



ANEXOS



ANEXO I. ADQUISICIÓN DE DATOS

A continuación, se mostrará de manera breve las bases de datos desde las cuales hemos seleccionado los datos necesarios para llevar a cabo el presente proyecto:

- a) Consumo eléctrico desglose horario del año 2017 [6]:

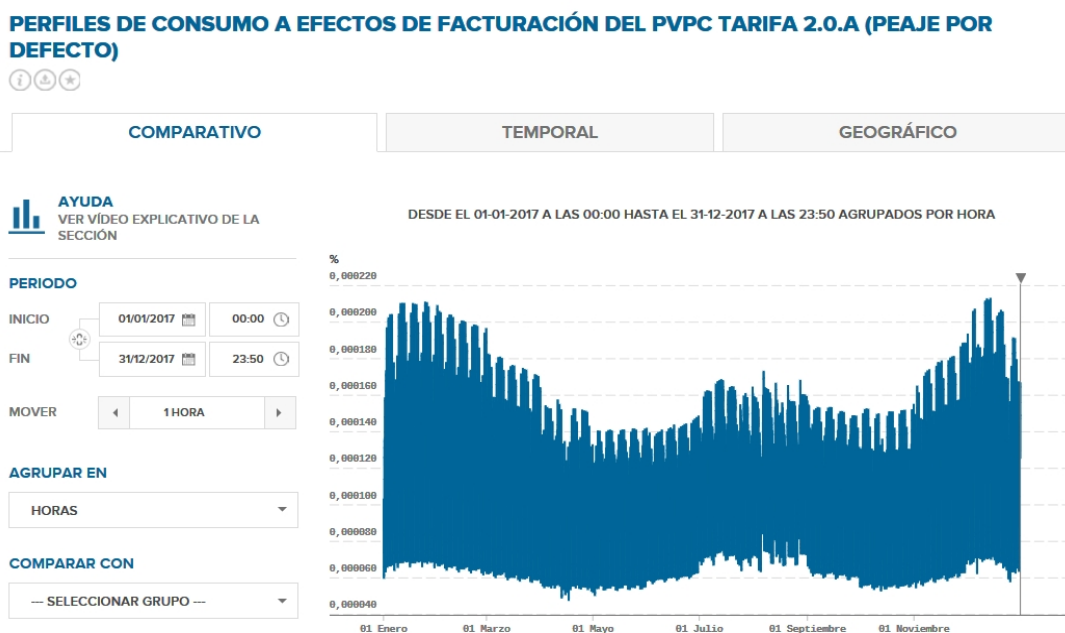


Figura 28: Captura de pantalla ESIOS – Perfil de consumo eléctrico año 2017

- b) Precio de la electricidad desglose horario del año 2017 [6]:

