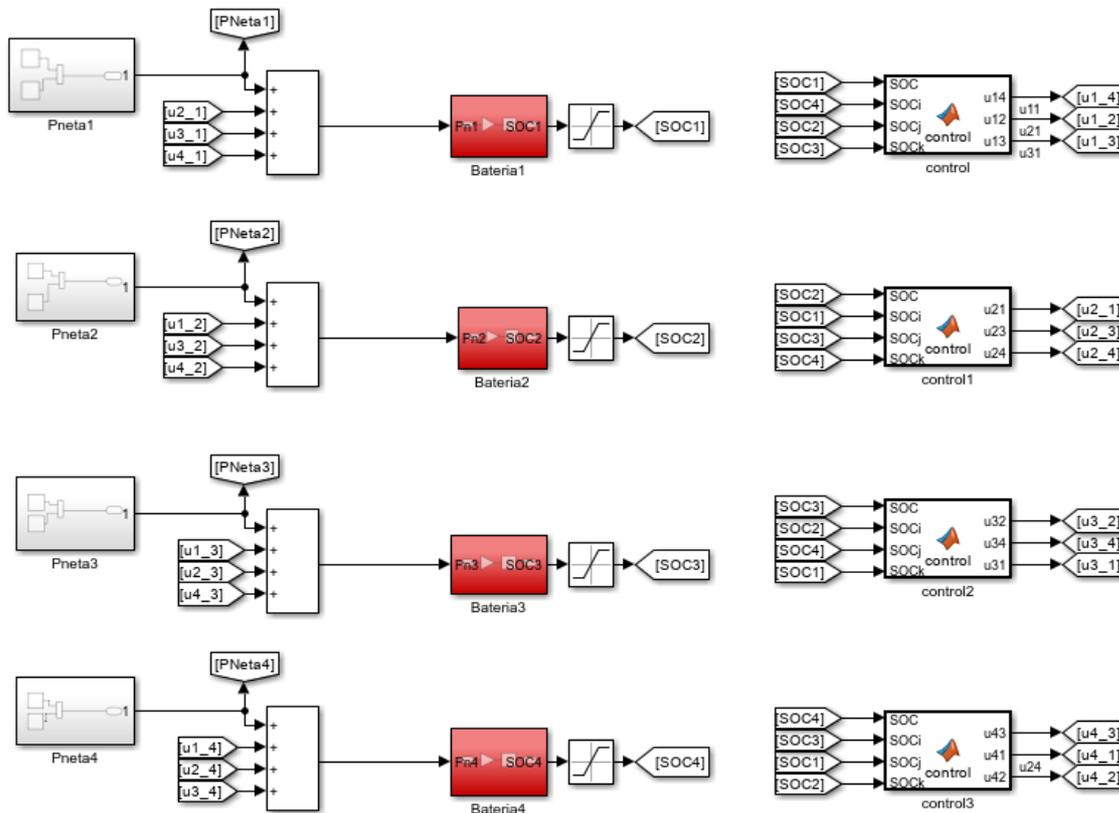


Figura 2.8 Esquema de control de prueba

La figura 2.8 muestra el esquema de control de prueba obedeciendo el esquema de conexionado de la figura 2.6. La denominación “de prueba” viene impuesta por la implementación de potencias netas de prueba que no reflejan patrones energéticos de ningún hogar común. Además, se trata a las baterías como modelos matemáticos donde no se tienen en cuenta comportamientos realistas.

En primer lugar, empezando por la izquierda se observa la suma de las señales de control proveniente de cada miembro de la microrred, así como la potencia neta del propio miembro. A este bloque de suma se le resta la propia dx del usuario para compensar su propia SOC como se comentó en el apartado anterior.

El bloque de suma está conectado al bloque de batería visto en el subapartado 2.2.2 del apartado 2.2 y finalmente su salida está saturada gracias a un bloque *Saturation* establecido entre 0.1 y 0.9 (límites del SOC).

Finalmente, disponemos de cuatro bloques de control explicados en el apartado 2.3 que gestionan el envío de las señales de control y de las dx de nuevo a los bloques de suma destinatarios.

2.5 Resultados

En esta sección mostramos algunos resultados de simular el modelo de la Fig. 2.8. Se establece un tiempo de simulación de 200 segundos y una K_c de 10000. Además, para analizar el desempeño del controlador, se ha creado un modelo adicional que representa a los cuatro hogares procesando sus potencias netas de forma independiente, es decir, sin estar interconectados en una red.

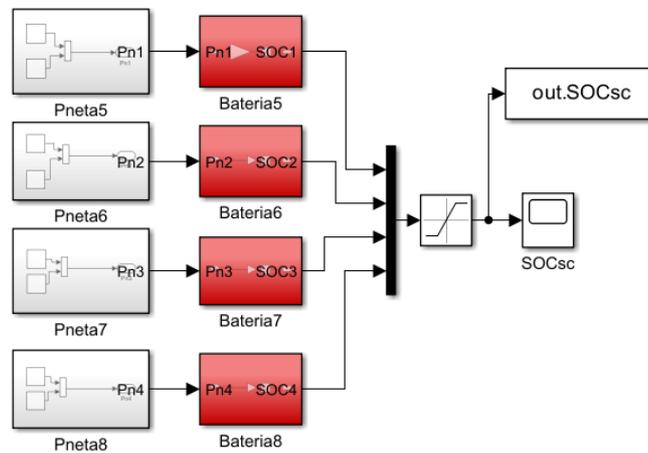


Figura 2.9 Esquema de hogares independientes

A continuación, se muestran los resultados obtenidos a través de Simulink:

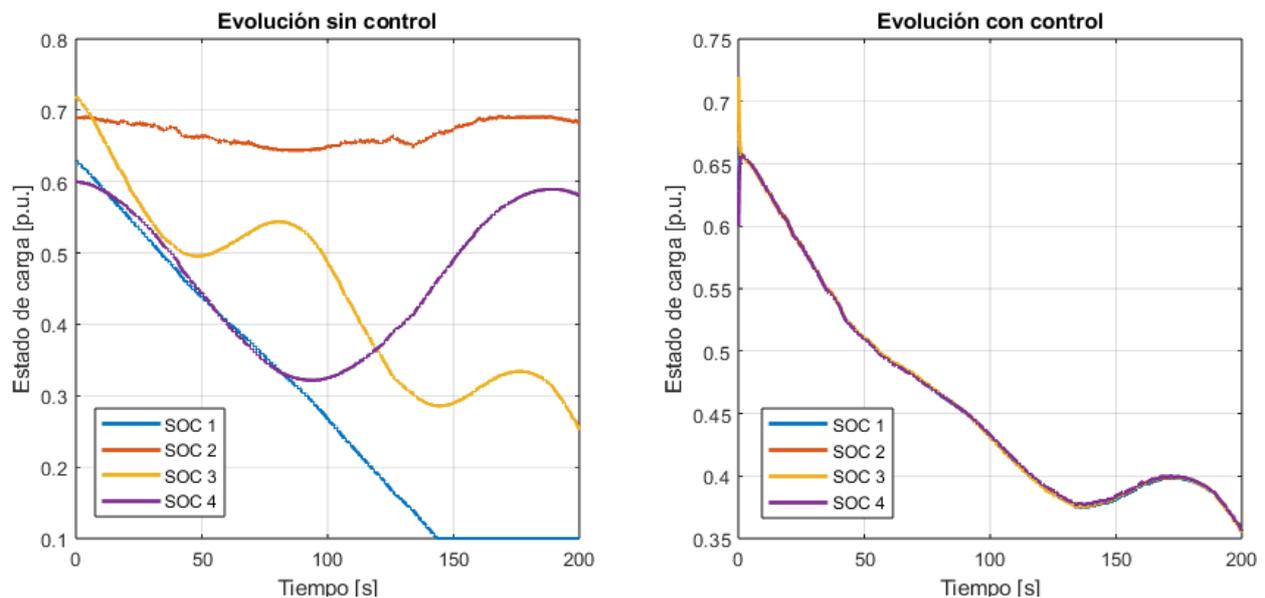


Figura 2.10 Comparativa de no control vs control distribuido

Con estos resultados mostramos los resultados de integrar un grupo de viviendas en una red de microrredes con control distribuido. En la gráfica de la izquierda, se pueden observar evoluciones de los SOC de las microrredes independientes, es decir, donde no se persigue el consenso. Es útil fijarnos en casos como el SOC1 donde la batería se descarga completamente alrededor del segundo 170 o el caso del SOC2 donde mantiene un valor de 0,7. Esta situación nos puede dar a entender que en el hogar 2 no hay nadie en ese momento.

Si nos fijamos en la gráfica de la derecha, la K_c establecida hace que los SOC de los cuatro hogares alcancen el consenso con un tiempo de establecimiento muy bajo. Si nos fijamos en el estado de carga final, las baterías se han quedado en torno al 35% de carga, cosa que, sin control, el hogar 1 no podría haberse beneficiado. Este usuario ha conseguido una reserva extra de estado de carga a costa de aprovechar otros estados de carga

mayores como puede ser los del hogar 2. Por tanto, el control diseñado genera una microrred más eficiente.

Con estos resultados se muestra que hay un mejor aprovechamiento de energía entre hogares, haciendo que los usuarios no tengan por qué conectarse a la red eléctrica ya que pueden beneficiarse de los excedentes de otros vecinos. En otras palabras, el control democratiza el uso de energía limpia, reduce costes y ayuda a la descarbonización.

Otra gráfica que se puede sacar de aquí es la de potencias netas.

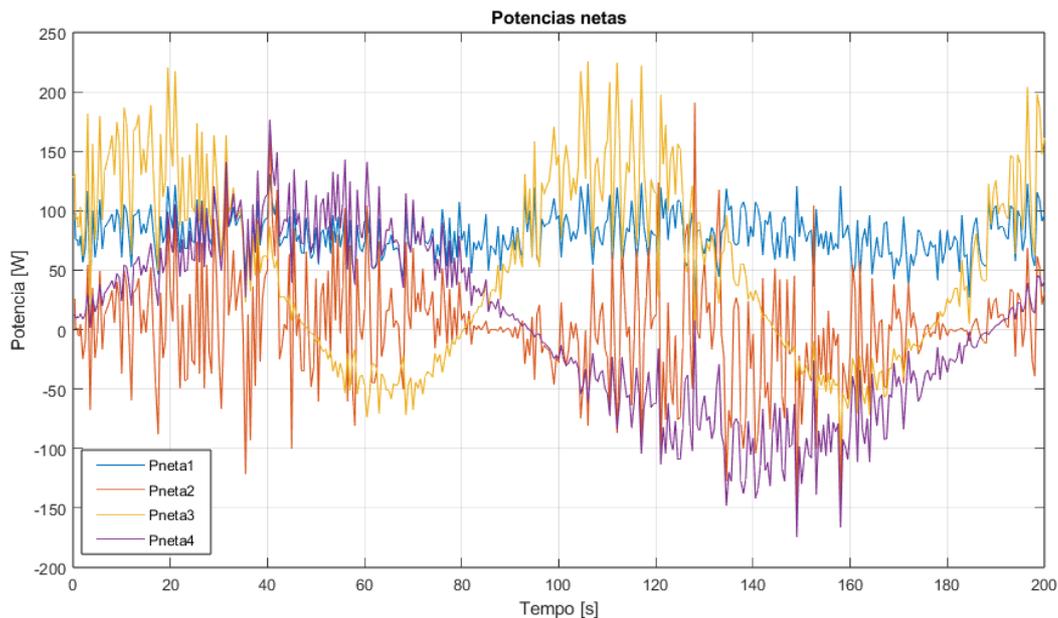


Figura 2.11 Potencias netas del modelo de prueba

Hemos visto una implementación del control que resulta ser funcional, pero no podemos decir que sea un modelo que se ajuste a la realidad. En la vida real, las personas tienen rutinas de consumo distintas que no se parecen en nada a las potencias netas modeladas. Los usuarios consumen energía las 24 horas del día y no solo durante 200 segundos, tampoco tenemos valores que fluctúen tanto y tampoco tenemos electrodomésticos que consuman pocas decenas de vatios. Además, las baterías tienen parámetros muy variados que van más allá de una simple fórmula matemática.

Por ello, en el capítulo siguiente abordaremos el modelado de la misma microrred teniendo en cuenta modelos de batería más parecidos a la realidad y patrones de consumo energético propio de las rutinas que ha establecido la sociedad moderna.

3 HACIA UNA SIMULACIÓN REALISTA

Este capítulo se centra exclusivamente en el afinamiento del modelo original. En él se hará un modelado realista de las distintas rutinas energéticas que pueden seguir varios miembros de una misma comunidad. Cada hogar dispondrá de un sistema fotovoltaico que simularemos mediante el *toolbox Simscape Electrical* (cosa que no se hizo en el capítulo anterior). Esta herramienta nos da la posibilidad de implementar bloques de baterías especializados donde podemos ajustar parámetros para lograr comportamientos mucho más realistas.

3.1 Instalación fotovoltaica

La fuente de energía renovable elegida es la fotovoltaica. Como comentamos en la introducción, el cambio climático, los recientes conflictos bélicos y la pandemia del COVID-19 han hecho que el precio de la electricidad alcance valores elevados. Estos sucesos, acompañados de las ayudas gubernamentales, han incentivado la instalación de placas fotovoltaicas en los hogares del país. [4]

En mi caso, he sido participe de ese aumento de instalaciones por lo que no podía perder la oportunidad de basar mi proyecto en los datos que refleja la instalación de mi domicilio. Así podemos tener una referencia sólida a la hora de catalogar nuestros datos de simulación como realistas.



Figura 3.1 Instalación fotovoltaica propia

La instalación se compone de 12 placas fotovoltaicas orientadas al sur con una estructura a modo de pérgola.

3.1.1 Especificaciones

Cada uno de los doce paneles fotovoltaicos son del modelo JAM72S20-460/MR de la compañía china JA Solar de 460W donde juntos pueden generar una potencia de 5520W. La superficie total que recibe la radiación solar es de 26,73m²



Figura 3.2 Logotipo de JA Solar

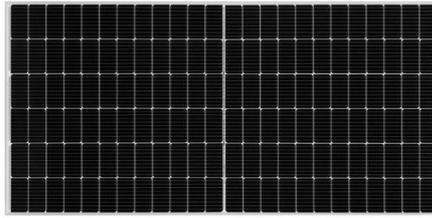


Figura 3.3 Panel FV usado

Tabla 3.3.1 Especificaciones del panel

Parámetro	Valor
Potencia máxima nominal (P_{max}) [W]	460
Voltaje de circuito abierto (V_{oc}) [V]	50.01
Tensión de potencia máxima (V_{mp}) [V]	42.13
Corriente de cortocircuito (I_{sc}) [A]	11.45
Corriente de potencia máxima (I_{mp}) [A]	10.92
Eficiencia del módulo [%]	20.7
Coeficiente de Temperatura de I_{sc} ($\alpha_{I_{sc}}$)	+0.044%/°C
Coeficiente de Temperatura de V_{oc} ($\beta_{V_{oc}}$)	-0.272%/°C
Condiciones de prueba estándar (STC)	Irradiancia 1000W/m ² Temperatura célula 25°C AM1.5G

3.1.2 Bloque PV Array

El bloque PV Array es una de las numerosas herramientas disponibles en *Simscape Electrical*. Se puede encontrar dentro de *Specialiced Power Systems*, en la pestaña *Sources*.

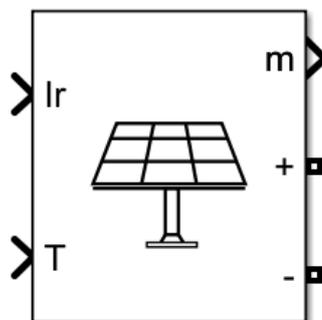


Figura 3.4 Bloque PV Array

Como se puede observar en la figura 3.4, el bloque PV Array dispone de dos entradas y dos salidas.

- La entrada I_r : recibe la cantidad de irradiancia solar que actúa sobre los paneles fotovoltaicos. Puede ser una constante para medir la tensión generada en un momento determinado o puede ser una señal que evolucione con el tiempo.
- La entrada T : recibe la temperatura a la que se encuentra la superficie de los paneles. Una temperatura superior a las condiciones de prueba estándar (25°C) repercutirá negativamente en la generación de energía ya que pierde eficiencia por la degradación del material. Las instalaciones solares se rigen por la incidencia de fotones, no por la cantidad de calor que reciben como erróneamente muchas personas creen.
- La salida m : entrega valores que pueden ser visualizados por un *display*. Estos valores pueden ser corrientes, tensiones, potencias...
- Las salidas $+ y -$: son las bornas positivas y negativas respectivamente por las que sale la tensión generada. Desde aquí podemos conectar otros elementos del circuito como resistencias, condensadores, convertidores de potencia...

Si abrimos el bloque, podemos ver una colección de modelos de paneles solares de distintas empresas, pero, en nuestro caso, no tienen el modelo que poseemos así que hemos introducido las especificaciones manualmente.

Array data	
Parallel strings	1
Series-connected modules per string	12
Module data	
Module:	User-defined
Maximum Power (W)	460.0596
Cells per module (Ncell)	144
Open circuit voltage Voc (V)	50.01
Short-circuit current Isc (A)	11.45
Voltage at maximum power point Vmp (V)	42.13
Current at maximum power point Imp (A)	10.92
Temperature coefficient of Voc (%/deg.C)	-0.272

Figura 3.5 Panel de propiedades del PV Array

Una vez introducidos los datos, podemos visualizar la curva característica I-V y P-V del módulo a condiciones de prueba estándar (STC).

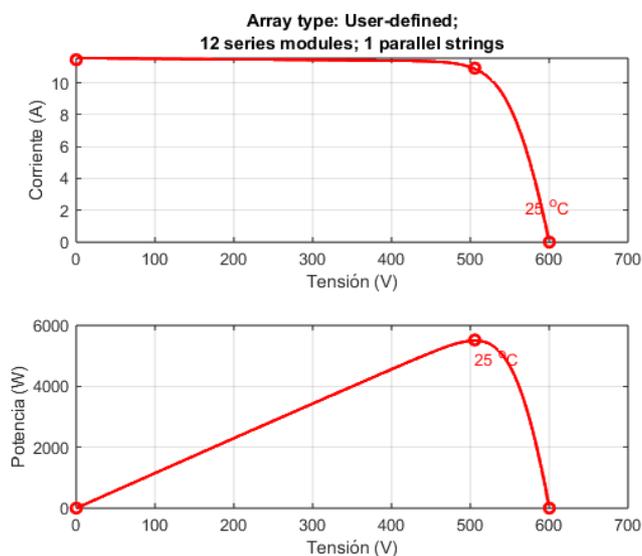


Figura 3.6 Curva característica I-V y P-V

3.1.3 Irradiancia

Según la RAE, la irradiancia se define como una magnitud que expresa el flujo radiante que incide sobre la unidad de superficie, y cuya unidad en el sistema internacional es el vatio por metro cuadrado. El sol es una estrella que incide radiación sobre nuestro planeta de modo que, cuanto más radiación incida sobre los paneles, más energía generarán. Existen tres tipos de irradiancia que acaban alimentando los paneles:

- **Radiación solar directa:** Es aquella que proviene directamente del sol.
- **Radiación solar difusa:** Es aquella que proviene, por el desvío de los rayos del sol al atravesar las nubes. En los días nublados en los que se reduce la cantidad de radiación directa este tipo de radiación aumenta de manera considerable.
- **Radiación solar reflejada:** Es aquella que “rebota” sobre superficies como edificios. No la tenemos en cuenta porque normalmente está apuntando al cielo.

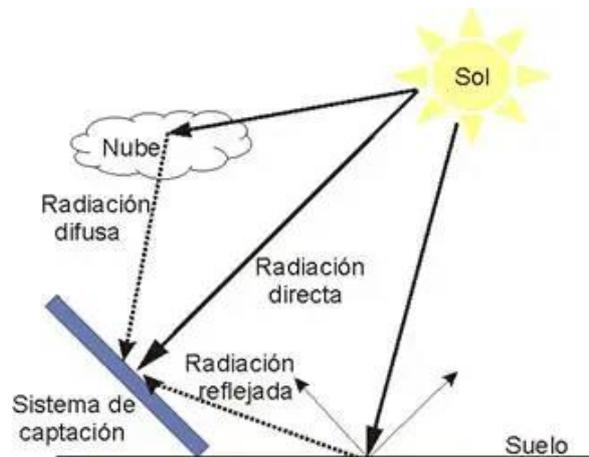


Figura 3.7 Tipos de irradiancia

La irradiancia global es la suma de todos los tipos de irradiancias y son los que tenemos en cuenta a la hora de introducirlos al bloque de Simulink.

La irradiancia no es la misma en todos los lugares del mundo. Esta aumenta a medida que nos acercamos a la línea del ecuador sobre todo la de tipo directa. Sin embargo, hay que tener en cuenta que otros factores, como la altitud, la estación del año, la nubosidad y la atmósfera, también pueden afectar la irradiancia en un lugar específico. Por ello hemos usado la herramienta interactiva del [Sistema de Información Geográfica Fotovoltaica \(PVGIS\) de la Unión Europea](#) para sacar un registro de las irradiancias de un día completo soleado de primavera en la ciudad de Sevilla [11].

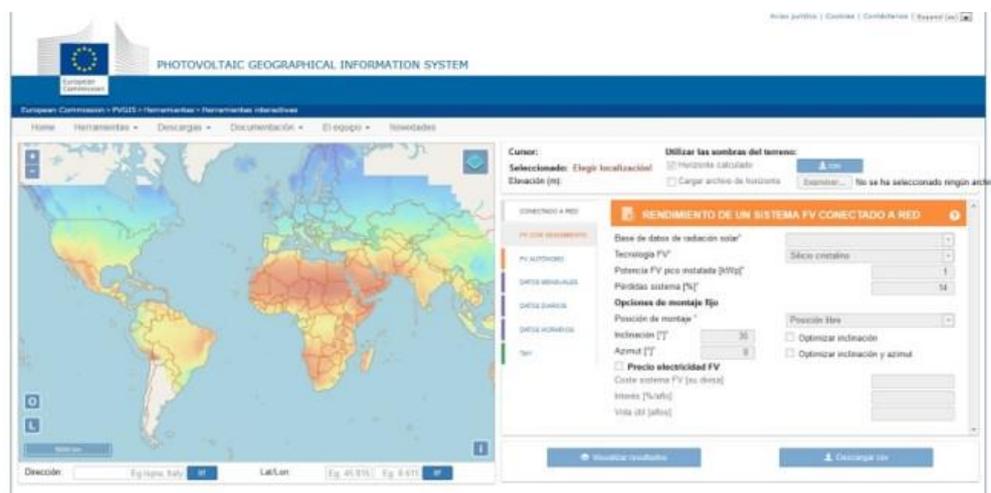


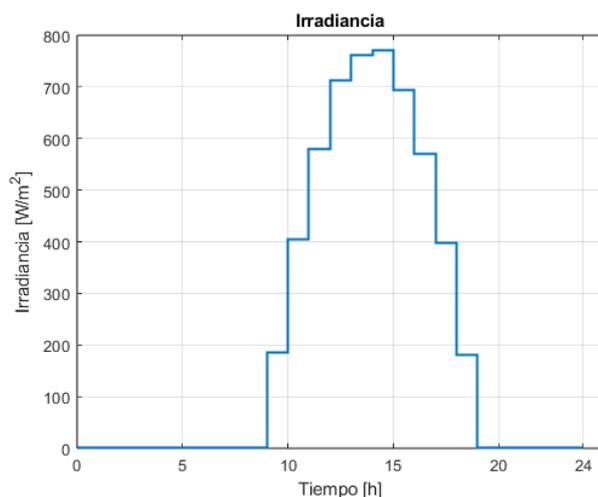
Figura 3.8 Sistema de Información Geográfica Fotovoltaica (PVGIS) de la Unión Europea

Tabla 3.2 Evolución de la irradiancia en Sevilla (24h)

Hora	Valor [W/m^2]
08:00	0
09:00	186.21
10:00	405.1
11:00	579.54
12:00	712.02
13:00	761.83
14:00	769.76
15:00	693.03
16:00	571.33
17:00	398.66
18:00	181.12
19:00	0.04
20:00	0
21:00	0

En la tabla anterior se ven los valores de irradiancia totales (directa + difusa) que se alcanzan cada hora en Sevilla un día de febrero. El resto de horas se entiende que es de noche por lo tanto no incide irradiancia sobre los paneles.

La manera en la que introducimos estos datos por el puerto *Ir* del bloque *PV Array* es mediante un *Repeating Sequence Stair*. Para tener una visualización clara de la evolución de la potencia generada y en vista a monitorizar las rutinas energéticas de los hogares, hemos establecido un tiempo de simulación de 24 segundos donde cada segundo será equivalente a una hora real.

**Figura 3.9** Evolución de la irradiancia a lo largo del día

La curva de irradiancia se podría afinar ya que este valor no va por escalones en la naturaleza. El ciclo solar tiene un movimiento continuo y no discreto. Este suavizado se puede realizar mediante la realización de un filtro sencillo justo antes de introducir los datos por el puerto Ir del bloque *PV Array*.

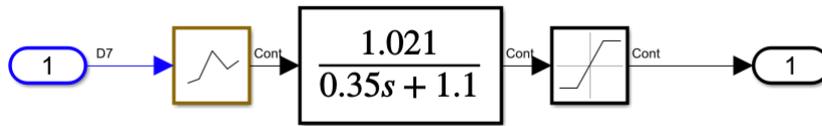


Figura 3.10 Filtro de suavizado

La secuencia de bloques que atraviesa la señal escalonada que entra en el filtro sigue este orden:

- **Mantenedor de primer orden:** Un mantenedor de primer orden es un dispositivo utilizado en sistemas de control muestreado. Este tipo de mantenedor se basa en obtener la derivada en la muestra exacta, calcular su pendiente y entre muestra y muestra crear una recta con la pendiente obtenida anteriormente.
- **Función de transferencia de primer orden:** La función de transferencia de primer orden utilizada ($1.021/(0.35s+1.1)$) suaviza los cambios de pendiente del mantenedor anterior y se adapta a los picos máximos de irradiancia.
- **Saturador:** El saturador está colocado enfocado sobre todo en el límite inferior ya que los dos bloques anteriores hacen que en la bajada exista una sobre oscilación por debajo de cero. Por ello, una vez que la señal llegue a 0, se satura ya que la irradiancia no puede ser negativa.

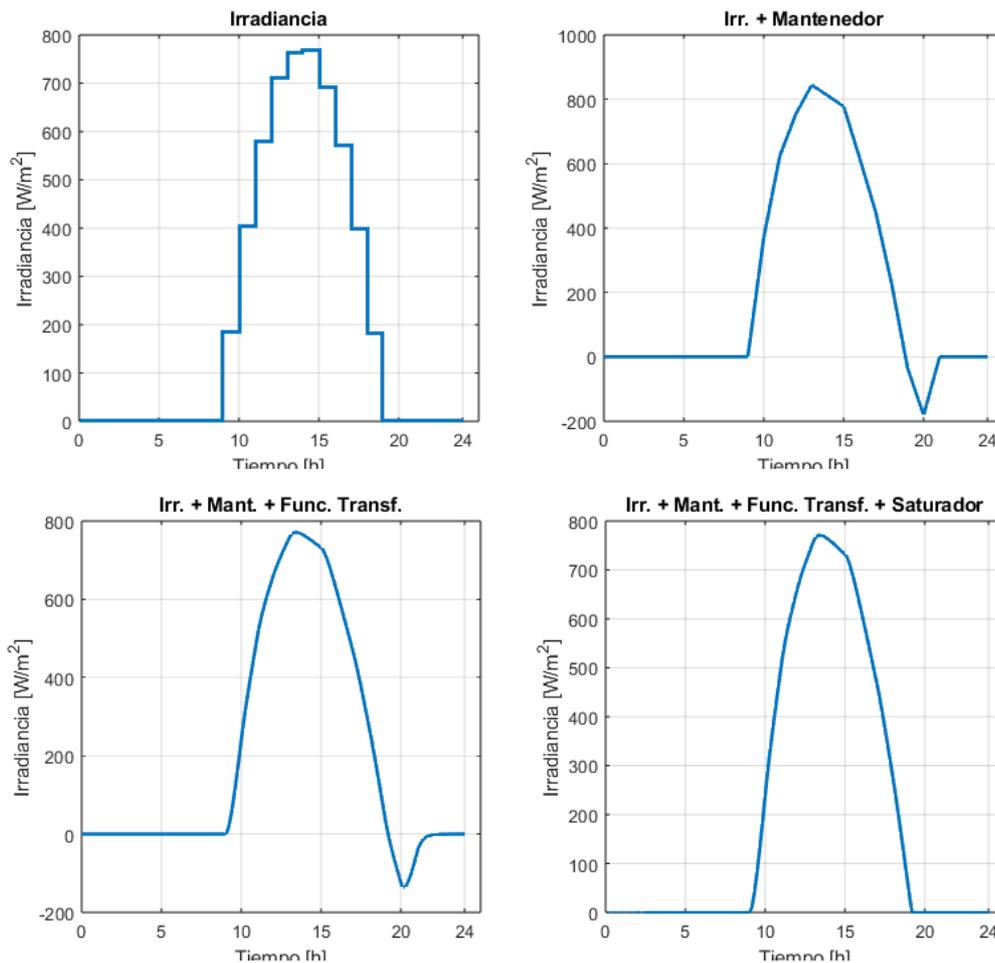


Figura 3.11 Evolución del filtrado

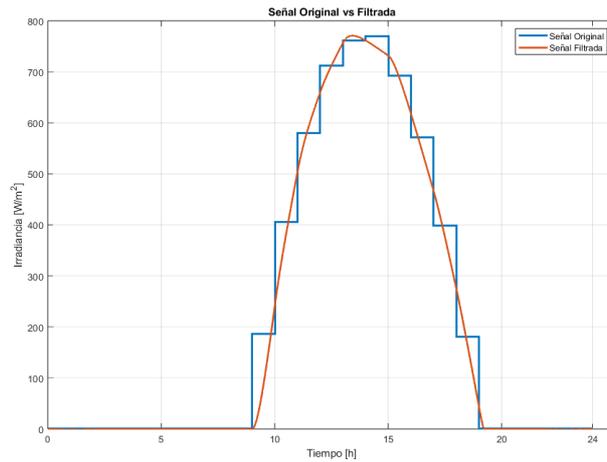


Figura 3.12 Comparativa entre la señal original y la señal filtrada

3.1.4 Temperatura

La otra entrada del bloque *PV Array* es la temperatura. Esta afecta de forma directa al rendimiento de la instalación. Las temperaturas de más de 30°C pueden llegar a reducir la eficiencia en un 10%. Una temperatura ideal donde la producción de energía es óptima se sitúa entre los 20 y 25°C [12].