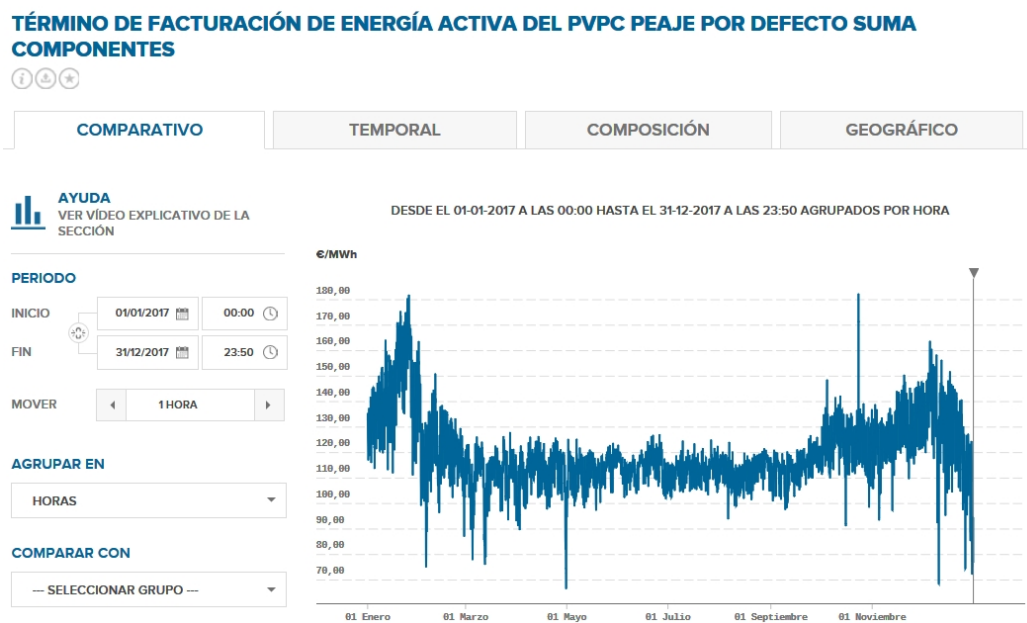


Figura 29: Captura de pantalla ESIOS – Precios desglose horario año 2017



c) Temperatura con desglose horario de Zaragoza del año 2017 [8]:

rp5.lt
real pronóstico

Belarus Lituania Rusia Ucrania Todos los países

Nombre de la ciudad o pueblo

Language Unidad de medida Aplicaciones Versión móvil

Todos los países Rusia Moscú

Archivo de tiempo en Zaragoza (aeropuerto) ver en mapa Archivo de tiempo en el aeropuerto (+23 °C)

pronóstico de tiempo

estación meteorológica (WMO ID) 08160 desde el 1 de febrero del 2005

Consultar el archivo de tiempo Descargar el archivo de tiempo Estadísticas del tiempo

1. Rango de fechas: 01.01.2017 — 31.12.2017

2. Seleccionar para el rango determinado: todos los días sólo mes Mayo sólo fecha 23 de mayo

3. Formato: XLS (Excel) CSV (texto plano)

Mostrar en archivo GZ (archivo) Bajar

Figura 30: Captura de pantalla RP5.it – Temperatura desglose horario año 2017

d) Temperatura, irradiación solar y estimación de generación fotovoltaica de Zaragoza obtenidos desde la base de datos de PVGIS [9]:

JRC CM SAF Sistema de Información geográfica fotovoltaica - mapa interactivo

EUROPA > CE > CCI > IET > RE > SOLAREC > PVGIS > Mapa interactivo > Europa

por ejemplo, "Ispra, Italy" "45.256N, 16.9589E"

zagoza Buscar

posición del cursor: 41.548, -0.719
posición elegida: 41.649, -0.889

Latitud: Longitud: Va a lat/lon

Mapa Satélite

NEW: PVGIS 5 release candidate. Read about it here and try it out!

Estimación FV Radiación mensual Radiación diaria FV autónomo

Irradiancia solar media diaria

Base de datos de radiación: Classic PVGIS

Seleccionar mes: Enero

Irradiancia sobre un plano fijo

Inclinación [0;90] 35 grados (horizontal=0)

Orientación [-180;180] 0 grados (este=-90, sur=0)

☒ Irradiancia media global
☒ Irradiancia global cielo claro
☐ Irradiancia directa normal

Irradiancia sobre un plano con seguimiento a dos ejes

☒ Irradiancia global media, seguimiento a 2 ejes
☒ Irradiancia global cielo claro, seguimiento a 2 ejes

☒ Temperatura durante el día

Fichero del horizonte Examinar...

Formatos de salida

☒ Mostrar gráficos ☐ Mostrar el horizonte

☒ Página web ☐ Fichero de texto ☐ PDF

Calcular [ayuda]

Figura 31: Captura de pantalla PVGIS – Temperatura, irradiación y estimación de generación PV



ANEXOS

La temperatura ($^{\circ}\text{C}$) e irradiación solar (W/m^2) medias mensuales obtenidas desde esta base de datos, después de ordenarlas y obtener valores horarios, se muestran en la siguiente tabla:

Mes	Enero		Febrero		Marzo		Abril	
Hora	temperatura	irradiación	temperatura	irradiación	temperatura	irradiación	temperatura	irradiación
0--1	6,21	0,00	5,69	0,00	9,19	0,00	12,90	0,00
1--2	6,02	0,00	5,37	0,00	8,68	0,00	12,26	0,00
2--3	5,82	0,00	5,05	0,00	8,17	0,00	11,61	0,00
3--4	5,62	0,00	4,74	0,00	7,65	0,00	10,97	0,00
4--5	5,42	0,00	4,42	0,00	7,14	0,00	10,32	0,00
5--6	5,23	0,00	4,10	0,00	6,63	0,00	9,68	7,75
6--7	5,33	0,00	4,30	0,00	7,13	35,75	10,38	89,75
7--8	5,65	23,25	5,00	88,75	8,28	180,25	11,78	228,25
8--9	6,30	173,50	5,95	238,75	9,75	343,75	13,45	370,25
9--10	7,20	306,00	7,05	367,75	11,20	485,00	15,00	490,00
10--11	8,30	400,50	8,25	461,25	12,45	588,00	16,20	576,00
11--12	9,43	454,50	9,33	514,50	13,53	646,50	17,20	624,25
12--13	10,40	465,25	10,30	525,25	14,48	658,00	18,10	634,25
13--14	11,00	432,75	11,03	493,25	15,20	623,25	18,90	605,00
14--15	11,13	358,25	11,23	419,25	15,53	541,75	19,43	537,50
15--16	10,58	244,50	10,80	307,00	15,30	418,50	19,43	434,00
16--17	9,55	91,00	9,85	165,00	14,48	263,25	18,90	300,75
17--18	8,35	0,00	8,65	0,00	13,40	0,00	18,00	0,00
18--19	7,40	0,00	7,60	0,00	12,28	0,00	16,78	0,00
19--20	7,20	0,00	7,28	0,00	11,76	0,00	16,13	0,00
20--21	7,00	0,00	6,96	0,00	11,25	0,00	15,48	0,00
21--22	6,81	0,00	6,65	0,00	10,73	0,00	14,84	0,00
22--23	6,61	0,00	6,33	0,00	10,22	0,00	14,19	0,00
23--00	6,41	0,00	6,01	0,00	9,71	0,00	13,55	0,00

Tabla 12 Temperatura ($^{\circ}\text{C}$), irradiación solar (W/m^2) medias mensuales PVGIS



ANEXOS

Mes	Mayo		Junio		Julio		Agosto	
Hora	temperatura	irradiación	temperatura	irradiación	temperatura	irradiación	temperatura	irradiación
0--1	16,39	0,00	20,78	0,00	23,69	0,00	23,85	0,00
1--2	15,68	0,00	19,98	0,00	22,77	0,00	22,95	0,00
2--3	14,97	0,00	19,18	0,00	21,85	0,00	22,05	0,00
3--4	14,25	0,00	18,38	0,00	20,94	0,00	21,15	0,00
4--5	13,54	0,00	17,58	0,00	20,02	0,00	20,25	0,00
5--6	12,83	43,25	16,78	59,25	19,10	49,00	19,35	16,50
6--7	13,70	127,00	17,65	141,00	19,88	131,75	19,95	101,50
7--8	15,13	267,00	19,10	284,50	21,30	283,25	21,23	252,75
8--9	16,80	406,50	20,85	428,25	23,03	437,75	22,95	415,25
9--10	18,38	523,00	22,50	549,25	24,70	570,25	24,60	556,75
10--11	19,65	607,25	23,93	637,25	26,25	667,50	26,20	661,25
11--12	20,68	655,25	25,15	687,75	27,65	723,00	27,63	721,00
12--13	21,60	664,75	26,25	697,75	28,98	734,25	28,95	733,00
13--14	22,38	636,00	27,10	667,25	30,18	700,75	30,13	697,00
14--15	22,90	569,50	27,70	598,00	31,05	623,75	30,98	614,25
15--16	23,00	468,25	27,88	492,25	31,43	507,75	31,38	490,00
16--17	22,70	338,50	27,60	358,00	31,23	362,00	31,18	335,00
17--18	21,90	0,00	26,83	0,00	30,50	0,00	30,48	0,00
18--19	20,68	0,00	25,58	0,00	29,20	0,00	29,25	0,00
19--20	19,96	0,00	24,78	0,00	28,28	0,00	28,35	0,00
20--21	19,25	0,00	23,98	0,00	27,36	0,00	27,45	0,00
21--22	18,53	0,00	23,18	0,00	26,45	0,00	26,55	0,00
22--23	17,82	0,00	22,38	0,00	25,53	0,00	25,65	0,00
23--00	17,11	0,00	21,58	0,00	24,61	0,00	24,75	0,00

Tabla 13 Temperatura (°C), irradiación solar (W/m²) medias mensuales PVGIS. Continuación