Para medir la temperatura a la que se encuentran los paneles en la vida real, hemos medido de forma empírica los datos durante 24h con un termómetro y sensor de humedad Wifi, un dispositivo integrado dentro de la tecnología IoT. En concreto hemos usado un sensor TH08 conectado a la app *Smartlife* del desarrollador *Volcano Technology Limited*.



Figura 3.13 Sensor junto con app colocado bajo panel

El sensor se ha colocado bajo el panel para no recibir el calor directo del sol y medir la temperatura de las celdas lo más pegado posible a la superficie de estas.

Tabla 3.3 Evolución de la temperatura en Sevilla (24h)

Hora	Temperatura [°C]
00:00	12.9
01:00	11.4

02:00	10.3
03:00	10
04:00	9.4
05:00	8.8
06:00	7.9
07:00	7.2
08:00	8.8
09:00	10.9
10:00	16
11:00	19.4
12:00	21.2
13:00	24.6
14:00	25.7
15:00	25.8
16:00	25.4
17:00	23.3
18:00	20.1
19:00	16.7
20:00	14.5
21:00	12.3
22:00	11.7
23:00	11

La evolución de la temperatura se introduce de la misma forma que la irradiancia en el apartado anterior. Hemos usado un *Repeating Sequence Stair* seguido de un filtro para suavizar los escalones ya que la temperatura también es una magnitud que evoluciona de forma continua y suave. La modificación que recibe el filtro en esta ocasión es la eliminación del saturador ya que las temperaturas pueden ser negativas y la naturaleza de los escalones no hace que el control genere tanta sobre oscilación. Aun así, se ha modificado la función de transferencia por $1.1/(0.2s+1.13)$ para adaptar mejor la señal filtrada a los valores originales.

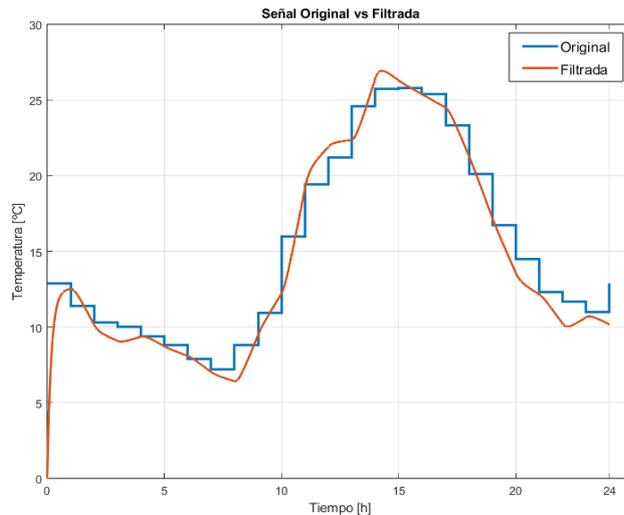


Figura 3.14 Comparativa entre la señal original y la señal filtrada

Recordamos que las señales filtradas son ilustrativas y suponen una mera aproximación para poder simular la microrred de una forma más realista. El estudio detallado de estos parámetros supone un área de estudio fuera de este proyecto.

3.1.5 Potencia generada

Una vez configurado el bloque *PV Array* con las especificaciones de nuestros paneles e introducidos los datos de irradiancia y temperatura propios de nuestro clima, se realiza el modelo simple de la instalación. Al trabajar con balances de potencia, debemos pasar la tensión en bornas del bloque solar a niveles de potencia. Para ello, hemos obtenido el producto de la corriente y la tensión producidas por la instalación. Estos voltímetros y amperímetros vienen incluidos con el toolbox de *Simscape Electrical* y sus resultados vienen dados por una pequeña resistencia de 50Ω conectada a las bornas de los paneles. Esta resistencia es teórica y no representa ningún estudio práctico.

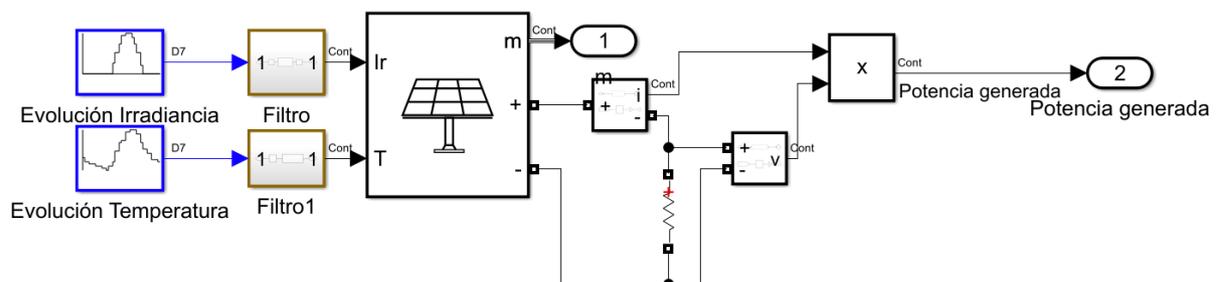


Figura 3.15 Modelo de la instalación fotovoltaica

Cuando simulamos el modelo, podemos observar que los niveles de irradiancia y temperatura no permiten que la instalación dé su máximo potencial.

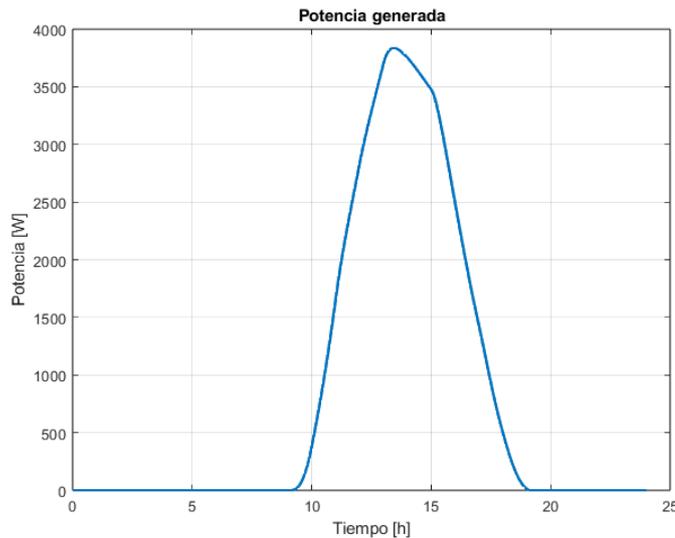


Figura 3.16 Potencia generada a lo largo de un día

3.2 Energía consumida

La cantidad de energía que consume un hogar está influenciada por una variedad de factores, tales como el número de dispositivos eléctricos que posee, el tamaño de la vivienda, la cantidad de personas que la habitan, e incluso variables ambientales como la estación del año y las condiciones meteorológicas. No sorprende, entonces, que cada hogar presente patrones de consumo energético únicos, moldeados por las distintas rutinas y estilos de vida de sus ocupantes.

Este proyecto surge precisamente de reconocer esa diversidad en los patrones de consumo. Observamos que los periodos de bajo consumo en un hogar pueden coincidir con picos de demanda en otro. De esta manera, el excedente energético de una vivienda podría ser aprovechado por otra, creando un sistema de intercambio que optimiza el uso de la energía a nivel comunitario. Esta idea subyacente no solo es la inspiración sino también el fundamento de nuestro estudio, buscando una forma más inteligente y sostenible de gestionar el consumo de energía en las comunidades.

3.2.1 Rutinas energéticas

Para llevar a cabo este estudio, hemos definido cuatro patrones energéticos distintos en hogares, diseñados para intercarse de manera que faciliten un intercambio de potencias durante las simulaciones, proporcionando una visión más clara del proceso. A continuación, describimos cada uno de estos escenarios:

- **Familia consumiendo todo el día:** En el hogar 1, la actividad comienza temprano, a las 6:15 a.m., con el uso intensivo de electrodomésticos como secadores de pelo y termos eléctricos, lo que genera picos de consumo matutinos. A lo largo del día, el consumo se estabiliza, pero aumenta significativamente hacia la hora de comer, especialmente con el uso de electrodomésticos mayores como hornos y vitrocerámicas. A las 6:30 p.m., se observa un breve aumento en la demanda, y por la noche, la preparación de la cena, junto con el uso de la televisión, calefacción e iluminación, eleva nuevamente el consumo. Es importante destacar que, durante todo el día, existe una carga constante debida al consumo "fantasma" de dispositivos que permanecen conectados, como el frigorífico, los cargadores de móviles sin estar conectados a ellos o los electrodomésticos en modo espera.
- **Familia no cena en casa:** La familia del hogar 2 sigue un patrón energético similar al del hogar 1 durante el día. Sin embargo, por la noche, este hogar solo registra el consumo "fantasma", ya que sus miembros suelen cenar fuera.
- **Familia está de vacaciones:** En el hogar 3, la familia está ausente por vacaciones. En este periodo, la única energía consumida es la carga "fantasma". Este escenario ofrece una oportunidad para observar

cómo sería la evolución del estado de carga de las baterías en un día sin fluctuaciones significativas de consumo.

- **Familia llega a casa de noche:** La familia del hogar 4 lleva un ritmo de vida que los mantiene fuera de casa durante el día. Solo consumen energía en la rutina matutina y luego por la noche, al regresar para cenar y disfrutar de actividades de ocio antes de dormir.

Antes de explicar cómo se ha recreado el consumo de cada vivienda, debemos explicar en qué nos hemos basado para establecer estas rutinas. Hemos aprovechado la app de monitorización de nuestra instalación doméstica real para imitar el consumo de estas casas ficticias. La app en particular es *FusionSolar* de la china *Huawei*.



Figura 3.17 Monitorización de la potencia mediante app

Puede observarse una rutina muy parecida a las propuestas para simulación.

3.2.2 Recreación del consumo

Para recrear la demanda de energía de un hogar, nos hemos inspirado en las formas de la figura 3.15 del apartado anterior. Lo primero en lo que nos hemos fijado ha sido que la señal tiene un offset con ruido fruto del consumo fantasma por lo que hemos partido de un bloque *Random Number* para después sumarle los valles de consumo. A partir de aquí, realizamos una combinación de bloques tales como:

- **Sine Wave:** Este bloque genera una señal sinusoidal cuya amplitud corresponde al pico de consumo que deseamos representar, emulando las variaciones típicas en la demanda energética a lo largo del día.
- **Pulse generator:** El generador de pulsos nos sirve para habilitar aquellas señales en el espacio de tiempo deseado y anularlas cuando no las necesitamos. Al combinar un pulso de amplitud unitaria y período específico con una señal sinusoidal, por ejemplo, la señal resultante se activa solo durante la duración del pulso. Este método se repite y ajusta a lo largo de las 24 horas de la simulación para simular los picos de consumo que ocurren en momentos específicos del día.
- **Random Number:** Este bloque nos añade un ruido característico de la instalación que gobierna todo el consumo diario. También nos puede servir para producir consumos escalonados modulando la varianza y el tiempo de muestreo.
- **Multiplexor:** Este bloque sirve para combinar todas las señales en un solo flujo de datos, asegurando que todas las modificaciones y adiciones se manejen dentro del mismo canal.
- **Max:** Este bloque actúa como un filtro seleccionando únicamente los valores máximos de la combinación de señales, generando así un perfil de consumo continuo y único.

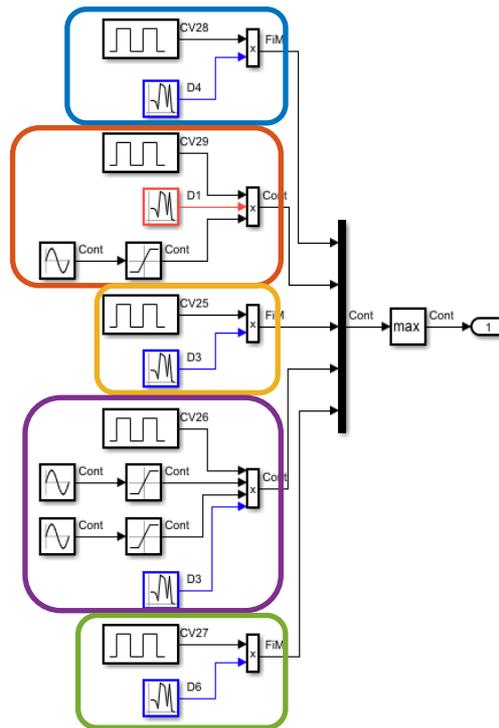


Figura 3.18 Combinación de señales

En la figura anterior podemos observar una recreación de una rutina energética a base de combinar los bloques mencionados. Los cuadros de color que rodean cada rama corresponden con el color de la señal de la figura siguiente donde se muestran los resultados graficados.

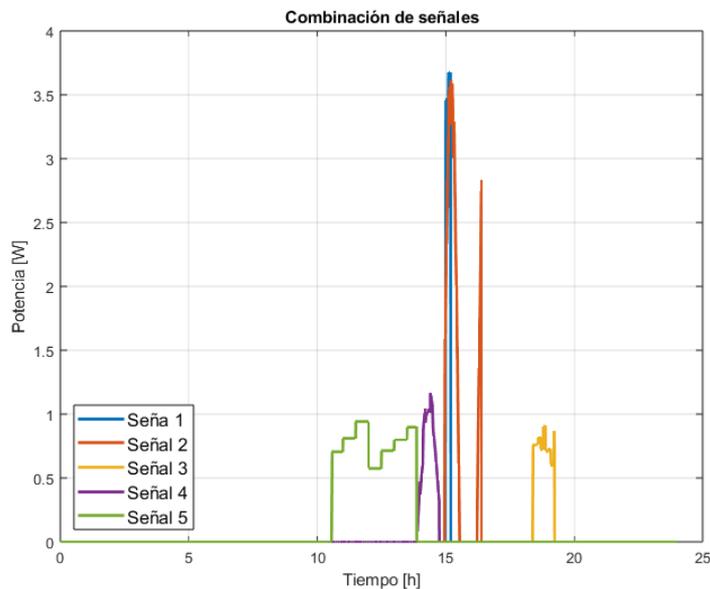


Figura 3.19 Potencias combinadas en una gráfica

Como mencionamos anteriormente, el bloque *max* filtra los valores máximos de toda la gráfica y se queda con el perfil superior aunando todas las curvas en una sola.