ANEXOS

Mes	Septiembre		Octubre		Noviembre		Diciembre	
Hora	temperatura	irradiación	temperatura	irradiación	temperatura	irradiación	temperatura	irradiación
0--1	19,75	0,00	15,31	0,00	10,01	0,00	6,41	0,00
1--2	19,09	0,00	14,87	0,00	9,81	0,00	6,21	0,00
2--3	18,42	0,00	14,42	0,00	9,61	0,00	6,01	0,00
3--4	17,76	0,00	13,97	0,00	9,40	0,00	5,80	0,00
4--5	17,09	0,00	13,52	0,00	9,20	0,00	5,60	0,00
5--6	16,43	0,00	13,08	0,00	9,00	0,00	5,40	0,00
6--7	16,93	52,50	13,55	7,00	9,25	0,00	5,48	0,00
7--8	18,20	203,50	14,68	119,75	9,83	36,25	5,88	12,75
8--9	19,80	379,75	16,05	284,50	10,68	190,50	6,53	144,25
9--10	21,35	535,75	17,45	430,75	11,65	324,50	7,55	271,50
10--11	22,73	652,00	18,65	539,50	12,63	422,50	8,75	361,75
11--12	23,95	718,50	19,73	601,75	13,50	478,50	9,93	412,75
12--13	25,03	731,75	20,68	614,50	14,28	490,00	10,88	423,25
13--14	26,00	691,75	21,43	576,75	14,75	456,00	11,43	392,25
14--15	26,53	599,50	21,73	490,50	14,83	378,50	11,43	321,25
15--16	26,53	462,00	21,38	361,50	14,30	261,25	10,78	212,75
16--17	25,90	292,25	20,43	202,75	13,30	113,00	9,75	66,50
17--18	24,93	0,00	19,18	0,00	12,18	0,00	8,55	0,00
18--19	23,75	0,00	18,00	0,00	11,23	0,00	7,63	0,00
19--20	23,08	0,00	17,55	0,00	11,02	0,00	7,42	0,00
20--21	22,42	0,00	17,10	0,00	10,82	0,00	7,22	0,00
21--22	21,75	0,00	16,66	0,00	10,62	0,00	7,02	0,00
22--23	21,09	0,00	16,21	0,00	10,42	0,00	6,82	0,00
23--00	20,42	0,00	15,76	0,00	10,21	0,00	6,61	0,00

Tabla 14 Temperatura (°C), irradiación solar (W/m²) medias mensuales PVGIS. Continuación



- e) En la red después de una búsqueda exhaustiva, encontramos datos de irradiación solar en la base de datos de la Agencia Andaluza de la energía [10] en donde se puede conseguir valores de irradiación solar media diaria global, difusa y directa entre otros, todos ellos con desglose horario.

En un principio se intentó hacer uso de estos datos, pero llegamos a la conclusión de que no sería favorable ni realista mezclar varias bases de datos, pero se incluye en este anexo por la dificultad que surgió en encontrarlos.

Figura 32: Captura de pantalla AAE – Radiación solar desglose horario



ANEXO II. PREVISIÓN DEL CONSUMO (LOAD FORECAST)

Para realizar la previsión del consumo que usaremos en nuestro modelo, vamos a utilizar un conjunto de variables que se detallarán a continuación [12], véase Anexo IV:

- Fecha (date)
- Hora (hour)
- Temperatura ambiente y de rocío (Drybulb and Dew temperature)
- Temperatura ambiente media mensual (Average temperature)
- Precio de la electricidad (Electricity price)
- El Consumo del sistema (SysLoad)

Todos estos datos con desglose horario para todo el año 2017.

También introduciremos como dato los días festivos (Holidays), que en este caso se corresponden con los festivos nacionales y autonómicos de Zaragoza.

A partir de aquí se genera una matriz de predictores en la cual se incluyen los datos previamente nombrados, así como los siguientes:

- Día de la semana (Weekday)
- Día laborable (IsWorkingDay)
- Consumo del sistema a la misma hora de la semana pasada (PrevWeekSameHourLoad)
- Consumo del sistema a la misma hora del día anterior (prevDaySameHourLoad)
- Consumo medio del sistema de las últimas 24 horas (prev24HrAveLoad).

En este punto cabe señalar que el consumo del sistema (SysLoad) no forma parte de esta matriz de predictores ya que es el resultado de nuestra previsión.