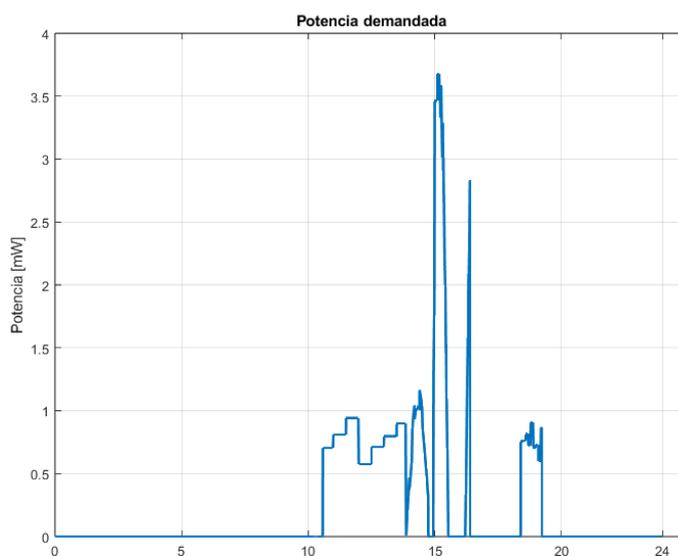


Figura 3.20 Perfil construido por max

Una vez establecida la metodología para generar picos de consumo y combinarlos adecuadamente, procedemos a crear las cuatro rutinas energéticas delineadas inicialmente. Para facilitar la identificación y posible modificación de los componentes del modelo, hemos segmentado las rutinas en periodos de mañana y tarde. Esta segmentación se implementa utilizando subsistemas, que agrupan y organizan bloques múltiples en configuraciones más compactas y manejables.

Cada subsistema se enriquece con un bloque de *Random Number*, el cual simula la carga fantasma constante en el sistema. Es importante mencionar que, dado que inicialmente trabajamos con números en mili vatios (mW), efectuamos una conversión final multiplicando los valores por 1000, lo que nos permite expresar el consumo en vatios (W). Esta conversión asegura que los valores sean coherentes y comparables con estándares de medición realistas, facilitando así la interpretación y análisis de los resultados. Esta estructuración no solo mejora la claridad del modelo, sino que también optimiza el proceso de simulación, permitiendo ajustes precisos y específicos según las necesidades de cualquier escenario energético simulado.

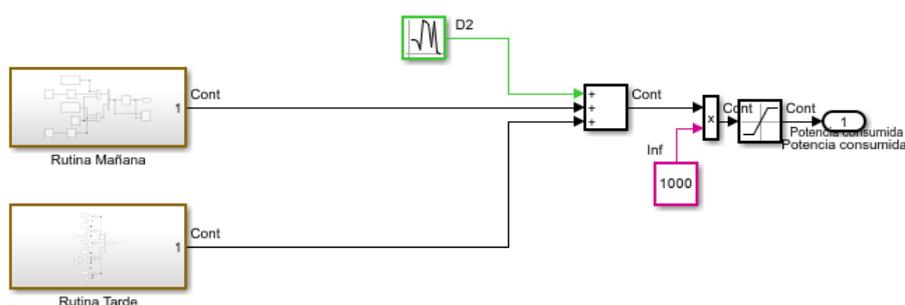


Figura 3.21 Modelo de Simulink de la demanda de un hogar

Finalmente, la recreación del consumo de cada uno de los cuatro hogares realizando la combinación de bloques queda de esta forma:

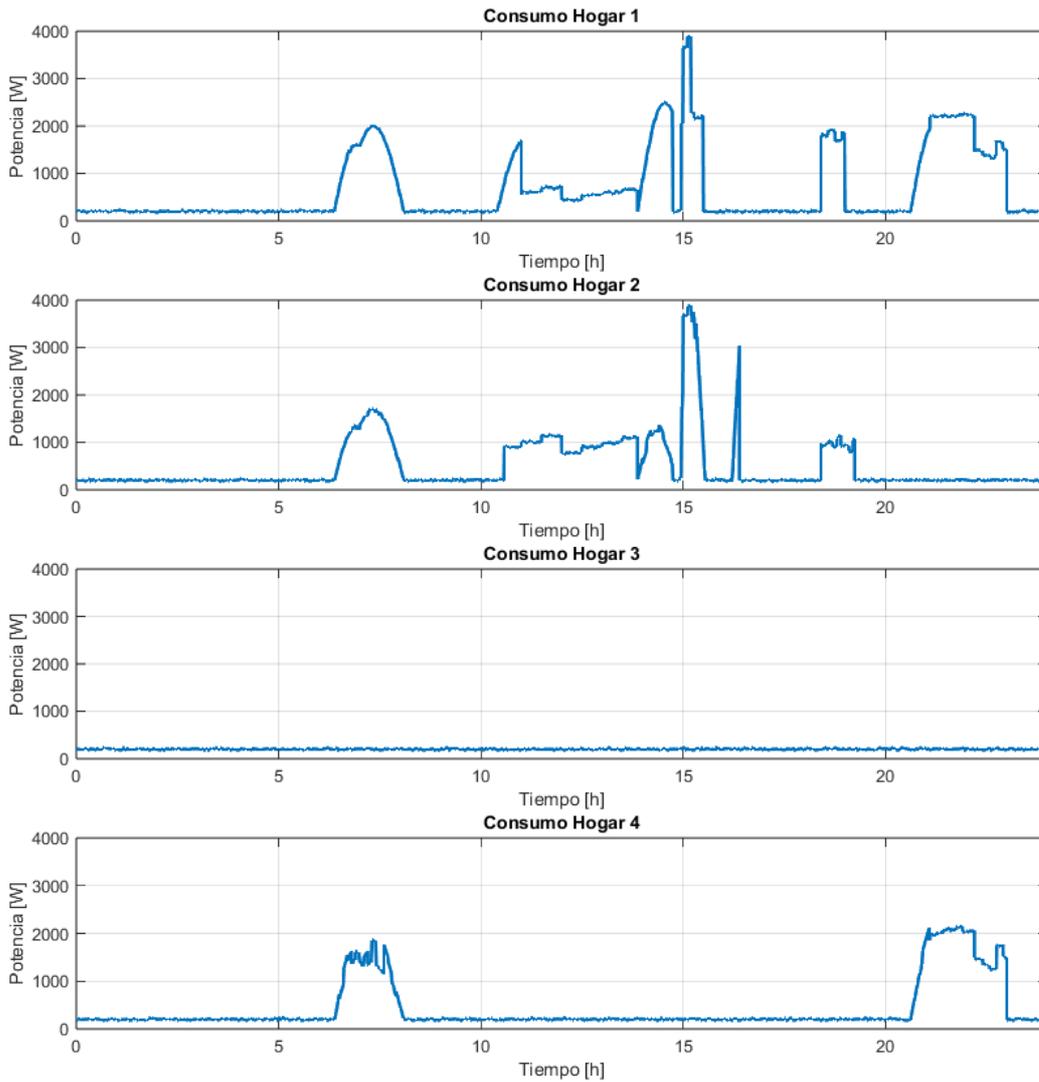


Figura 3.22 Consumos de cada hogar

3.2.3 Potencias netas

Como vimos en el subapartado 2.2.1, la potencia neta es la diferencia entre la potencia consumida y la generada de tal forma que, si esta diferencia es negativa, habrá excedentes y si es positiva, se consumirá más de lo que se genera.

Para calcular la potencia neta en el modelo, basta con restar los bloques de los apartados 3.1 y 3.2 de este capítulo.

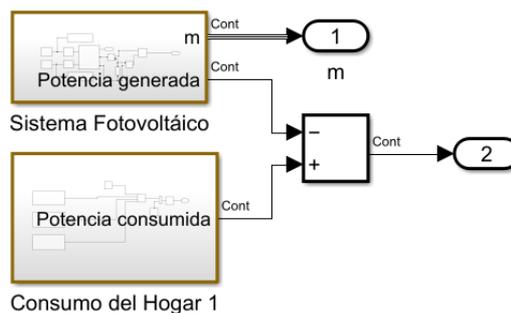


Figura 3.23 Modelado de Potencia Neta

En la figura anterior, el subsistema *Sistema Fotovoltaico* es la instalación desarrollada en el apartado 3.1 y es el mismo subsistema para las cuatro viviendas ya que la instalación posee las mismas características y el sol incide de la misma forma en cada una debido a que todas forman parte del mismo vecindario.

El subsistema *Consumo del hogar X* es el creado en el apartado 3.2 y produce la combinación de todos los bloques utilizados para imitar el consumo de un hogar según las rutinas antes explicadas.

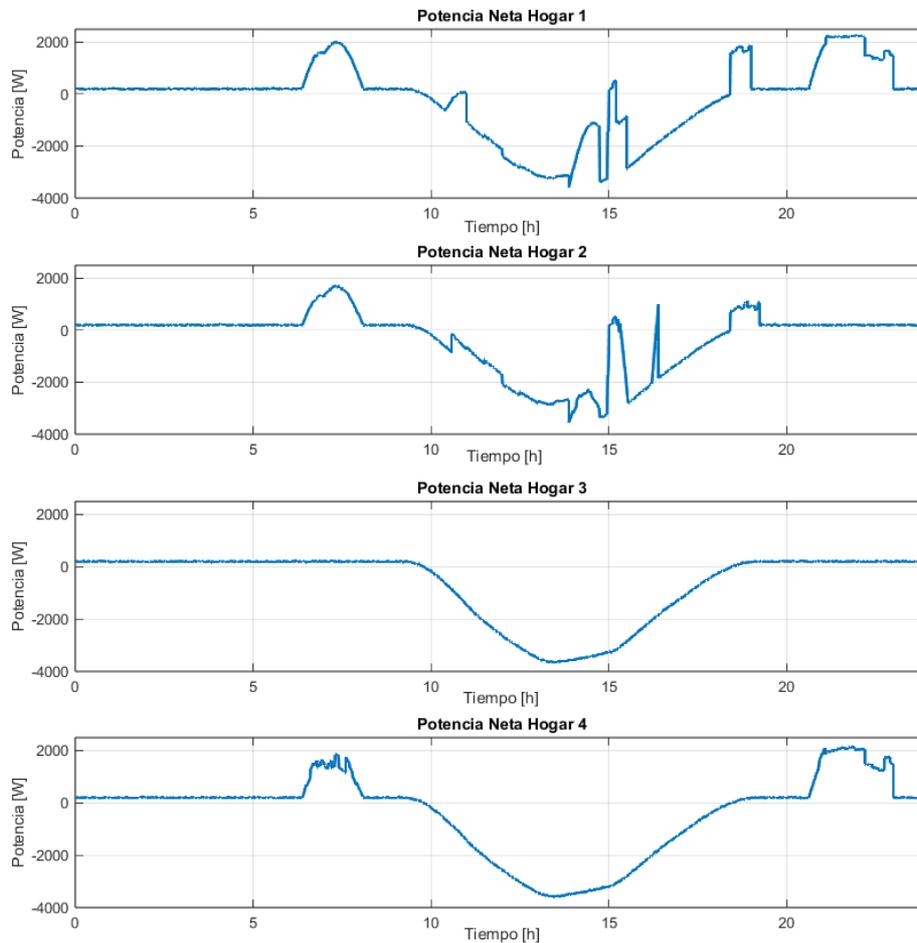


Figura 3.24 Potencias netas

3.3 Batería

Si recordamos la batería del modelo inicial descrito en el apartado 2.2, implementábamos un modelo matemático que relacionaba la potencia neta con el estado de carga de la batería. Ahora, avanzando en la precisión y realismo de nuestra simulación, *Simscape Electrical* nos ofrece la posibilidad de simular un bloque de batería que incorpora especificaciones técnicas avanzadas, replicando con mayor fidelidad el comportamiento real de una batería.

Al emplear *Simscape Electrical* para simular el comportamiento de la batería, podemos realizar análisis más complejos y obtener predicciones más precisas sobre el rendimiento de los sistemas de almacenamiento de energía en diferentes escenarios de uso. Esto es fundamental para la optimización de sistemas de gestión de la energía en hogares, donde la eficiencia y la fiabilidad son cruciales para el éxito del modelo de intercambio de energía propuesto.

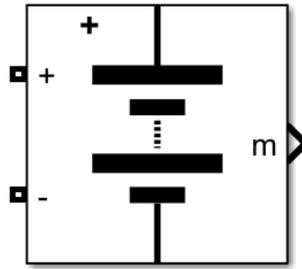


Figura 3.25 Bloque de batería

En el marco de este proyecto, hemos optado por utilizar baterías con especificaciones no convencionales, dado que nuestro interés principal es examinar los intercambios de potencia y, consecuentemente, la evolución de los estados de carga en condiciones extremas o críticas. Esta elección nos permite observar con claridad las dinámicas de carga y descarga en escenarios que ponen a prueba los límites del sistema.

Por otro lado, en un modelo convencional, las especificaciones estándar permiten que los estados de carga varíen de acuerdo al tiempo, medido en segundos. Sin embargo, en nuestro contexto, donde la simulación abarca un período de 24 segundos, dichas variaciones resultarían insignificantes para una batería destinada a este tipo de uso, ya que la evolución del estado de carga en un lapso tan corto sería prácticamente imperceptible.

Para adaptar este comportamiento a las necesidades de nuestro estudio, hemos realizado ajustes en las especificaciones técnicas de las baterías. Esta conversión consiste en multiplicar la capacidad instalada, expresada en amperios hora [A·h], por 3600, y dividir el tiempo de respuesta de la batería, que originalmente es de 30 segundos, también por 3600. De esta manera, aseguramos que la evolución del estado de carga durante un segundo de nuestra simulación refleje lo que normalmente ocurriría durante una hora real.

Tabla 3.4 Especificaciones de la batería

Parámetro	Valor
Tipo	Ion-Litio
Tensión nominal [V]	360
Corriente instalada [Ah]	0.027
Estado de carga inicial [%]	45
Tiempo de respuesta de la batería [s]	0.00833

Hemos ignorado los efectos de la temperatura y el envejecimiento.

Para integrar adecuadamente la batería en el sistema de balance de potencias, tanto de la instalación propia como de las adyacentes, ha sido esencial convertir la medida de potencia a corriente. Este proceso se ha logrado mediante la implementación de un bloque de función en el cual se divide la potencia por la tensión nominal de la batería. Este cálculo produce una corriente que luego utilizamos como referencia para una fuente de corriente. La corriente resultante de esta fuente es la que alimenta directamente a la batería, permitiendo así una integración efectiva y eficiente de la batería en el sistema general de gestión de energía. Esta conversión es crucial para asegurar que la batería pueda interactuar de manera precisa con las variaciones de potencia generadas por el consumo propio y el de los vecinos.

Por último, hemos inicializado los estados de carga de las baterías 1, 2, 3 y 4 a 45%, 70%, 35% y 60% respectivamente.

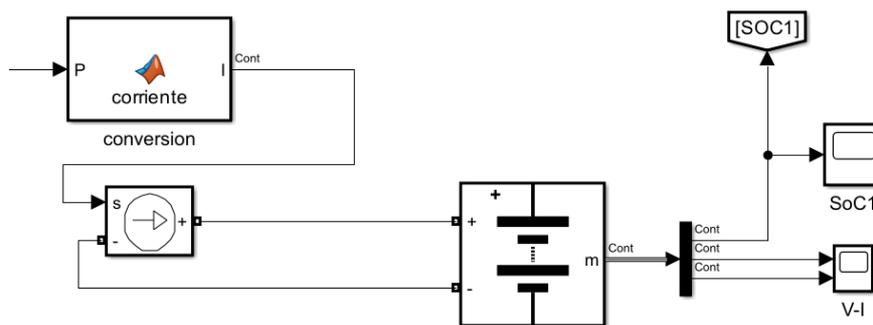


Figura 3.26 Implementación completa de la batería

3.4 Resultados del modelo aproximado a la realidad

Después de desarrollar las potencias netas y las baterías para la versión más realista de la microrred, solo tenemos que sustituir las etapas del modelo inicial por las nuevas. Además, añadimos el bloque *powergui* que se encarga de manejar la simulación cuando hay elementos de *Simscape* presentes.

El modelo con viviendas independientes (sin microrred ni control) quedaría de esta forma:

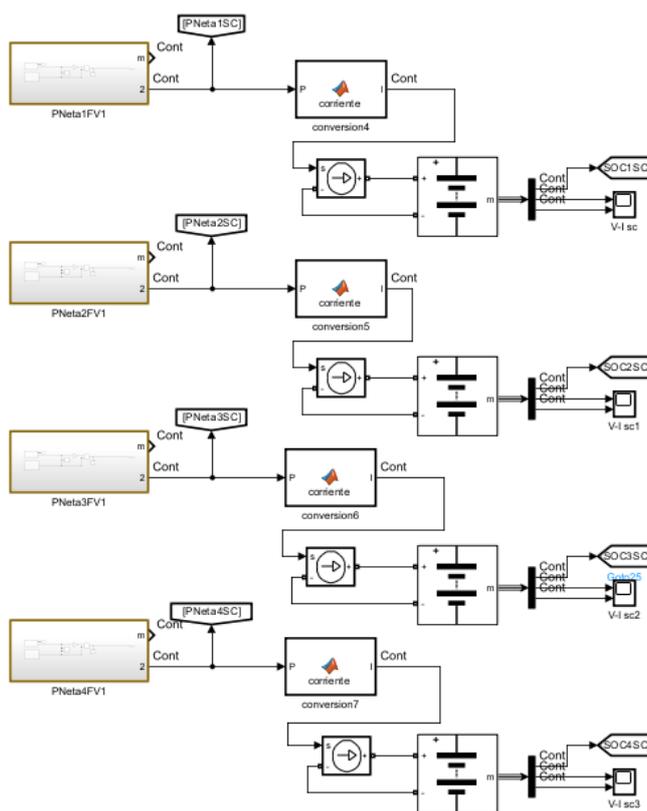


Figura 3.27 Modelo realista sin control

Tras muchas simulaciones, dimos con una configuración en la que el *solver* de Matlab pudiese mostrar resultados claros sin necesidad de que la duración de la simulación durase 10 minutos. Simplemente establecimos el tamaño máximo de paso en 0.001 con una selección automática del *solver* y el simulador en modo *Acceleration*.

Los resultados de la simulación sin controlador son:

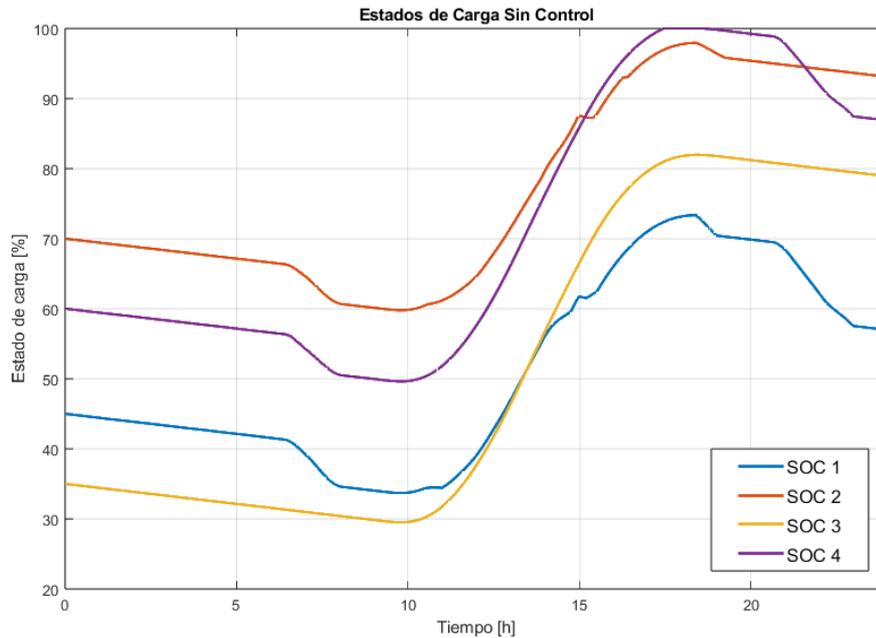


Figura 3.28 Evolución de los SOC sin control

Inicialmente, se aprecia una disminución uniforme del estado de carga (SOC) en todas las viviendas, atribuible al consumo fantasma nocturno mientras los residentes duermen. Este descenso representa una tasa constante de 0.57% por hora. Con el amanecer y el inicio de la rutina matutina, se observa una reducción adicional del SOC debido al uso intensivo de aparatos como termos, cafeteras, secadores y luces. Sin embargo, esta disminución se estabiliza y comienza a revertirse con la generación de energía solar, gracias a los excedentes energéticos presentes durante las horas de luz diurna, salvo en momentos puntuales donde el consumo supera la generación.

Al anochecer, con el inicio de las actividades nocturnas como la cena y el tiempo de ocio, se evidencia una caída notable en los SOC, marcando el fin de la jornada energética que se repetirá al siguiente día. Es relevante destacar el caso del SOC 3, perteneciente a una vivienda desocupada, donde la carga y descarga de la batería ocurren sin otras interferencias más allá del consumo fantasma, proporcionando un claro ejemplo de la dinámica de la batería bajo condiciones controladas.

Durante la simulación, se ha cuidado especialmente evitar la saturación de las baterías en sus límites de carga superior e inferior. Mantener el SOC dentro de un rango óptimo es crucial, ya que extremos en el nivel de carga pueden acelerar significativamente la degradación de las baterías, afectando su eficacia y vida útil.

Finalmente, sólo nos queda conectar los bloques de control y crear las conexiones entre vecinos, tal y como hicimos en el modelo inicial completo.

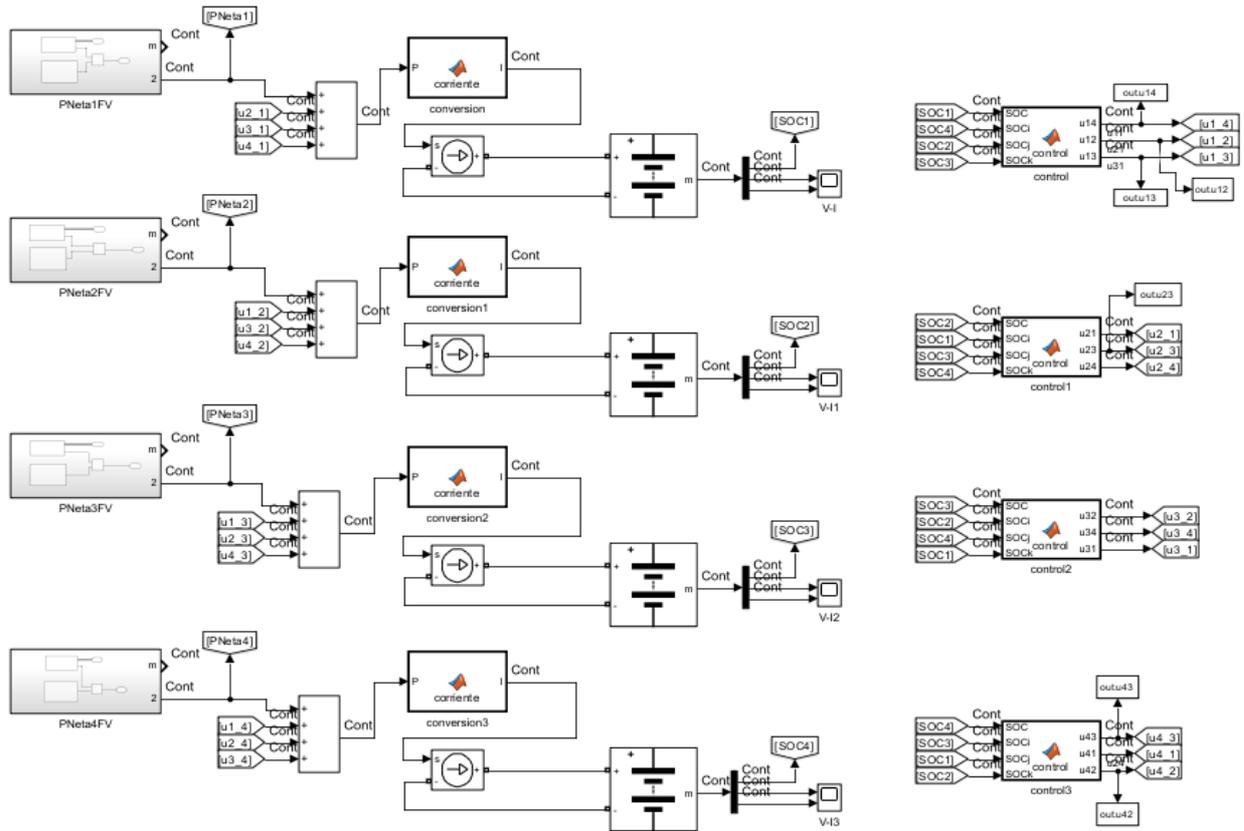


Figura 3.29 Modelo de microrred completo con control distribuido

Este es el modelo completo de la microrred. Las señales de interés son los estados de carga de las cuatro viviendas y los flujos de potencia entre vecinos. Para simplificar la visualización y análisis de los datos, se han seleccionado únicamente la mitad de los flujos de potencia, dado que la otra mitad refleja los mismos valores en sentido inverso; es decir, si u1_2 representa el flujo de potencia del hogar 1 al hogar 2, entonces u2_1 indicará el mismo valor, pero en dirección contraria y con signo negativo. Esta estructura permite una comprensión clara y eficiente de las interacciones energéticas dentro de la microrred, facilitando el seguimiento y la gestión de los recursos compartidos entre las viviendas conectadas.

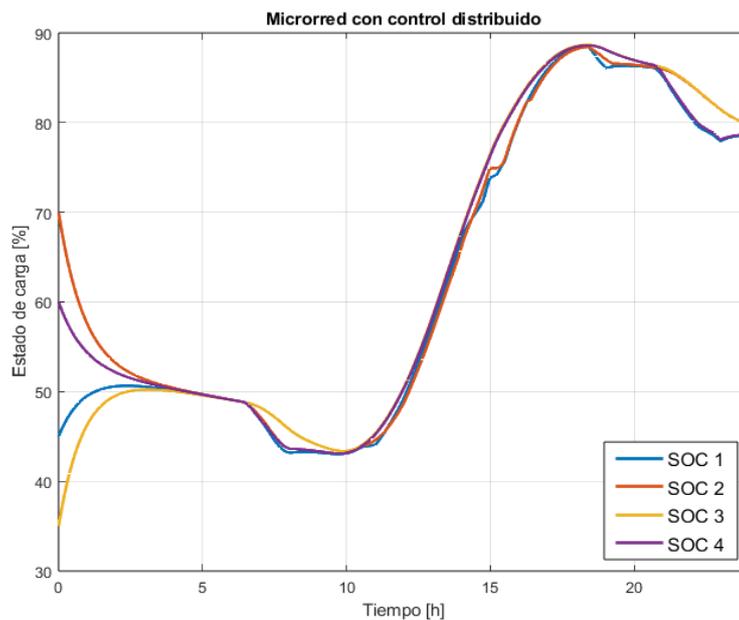


Figura 3.30 SOC con control distribuido

A medida que avanza la noche, se observa una tendencia hacia la estabilización y consenso de los estados de carga entre las viviendas, un fenómeno facilitado por establecer el coeficiente de control K_C en 100, permitiendo así capturar con claridad los transitorios. Específicamente, los hogares 2 y 4 transfieren parte de su carga a los hogares 1 y 3, evidenciando una distribución dinámica y eficiente de la energía almacenada.

Al comparar estos resultados con los presentados en la figura 3.28, que ilustra el sistema operando sin control, se destacan varios puntos clave que subrayan la eficacia de la implementación de la microrred. En particular, en torno a las 10 a.m., los hogares 1 y 3 exhiben un aumento del 5 al 10% en su capacidad de carga disponible. Gracias al control distribuido, los SOC's de estos hogares no tienen una caída abrupta. Además, al finalizar la simulación, todos los hogares concluyen el día con alrededor del 80% de carga en sus baterías, lo que supone una mejora significativa respecto al modelo sin control, donde el hogar 1 terminaba el día con tan solo un 58% de carga. Este aumento representa aproximadamente un 22% adicional de capacidad disponible para el día siguiente, optimizando así el uso y la gestión de la energía en la microrred.

Los intercambios de potencia realizados a lo largo de la simulación son los siguientes:

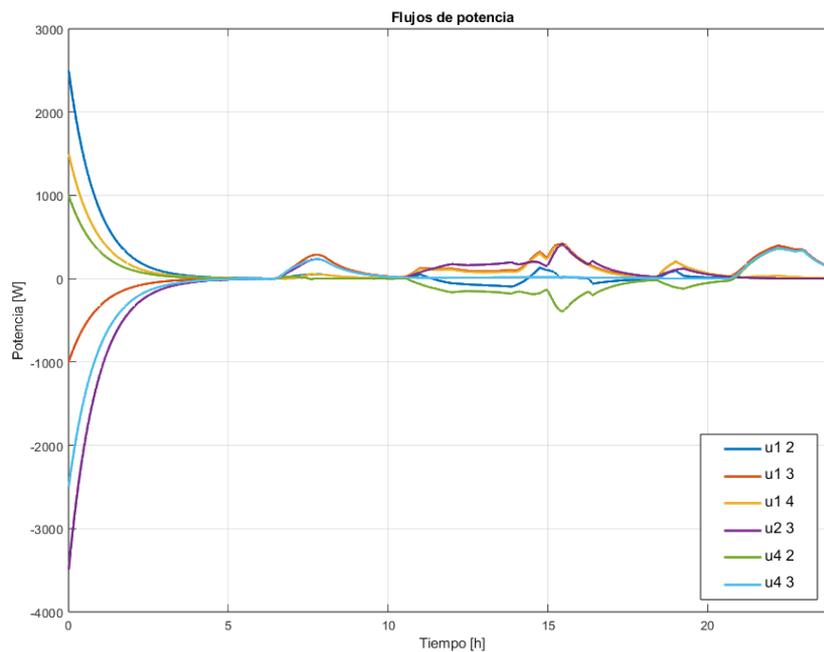


Figura 3.31 SOC's con control distribuido

3.5 Desconexión de una microrred de la red

Las microrredes son sistemas no exentos de fallos, degradaciones y averías. Por ello, estas tienen un punto de enlace con la red eléctrica del exterior en caso de reportar fallos temporales. El algoritmo de control es robusto ante desconexión de microrred cuando uno o más usuarios se dan de baja o de alta. Estas desconexiones temporales se pueden simular en nuestro modelo mediante el ya conocido *Repeating Sequence Stair*.

En este caso, vamos a desconectar el hogar 1 desde las 8 a.m. hasta las 6 p.m.

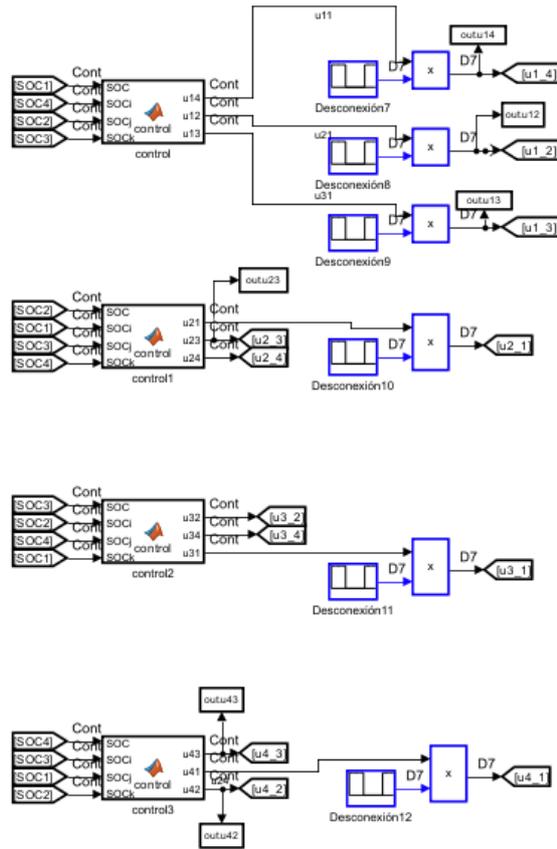


Figura 3.32 Implementación de la desconexión

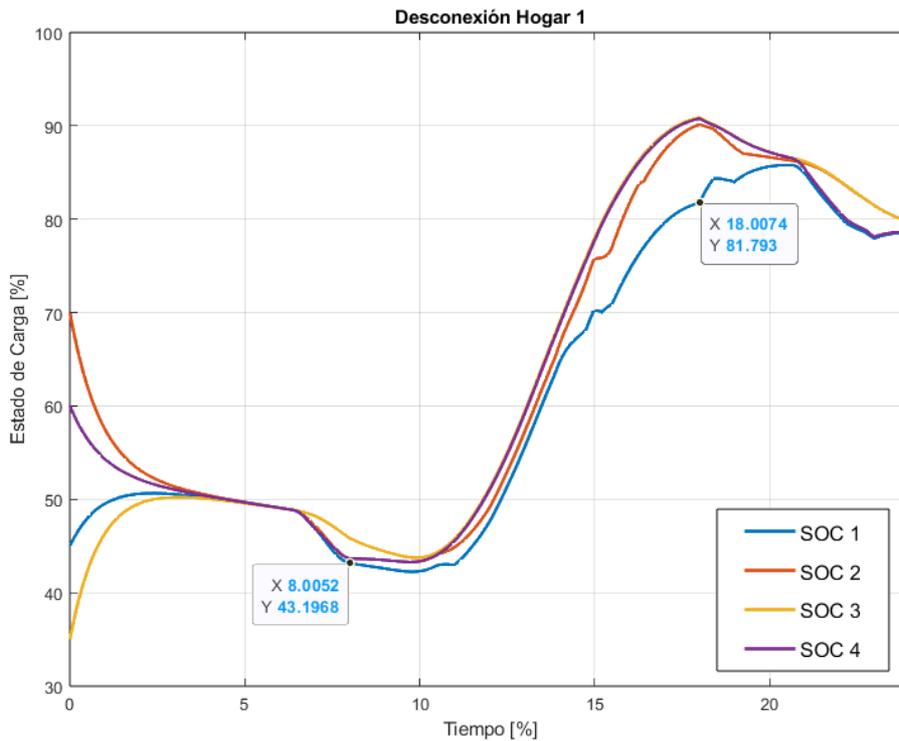


Figura 3.33 Desconexión del Hogar 1

Los marcadores de la figura 3.29 muestran el tramo donde el hogar 1 está desconectado. Se puede observar que el estado de carga de la batería de esta vivienda evoluciona de forma independiente mientras que los estados de carga restantes se auto regulan entre sí. Cuando acaba el tramo de desconexión, la ley de control hace que el hogar 1 se integre con el resto de *SOCs* continuando con el control típico.

Aquí se observa la anulación de los flujos conectados al hogar 1 durante este tramo:

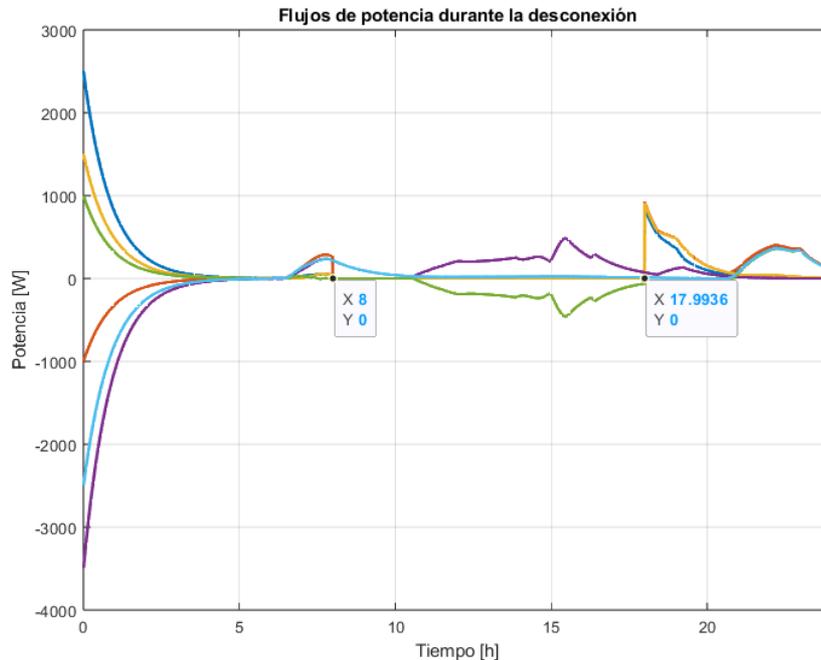


Figura 3.34 Flujos de potencia durante la desconexión

Con esto finaliza este capítulo donde hemos modelado una microrred desde cero. En conjunto, los resultados obtenidos muestran una eficacia al promover una distribución equitativa de la energía y en mantener altos niveles de carga en las baterías de las viviendas involucradas, asegurando así que el sistema pueda soportar las demandas energéticas diarias de manera eficiente. Estos resultados son prometedores para la aplicación futura de microrredes en entornos urbanos y rurales, donde pueden desempeñar un papel crucial en la transición hacia sistemas energéticos más sostenibles y menos dependientes de las redes eléctricas tradicionales.

4 APLICACIÓN DE MONITOREO

En este capítulo vamos a reflejar los resultados de la simulación mediante su reproducción en un software propio. En él se pretende mostrar de una forma visual los consumos de energía, los flujos de potencia entre hogares, la conexión y desconexión de la red.

4.1 Motivación

El uso de aplicaciones de monitorización ha experimentado un crecimiento significativo a lo largo de los años, impulsado por los avances en tecnologías inalámbricas. Este progreso ha facilitado el desarrollo de dispositivos del *Internet de las Cosas (IoT)*, que se integran en ecosistemas de dispositivos interconectados. La domótica y la automatización residencial, en constante auge, están aprovechando estas tecnologías para crear hogares más inteligentes y conectados.

Las empresas instaladoras están incorporando las tecnologías fotovoltaicas a estos sistemas domóticos, permitiendo a los usuarios controlar y adaptar su consumo energético en función de las condiciones disponibles. Esta integración se realiza mediante la implementación de antenas Wifi en los inversores de potencia que forman parte de los kits solares actuales. Dichas antenas transmiten datos en tiempo real a aplicaciones móviles o de escritorio asociadas con el inversor, permitiendo una gestión detallada y accesible del consumo energético. Además, los principales fabricantes mantienen bases de datos que almacenan históricos de consumo, proporcionando a los clientes acceso continuo a sus datos energéticos.

Para maximizar la utilidad de estas aplicaciones, es fundamental que presenten la información de manera clara y concisa, asegurando que sean intuitivas y fáciles de usar para cualquier usuario. Se invierte considerablemente en el diseño de interfaces usuario-amigables, con un enfoque en la estética que transmita la sensación de un producto robusto y exclusivo. Empresas como *Huawei* lideran el mercado con sistemas inversores que incluyen sus propias aplicaciones y servidores, destacando en este sector en rápido desarrollo.

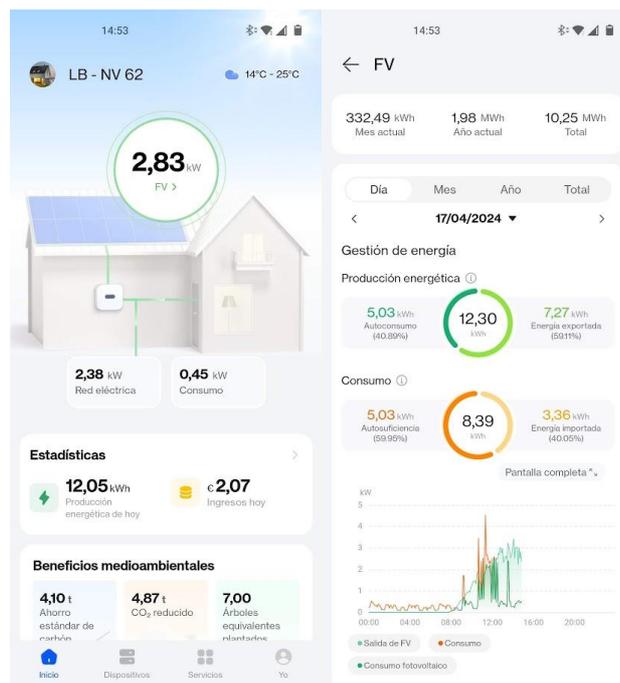


Figura 4.1 Aplicación FusionSolar de Huawei

La figura anterior sirve como inspiración a la hora de desarrollar nuestro propio software. Vemos los datos de generación fotovoltaica acompañados de los flujos de energía que derivan en el tendido eléctrico o en el consumo del hogar. Si abrimos el bloque de sobre la potencia generada, podremos observar un desglose de la gestión de energía del día de hoy.

4.2 Software utilizado

Para crear la aplicación, no hemos querido abandonar el entorno de simulación de Matlab ya que el programa guarda los vectores de miles de datos listos para ser tratados. Por ello, vimos conveniente por comodidad y, a modo de reto, aprovechar la herramienta de Matlab: *App Designer*.

App Designer es una herramienta que permite crear apps profesionales de una manera visual con nociones básicas en programación. Sustituyó recientemente a la herramienta GUIDE que tenía un uso parecido. Este entorno integra las dos tareas principales en la creación de una app: la distribución de los componentes visuales de una interfaz de usuario (GUI) y la programación del comportamiento de la app.

Para crear la interfaz de usuario, disponemos de un cajón con componentes visuales los cuales podemos arrastrar al área de diseño y configurarlos como deseamos. Podemos encontrar *displays*, interruptores, luces, botones, etc.

En la vista “Code View” podemos editar partes del código asociadas a los componentes que hemos arrastrado previamente. No nos permitirá editar líneas esenciales por lo que no correremos peligro de desestructurar el programa y plagarlo de errores de compilación.

La unión entre la parte gráfica y la codificación se hace mediante “Callbacks” que son funciones capaces de ejecutar ordenes cada vez que el ratón o el teclado interactúen con el elemento en cuestión.

A lo largo del capítulo veremos las dificultades y limitaciones que hemos experimentado con este entorno, pero también veremos qué ventajas tiene en cuanto a facilidad de programación y tratamiento de datos.

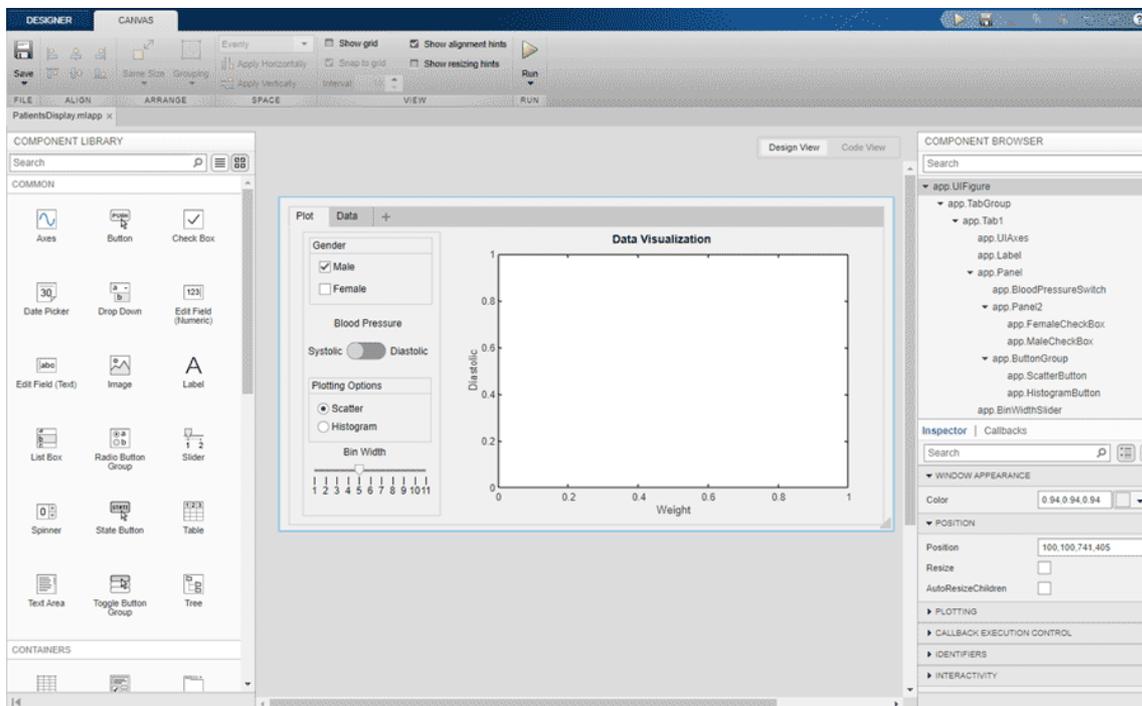


Figura 4.2 Entorno de App Designer

4.3 Interfaz de usuario gráfica (GUI)

Después de varios bocetos, finalizamos el diseño de la interfaz de usuario. La aplicación consta principalmente de cuatro bloques claramente definidos que representan datos de cada hogar, incluyendo el estado de carga y la conexión con la microrred. Estas secciones están interconectadas mediante flechas que señalan la dirección del flujo de potencia. A cada vivienda se le corresponde un color de tal forma que las flechas de los flujos se toman del color de la vivienda a la que se dirige. Esta regla hace que la transferencia de potencias sea mucho más ilustrativa.

En el lateral izquierdo de la interfaz, se ha integrado un módulo con dos pestañas que exhiben, respectivamente, el consumo de potencia neta de cada vivienda y los valores de potencia intercambiados en cada flujo. Además, en la esquina superior izquierda, se ha dispuesto un panel informativo que proporciona detalles sobre la ubicación de la urbanización, su nombre y la temperatura actual.

Para completar la interfaz, hemos incluido un botón de reproducción junto a un panel que indica la hora actual. Dado que se trata de una simulación de un programa, no es posible visualizar los datos en tiempo real conforme se actualizan en *Simulink*. Es necesario esperar a que finalice la simulación antes de poder reproducir los datos en la aplicación. Además, la duración de la reproducción se ha ajustado para no ser en tiempo real; el ciclo energético completo de un día se simula en aproximadamente 3 minutos, facilitando la demostración y comprensión de la dinámica del sistema sin extenderse durante 24 horas.

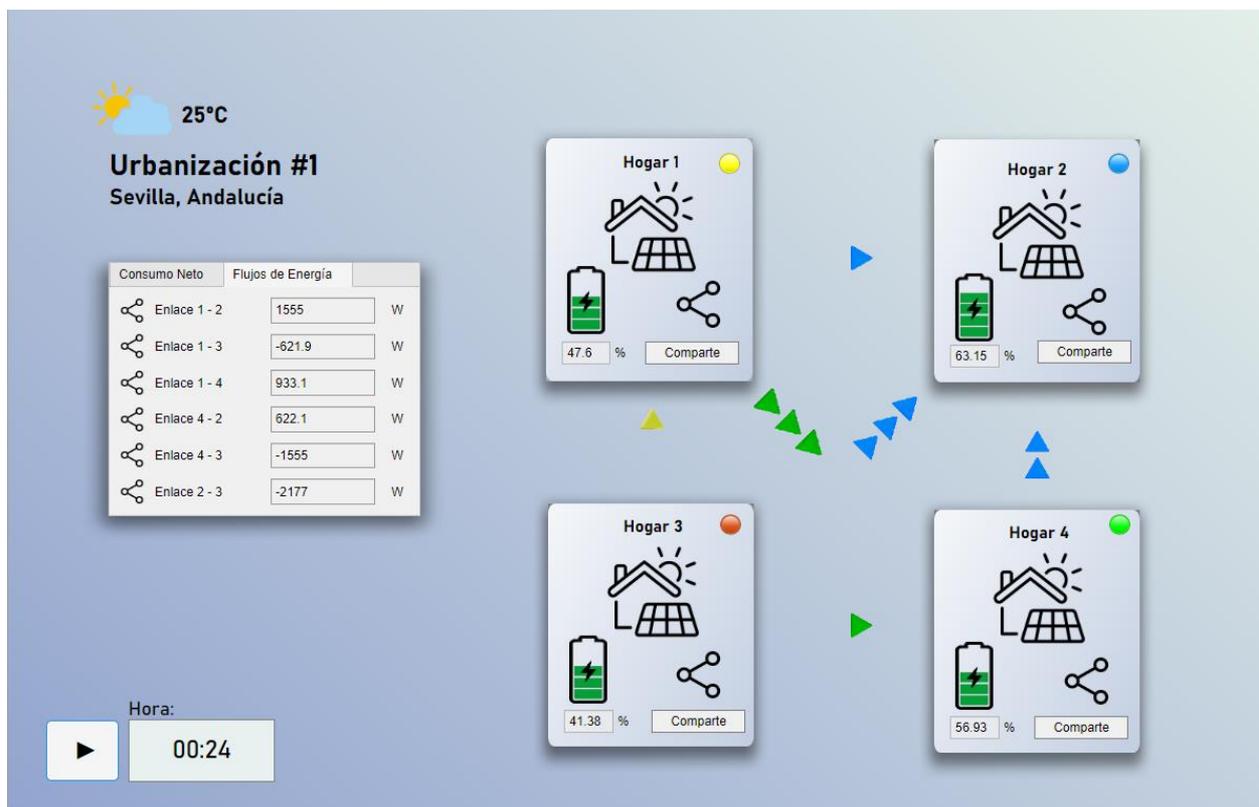


Figura 4.3 Interfaz gráfica de la aplicación

4.3.1 Estado de vivienda

Para dar una imagen de integración de todos los iconos, se han usado imágenes de formato *PNG* en toda la interfaz. Esto lo conseguimos gracias a la herramienta *Image* del listado de componentes.

Gobernando el cajón de estado nos encontramos con un icono de una casa con un panel solar indicando la naturaleza de la edificación. Sobre esta se encuentra un componente *Label* que informa sobre el número de hogar. A su derecha hemos colocado un componente *Lamp* que identifica al hogar con un color determinado.

Las esquinas inferiores contienen los datos de interés. A la derecha tenemos un icono de compartir que se

tachará si el hogar se desconecta de la microrred o si no es necesario compartir. Bajo el icono tendremos un *Edit Field (Text)* que imprime *Comparte* o *No comparte* dependiendo del caso.

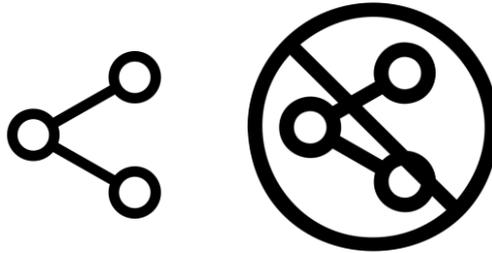


Figura 4.4 Iconos de compartir

A la izquierda tenemos una ilustración de una pila que añade o quita partes de relleno dependiendo del estado de carga actual reflejándose numéricamente en un *Edit Field (Numeric)*. En este componente se muestra el estado de carga en tanto por ciento.

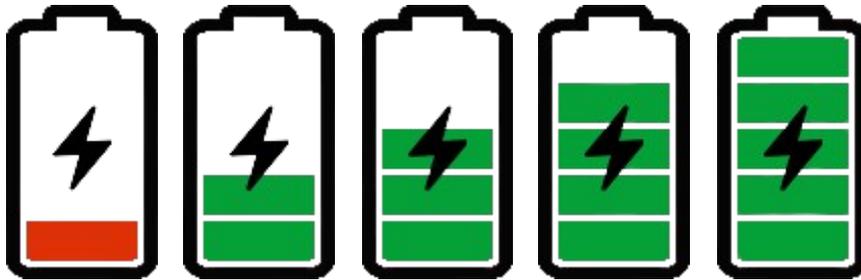


Figura 4.5 Iconos de batería

4.3.2 Flechas de flujos

Las flechas que indican la dirección del flujo son una imagen animada *GIF* y se han colocado también usando el bloque *Image*, pero, las limitaciones del programa no nos permitían rotar un *Image* así que hemos tenido que realizar varias flechas rotadas con distintos ángulos.

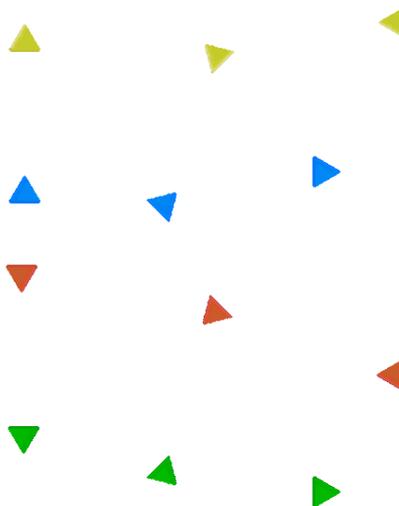


Figura 4.6 Flechas de flujos

4.3.3 Panel de información

El panel de información es un componente *Tab Group* que sirve para agrupar distintos paneles en pestañas. En este caso hemos creado dos pestañas: El consumo neto y los flujos de energía. Ambos paneles están estructurados con un componente *Grid* que se encarga de ajustar componentes en el panel como si de una tabla se tratase. Así los componentes estarán más equiespaciados.

En el panel de consumo neto, vamos a tener cuatro columnas que muestran la potencia neta de cada hogar. Cada fila del *Grid* tiene un icono de un rayo o de una hoja dependiendo de si hay gastos o excedentes respectivamente, un *Label* para identificar el número de hogar y un par de *Edit Field* que muestran un mensaje de *Consumiendo*, *Excedente* o *No consumo* y otro que muestra el valor numérico de la potencia.

Consumo Neto	Flujos de Energía		
⚡ Hogar 1	Consumiendo	210.9	W
⚡ Hogar 2	Consumiendo	218.7	W
⚡ Hogar 3	Consumiendo	233.9	W
⚡ Hogar 4	Consumiendo	224.6	W

Figura 4.7 Panel de consumo neto

En el panel de flujos de energía tenemos un *Grid* de seis filas correspondientes a los seis flujos de potencia establecidos. Aprovechamos los iconos de la figura 4.4 para mostrar el estado de compartir o no.

Consumo Neto	Flujos de Energía		
🔗 Enlace 1 - 2	111.6		W
🔗 Enlace 1 - 3	112.2		W
🔗 Enlace 1 - 4	5.898		W
🔗 Enlace 4 - 2	105.7		W
🔗 Enlace 4 - 3	106.3		W
🔗 Enlace 2 - 3	0.5871		W

Figura 4.8 Panel de flujos de energía

4.3.4 Panel de información

Este panel ofrece datos identificativos sobre la microrred como es el nombre de la urbanización, barrio o edificio, la ciudad en la que se encuentra y una breve información meteorológica. El icono sobre la información meteorológica es un simple adorno debido a que carecemos de un componente que pueda implementar *APIs* de fuentes meteorológicas en línea. Se ha puesto para tener una idea de cómo puede ser la aplicación desde el punto de vista gráfico.

4.3.5 Panel de reproducción

El panel de reproducción simplemente está constituido por un botón (*Button*) con el carácter “▶” que

comienza la simulación acompañada de un *Edit Field* que muestra la hora real en formato digital.

4.3.6 Detalles gráficos

Para dar una imagen de modernidad y profesionalidad, se ha optado por rematar los cajones con una sombra en formato *PNG*. Los cajones y el fondo, además, son coloreados por una imagen degradada o gradiente.



Figura 4.9 (Dcha. a izq.) Sombra, fondo de cajón y fondo de app

4.4 Comportamiento de la app

Con el modo *Code View* podemos ver el código que gobierna el comportamiento de la aplicación. En él se encuentran las sentencias condicionales que hace que se muestre unos datos u otros. *App Designer* habilita las zonas a programar por el desarrollador bloqueando aquellas que deben estar por obligación, pero que no deben ser tocadas para evitar fallos de compilación.

4.4.1 Parte no editable

Un código de *App Designer* crea automáticamente una estructura que es común en todos los programas creados con este entorno. Se suelen definir las siguientes partes:

- Propiedades de los componentes: Parte en la que se declaran los componentes usados en la vista de diseño.
- Callbacks: Es una función que se ejecuta en respuesta a una acción específica realizada por el usuario. Esta función se asocia al componente colocado en la interfaz. Por ejemplo, un botón, un deslizador, un menú desplegable... Cabe destacar un *Callback* llamado *ButtonValueChanged* que inicializa la simulación cuando clicamos sobre el botón de *Play*.

```
function ButtonValueChanged(app, event)
    app.simulacion();
end
```

- Inicialización de los componentes: En esta parte del código se crean e inicializan los componentes dándoles tamaño, color, nombre, método de escala...
- Creación y eliminación de la app: En este último trozo se montan los componentes en la aplicación, se registra como *app* y se inicializa. Finalmente, una línea de código es la encargada de eliminar la *app* de la pantalla cuando clicamos sobre el botón de cerrar.

4.4.2 Comunicación simulación-app

Para usar la variable tipo *SimulationOutput* que genera *Simulink* llamada *out*, debemos dejar claro que queremos usar elementos del *Workspace* ya que *App Designer* está pensado para hacer apps encapsuladas e independientes, pero dentro del entorno de Matlab. Si las llamamos directamente sin declararlas en la *app*, el compilador no lo entenderá. Por ello, en el primer bloque de propiedades, al no poder editar la parte de

properties de acceso público, debemos crear otra *properties* con la variable *out* dentro.

```
properties (Access = public)
    out
end
```

Seguidamente, en la parte de los *Callbacks*, tomamos aquel que inicializa la aplicación y hacemos que la variable *out* introducida al principio, pase a ser una variable usable por *App Designer* consiguiendo la denominación *app.out*.

```
function startupFcn(app, out)
    app.out=out;
end
```

A partir de este momento, podemos usar la variable en el código principal para extraer datos y demás operaciones. La variable *out* introducida en este programa, contiene estos 15 vectores:

Tabla 4.1 Vectores dentro de *out*

Propiedad	Valor
Bateria1	32709x1 double
Pneta1	32709x1 double
tout	32709x1 double
u12	32709x1 double
u13	32709x1 double
u14	32709x1 double
u23	32709x1 double
u42	32709x1 double
u43	32709x1 double

4.4.3 Código principal

Después de inicializar el programa, preparar las variables provenientes de la simulación y darle al botón de *Play*, el programa inicia la ejecución de la función *simulacion(app)*. Esa función contiene todo lo que hace

posible la animación del programa.

En primer lugar, llamamos x al tiempo de simulación y *hogares* al número de viviendas que componen la microrred. Para que la microrred sea escalable, hay que crear un programa de tal forma que detecte cuantas viviendas hay en la microrred.

Hemos visto que *out* tiene quince vectores de 32709 valores cada uno. Si tenemos 4 hogares en la microrred, habrá 4 vectores para los estados de carga de las 4 baterías, 4 potencias netas, un vector que guarde el tiempo y 6 vectores correspondientes a los flujos de potencia que estudiamos. Saber cuántos hogares forman parte de la microrred es crucial para la posterior animación ya que habrá un bucle que anime los datos donde cada hogar supondrá una iteración. Para generalizar esta identificación se ha seguido la siguiente resolución matemática.

1. Cada hogar comparte potencia con todos los miembros menos consigo mismo. Los flujos que estudiamos son la mitad ya que la otra mitad son los mismos flujos de potencia, pero de signo contrario.

$$Flujos = \frac{Hogares \cdot (Hogares - 1)}{2}$$

Ecuación 4-1 Cálculo de flujos en la microrred

2. A estos flujos, sumamos el número de miembros de la microrred que es el mismo que potencias netas y baterías instaladas. Además, sumamos el vector de tiempo. El resultado de esta suma es la longitud de la variable *out* ($Hogares \equiv H$)

$$Hogares + Potencias Netas + Flujos + tout = H + H + \frac{H(H-1)}{2} + tout = Longitud(out)$$

Ecuación 4-2 Longitud de *out*

3. Si restamos el lugar que ocupa el vector de tiempo *tout* a la longitud total, nos queda una expresión en función de H, el dato que necesitamos saber. Reordenando:

$$2 \cdot H + \frac{H^2}{2} - \frac{H}{2} = \frac{H^2}{2} + \frac{3 \cdot H}{2} = Longitud(out) - 1$$



$$H^2 + 3 \cdot H - 2 \cdot (Longitud(out) - 1) = 0$$

Ecuación 4-3 Ecuación de 2º grado obtenida

4. Resolviendo la ecuación de segundo grado obtenida y quedándonos con la solución positiva, el número de hogares en función de la longitud de la salida *out* (Obteniendo los datos correctamente) es:

$$H = \frac{-3 + \sqrt{9 + 4 \cdot 2 \cdot (Longitud(out) - 1)}}{2}$$

Ecuación 4-4 Número de hogares en función de la longitud de *out*

Seguidamente implementamos el código que adapta los 32709 valores del vector tiempo (*tout*) a un formato de 24 horas. Este formateo se logra extrayendo la parte entera del número decimal para determinar la hora, mientras que la fracción restante se multiplica por 60 para obtener los minutos correspondientes. Así, por ejemplo, un valor de tiempo como 1.5 se transforma en 01:30, indicando la una y media de la madrugada.

Con la orden *get* podemos extraer los vectores dentro de *out*. Para cada valor de tiempo, animamos cada uno de los iconos correspondientes a cada uno de los hogares por lo que hemos implementado un bucle con tantas iteraciones como hogares hay en la microrred. Dentro de este bucle simplemente tenemos una consecución de sentencias *if*.

Nos vamos a centrar en los cuatro bloques de código principales basados en condicionantes:

- Panel de consumo neto: Si la potencia neta es positiva, mostraremos el icono del rayo e imprimiremos *Consumiendo*. De lo contrario, mostraremos el icono de la hoja e imprimiremos *Excedente*.
- Batería: El componente *Image* dedicado a mostrar la ilustración de la batería sigue siendo el mismo para cualquiera de los casos, pero en él se mostrará una imagen de la pila más o menos cargada dependiendo del estado de carga leído en la iteración en cuestión. Dicho esto, la pila irá descargándose (cambiando de imagen) a medida que vaya desplazándose a través de los siguientes tramos: $85 < SOC < 100$, $60 < SOC < 85$, $40 < SOC < 60$, $20 < SOC < 40$ y $0 < SOC < 20$.
- Flujos de potencia: Estudiamos cada uno de los flujos de potencia dentro de *out* implementando una sentencia condicional. Gracias a un componente *Image* también, dependiendo del signo del flujo, mostraremos una flecha hacia un lado u otro. Si el flujo es cero, mostramos una “imagen” transparente, es decir un *PNG* vacío.
- Icono de compartir: Dentro de cada cajón de estado del hogar, mostramos si la vivienda está compartiendo potencia con los demás. Si esta comparte con al menos un vecino, mostraremos el icono de compartir e imprimiremos *Comparte*. Si, por el contrario, todos los flujos de potencia son cero, mostraremos el icono de compartir tachado e imprimiremos *No comparte*.

Finalmente, imprimimos la hora actual y pausamos las animaciones durante un pequeño lapso para poder simular el paso del tiempo con cierta velocidad.

Estos bloques son los que forman el código principal que simulan el comportamiento de una app de monitoreo de una microrred real con posibilidad de escalar a más miembros. Para más detalle, consulte el *Anexo*.

Se puede ver una demostración de la app mediante este [enlace](#).

5 CONCLUSIONES

Este proyecto tiene una visión innovadora sobre el control distribuido de las redes de microrredes, las cuales pueden ofrecer una respuesta a los desafíos energéticos actuales. La implementación de microrredes en entornos aislados, acompañada de sistemas de control distribuido, puede significar una reducción sustancial en los costos de instalación al reducir la necesidad de baterías de gran capacidad. Al permitir el intercambio de potencia entre ellas, es posible gestionar las capacidades de almacenamiento de manera mucho más eficiente, minimizando el desperdicio y maximizando la durabilidad del soporte energético.

Se vislumbra un futuro en el que las nuevas urbanizaciones puedan incorporar, desde su concepción, sistemas fotovoltaicos interconectados que no solo aumenten el valor de las propiedades, sino que también proporcionen un atractivo considerable para los compradores. Este modelo no solo favorece económicamente a los vendedores y compradores, sino que también contribuye a la independencia energética de los propietarios, emancipándolos de la red eléctrica tradicional y fomentando la sostenibilidad.

Los desafíos futuros incluirán la gestión de flujos de potencia bidireccionales y las consideraciones relativas al sincronismo y la infraestructura requerida para soportar tales sistemas. En términos de aplicabilidad, proyectamos un entorno en el cual los miembros de la microrred puedan monitorear y controlar el estado energético de sus viviendas mediante dispositivos inteligentes, ofreciendo la flexibilidad para desconectarse de la red y ajustar configuraciones según sea necesario.

Este proyecto no solo demuestra la viabilidad técnica y económica de las microrredes en entornos residenciales, sino que también allana el camino hacia una era de autonomía energética que es ambientalmente sostenible y económicamente beneficiosa. La implementación exitosa de estas tecnologías podría revolucionar el modo en que concebimos y gestionamos nuestro consumo energético, marcando un precedente significativo en la historia de la gestión energética residencial.

En conclusión, este proyecto ha unificado los múltiples usos que le hemos dado a Matlab a lo largo de varias asignaturas de esta carrera universitaria. Desde teoría de control hasta electrónica de potencia. Todo ello aplicado a un campo en auge que requerirá de numerosos esfuerzos por parte de todos los futuros profesionales del sector.

Por mi parte, la investigación y la curiosidad han sido claves para ir avanzando poco a poco, asentando conceptos y, pese a la complejidad técnica que se presuponía en un principio, se han logrado resultados prometedores e ilustrativos. En mi opinión, la implementación de este tipo de sistemas en las comunidades civiles debería de ser esencial si queremos reducir la huella de carbono a vista de proteger nuestra salud y nuestro planeta. No tengo dudas de que este tipo de sistemas se volverán mucho más presentes y comunes en nuestro día a día.

Código 1: Aplicación de *App Designer*

```
classdef Aplicacion < matlab.apps.AppBase

    % Properties that correspond to app components
    properties (Access = public) ...

    properties (Access = public)
        out
    end

    methods (Access = private)
        % Prepara los datos para la animación
        function simulacion(app)
            x=app.out.tout; %Tiempo de simulación
            hogares=(-3+sqrt(9+4*2*(length(app.out.who)-1)))/2; %Restamos 1 porque
            tenemos también el array del tiempo

            % Cell array para guardar las cadenas de tiempo formateadas
            horasFormateadas = cell(size(x));

            for i = 1:length(x)
                % Extrae el valor actual del tiempo
                tiempoActual = x(i);

                % Calcula horas y minutos
                horas = floor(tiempoActual);
                minutos = floor((tiempoActual - horas) * 60);

                % Formatea la hora en hh:mm
                horasFormateadas{i} = sprintf('%02d:%02d', horas, minutos);
            end

            %Extraemos los enlaces de energía
            u12=app.out.get('u12');
            u13=app.out.get('u13');
            u14=app.out.get('u14');
            u23=app.out.get('u23');
            u42=app.out.get('u42');
            u43=app.out.get('u43');

            % Dibujamos los datos de cada hogar
            format bank
            for i=1:length(x)
                for k=1:hogares
                    p=app.out.get(['Pneta' num2str(k)]); %Sacamos Pneta de la salida
                    de Simulink para cada hogar
                    b=app.out.get(['Bateria' num2str(k)]); %Sacamos Bateria de la
                    salida de Simulink para cada hogar
                    app.(['PNetaEditField_' num2str(k)]).Value = p(i);
                    if p(i)>0
                        app.(['PanelConsume' num2str(k)]).Value = 'Consumiendo';
                    end
                end
            end
        end
    end
end
```

```

        app.(['IconoConsumo' num2str(k)]).ImageSource = 'flash.png';
elseif p(i)<0
    app.(['PanelConsume' num2str(k)].Value = 'Excedente';
    app.(['IconoConsumo' num2str(k)]).ImageSource = 'leaf.png';
else
    app.(['PanelConsume' num2str(k)].Value = 'No Consumo';
end
app.(['EditFieldSoC_' num2str(k)].Value = b(i);

% Cambiamos la imagen solo si el estado de carga ha cambiado
if b(i)<=100 && b(i)>85
    app.(['Imagen_Pila_' num2str(k)]).ImageSource =
'Bateria100.png';
elseif b(i)<=85 && b(i)>60
    app.(['Imagen_Pila_' num2str(k)]).ImageSource =
'Bateria80.png';
elseif b(i)<=60 && b(i)>40
    app.(['Imagen_Pila_' num2str(k)]).ImageSource =
'Bateria60.png';
elseif b(i)<=40 && b(i)>20
    app.(['Imagen_Pila_' num2str(k)]).ImageSource =
'Bateria40.png';
elseif b(i)<=20 && b(i)>0
    app.(['Imagen_Pila_' num2str(k)]).ImageSource =
'Bateria20.png';
end
end

%Direcciones de los hogares
if u12(i)>0
    app.('flujo12').ImageSource = 'flujo0-2.gif';
elseif u12(i)<0
    app.('flujo12').ImageSource = 'flujo180-1.gif';
elseif u12(i)==0
    app.('flujo12').ImageSource = 'transparencia.png';
end
app.("FlujoW").Value = u12(i);

if u23(i)>0
    app.('flujo23').ImageSource = 'flujo225-3.gif';
elseif u23(i)<0
    app.('flujo23').ImageSource = 'flujo45-2.gif';
elseif u23(i)==0
    app.('flujo23').ImageSource = 'transparencia.png';
end
app.("FlujoW_6").Value = u23(i);

if u14(i)>0
    app.('flujo14').ImageSource = 'flujo315-4.gif';
elseif u14(i)<0
    app.('flujo14').ImageSource = 'flujo135-1.gif';
elseif u14(i)==0
    app.('flujo14').ImageSource = 'transparencia.png';
end
app.("FlujoW_3").Value = u14(i);

if u43(i)>0
    app.('flujo34').ImageSource = 'flujo180-3.gif';
elseif u43(i)<0
    app.('flujo34').ImageSource = 'flujo0-4.gif';
elseif u43(i)==0

```

```

        app('flujo34').ImageSource = 'transparencia.png';
    end
    app("FlujoW_5").Value = u43(i);

    if u13(i)>0
        app('flujo13').ImageSource = 'flujo270-3.gif';
    elseif u13(i)<0
        app('flujo13').ImageSource = 'flujo90-1.gif';
    elseif u13(i)==0
        app('flujo13').ImageSource = 'transparencia.png';
    end
    app("FlujoW_2").Value = u13(i);

    if u42(i)>0
        app('flujo24').ImageSource = 'flujo90-2.gif';
    elseif u42(i)<0
        app('flujo24').ImageSource = 'flujo270-4.gif';
    elseif u42(i)==0
        app('flujo24').ImageSource = 'transparencia.png';
    end
    app("FlujoW_4").Value = u42(i);
    %Icono de compartir
    if u12(i) == 0 && u13(i) == 0 && u14(i) == 0
        app('Imagen_Comp_1').ImageSource = 'nocompartir.png';
        app('Cartel_Comp_1').Value='No comparte';
    else
        app('Imagen_Comp_1').ImageSource = 'compartir.png';
        app('Cartel_Comp_1').Value='Comparte';
    end
    if u12(i) == 0 && u23(i) == 0 && u42(i) == 0
        app('Imagen_Comp_2').ImageSource = 'nocompartir.png';
        app('Cartel_Comp_2').Value='No comparte';
    else
        app('Imagen_Comp_2').ImageSource = 'compartir.png';
        app('Cartel_Comp_2').Value='Comparte';
    end
    if u13(i) == 0 && u23(i) == 0 && u43(i) == 0
        app('Imagen_Comp_3').ImageSource = 'nocompartir.png';
        app('Cartel_Comp_3').Value='No comparte';
    else
        app('Imagen_Comp_3').ImageSource = 'compartir.png';
        app('Cartel_Comp_3').Value='Comparte';
    end
    if u14(i) == 0 && u42(i) == 0 && u43(i) == 0
        app('Imagen_Comp_4').ImageSource = 'nocompartir.png';
        app('Cartel_Comp_4').Value='No comparte';
    else
        app('Imagen_Comp_4').ImageSource = 'compartir.png';
        app('Cartel_Comp_4').Value='Comparte';
    end

    app('Hora').Value = horasFormateadas{i};
    pause(0.0005);
end

end
end

% Callbacks that handle component events

```

```
methods (Access = private)

    % Code that executes after component creation
    function startupFcn(app, out)
        app.out=out;
    end

    % Value changed function: Button
    function ButtonValueChanged(app, event)
        app.simulacion();
    end
end

% Component initialization
methods (Access = private)

    % Create UIFigure and components
    function createComponents(app) (...)
end

% App creation and deletion
methods (Access = public)

    % Construct app
    function app = Aplicacion(varargin) (...)

    % Code that executes before app deletion
    function delete(app) (...)
end
end
```

REFERENCIAS

- [1] I. Fariza, «El País,» 01 06 2023. [En línea]. Available: <https://elpais.com/economia/2023-06-01/la-guerra-dispara-en-un-40-el-ritmo-de-instalacion-de-renovables-en-europa.html>. [Último acceso: 04 04 2024].
- [2] M. p. I. T. E. y. e. R. Demográfico, «Balance Energético de España,» 2021-2022.
- [3] A. G. Lobo, «El Orden Mundial,» 2021. [En línea]. Available: <https://elordenmundial.com/mapas-y-graficos/regiones-union-europea-potencial-energia-solar/>. [Último acceso: 04 04 2024].
- [4] U. E. Fotovoltaica, «UNEF,» 23 01 2023. [En línea]. Available: <https://www.unef.es/es/comunicacion/comunicacion-post/el-autoconsumo-fotovoltaico-instalado-en-espana-crecio-un-108-respecto-a-2021>. [Último acceso: 2024 04 04].
- [5] E. Fernández, «Sotysolar,» 23 01 2024. [En línea]. Available: <https://sotysolar.es/blog/precio-bateria-placas-solares#:~:text=El%20precio%20de%20una%20bater%C3%ADa%20para%20placas%20solares%20o%20entre,el%20tama%C3%B1o%20de%20tu%20instalaci%C3%B3n..> [Último acceso: 04 04 2024].
- [6] Funcas, «Focus on Spanish Society,» Madrid, 2021.
- [7] I. Morales, «En el mundo todavía viven 733 millones de personas sin electricidad,» *El Confidencial*, 09 06 2022.
- [8] Office of Electricity Delivery and Energy Reliability, «DOE Microgrid Workshop Report,» San Diego, California, 2011.
- [9] Honnold Foundation, «Adjuntas: A Solar Community,» [En línea]. Available: <https://www.honnoldfoundation.org/adjuntas-a-solar-community#adjuntas>. [Último acceso: 05 04 2024].
- [10] Schneider Electric, «How Montgomery County gained microgrids and electrical upgrades for no money down,» Montgomery, Maryland, EEUU, 2019.
- [11] Unión Europea, «Sistema de Información Geográfica Fotovoltaica (PVGIS),» [En línea]. Available: https://re.jrc.ec.europa.eu/pvg_tools/en/#MR. [Último acceso: 20 02 2024].
- [12] J. Mateo, «www.powen.es,» 30 11 2021. [En línea]. Available: <https://powen.es/temperatura/#:~:text=La%20temperatura%20afecta%20de%20forma,la%20eficiencia%20de%20las%20placas..> [Último acceso: 14 04 2024].
- [13] Autor, «Este es el ejemplo de una cita,» *Tesis Doctoral*, vol. 2, nº 13, 2012.

- [14] O. Autor, «Otra cita distinta,» *revista*, p. 12, 2001.