El siguiente paso consiste en clasificar los datos, acción que se realiza particionando por fecha los datos con los que queremos entrenar una red neuronal (con la ayuda de la herramienta Neuronal Net Fitting véase Anexo IV) y los datos con los que queremos realizar un test.

Nuestros datos quedan listos para ser usados en el Anexo IV.



ANEXO III. PREVISIÓN DE GENERACIÓN PV (PV GENERATION FORECAST)

La hemos realizado de manera muy similar que la previsión del consumo, pero esta vez con valores medios diarios replicados por el número de días de cada mes [12], véase Anexo IV, los datos que vamos a utilizar son:

- Fecha (Date)
- Hora (hour)
- Mes (month)
- Irradiancia media mensual (AverageIrr)
- Ha salido el sol (sol)
- Temperatura media mensual (averageTemp)
- Temperatura de la celda (TempCell)
- Potencia ideal generada (TeoricPower)
- Potencia ideal generada con el 25% de pérdidas (SysPowerGeneration)

También con desglose horario para todo el año 2017.

Para el cálculo de la temperatura de la celda se ha utilizado la siguiente expresión [11]:

$$T_c = T_a + G \cdot \frac{TONC - 20}{800}$$

Donde:

T_c: temperatura de trabajo de la célula (°C).

T_a: temperatura ambiente (°C).

TONC: temperatura de operación nominal de la célula (°C), valor tomado de referencia 47°C.

G: irradiancia (W/m²), hemos usado el valor de la irradiancia global media.

Para el cálculo de la Potencia generada (máxima) por el sistema fotovoltaico se obtiene con la expresión:

$$P_m = P_n \frac{G}{G_{stc}} * [1 - \gamma(T_c - 25)]$$

Donde:

P_n: Potencia nominal de los paneles en condiciones STC (Wp), que fue calculado en 2880 Wp.

G: Irradiancia instantánea (W/m²).

G_{stc}: Irradiancia en condiciones STC (1000 W/m²).

γ: coeficiente de variación de la temperatura de la potencia nominal.

T_c: Temperatura de las células fotovoltaicas.

La temperatura se considera constante para cada hora por lo que finalmente se ha hecho uso de la siguiente expresión abreviada:

$$P_m = P_n \frac{G}{G_{stc}}$$



La cual considera una variación de generación de potencia proporcional con la irradiancia instantánea.

Ahora se genera una matriz de predictores en la cual se incluyen los datos previamente nombrados, así como los siguientes:

- Potencia generada a la misma hora de la semana pasada (PrevWeekSameHourPVGen)
- Potencia generada a la misma hora del día anterior (prevDaySameHourPVGen)

En este punto cabe señalar que Potencia ideal generada con el 25% de pérdidas $PR=0,75$ (SysPowerGeneration), no forma parte de esta matriz de predictores ya que es el resultado de nuestra previsión.

El siguiente paso consiste en clasificar los datos, acción que se realiza particionando por fecha los datos con los que queremos entrenar una red neuronal (con la ayuda de la herramienta Neuronal Net Fitting véase Anexo IV) y los datos con los que queremos realizar un test.

Nuestros datos quedan listos para ser usados en el Anexo IV.



ANEXO IV. USO DE LA HERRAMIENTA NEURAL NET FITTING DE MATLAB

- Abrir la herramienta:

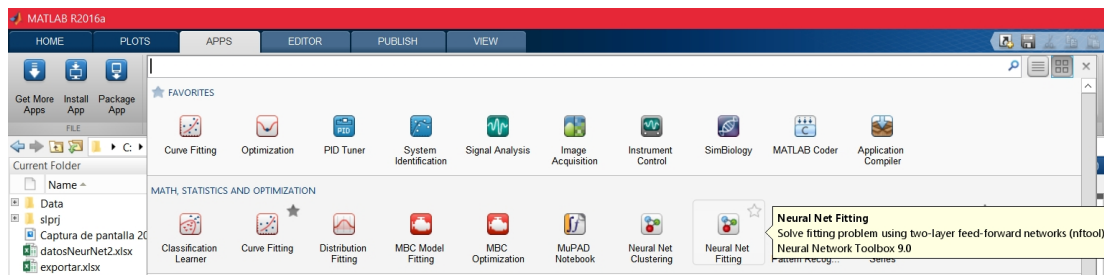


Figura 33: Herramienta Neural Net Fitting (NNF) MATLAB R2016a

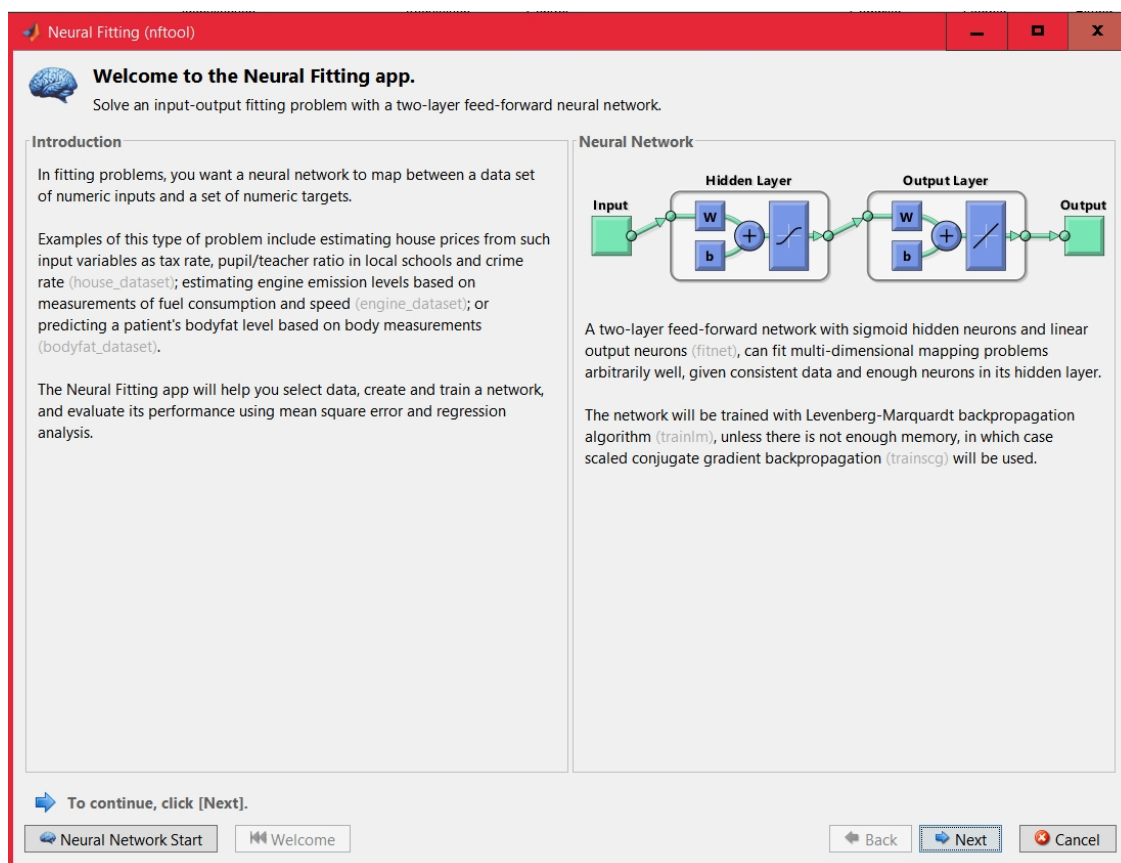


Figura 34: NNF - Portada



- Selección de los datos para entrenar la red:

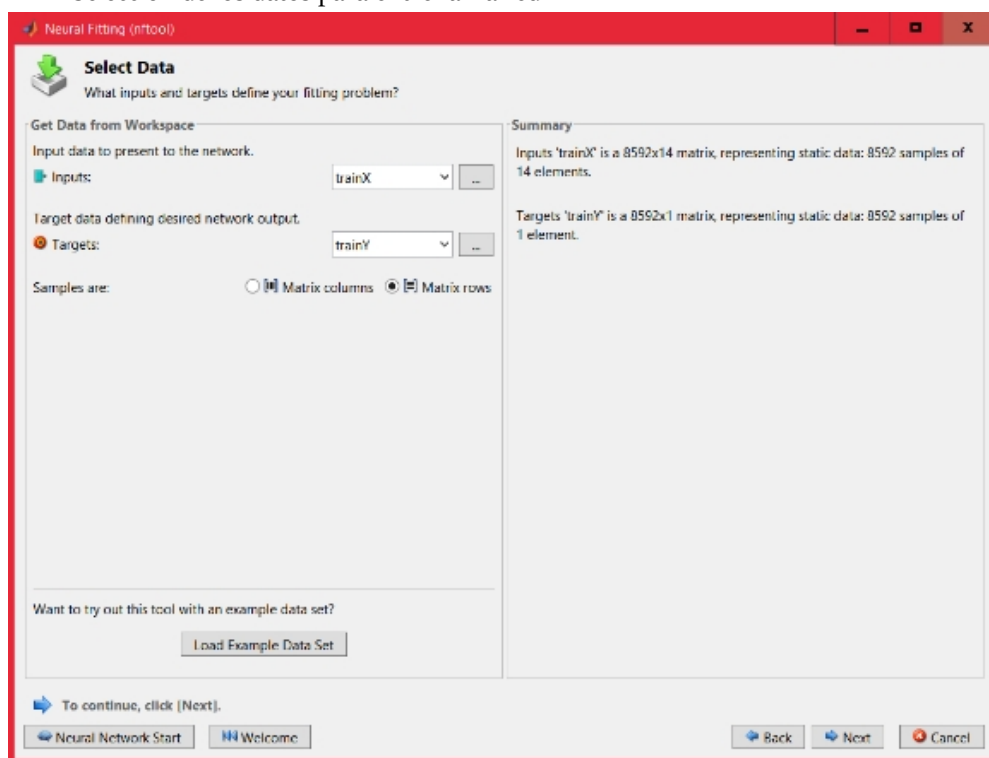


Figura 35: NNF - Selección de datos

- Selección de porcentaje de datos de test y validación:

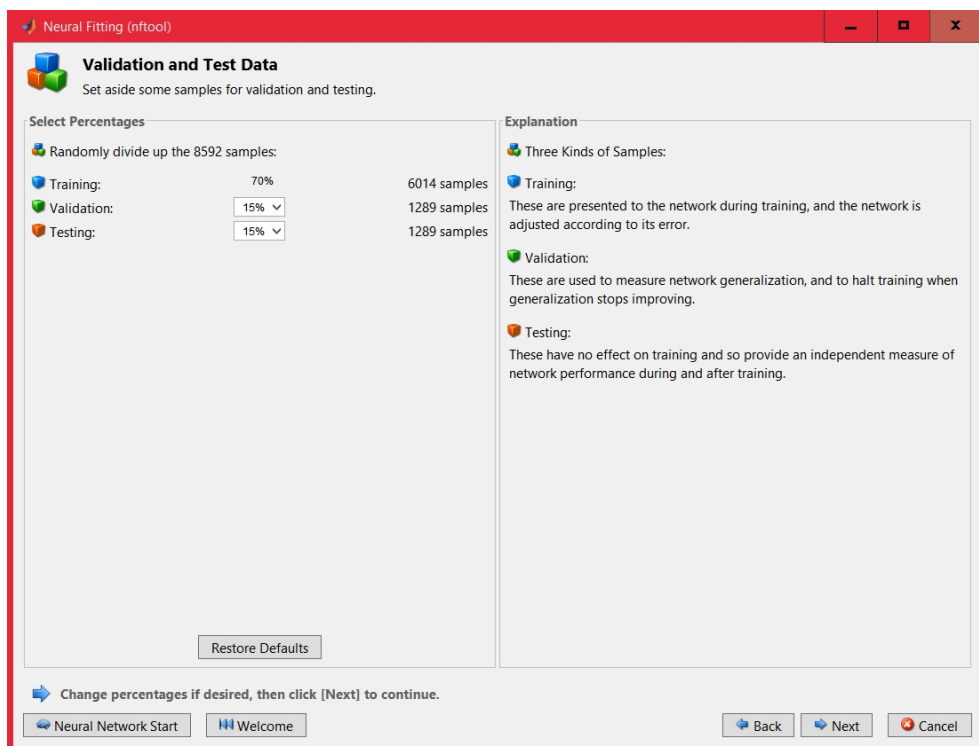


Figura 36: NNF - Porcentaje datos de test y validación



- Arquitectura de la red, elegir el número de capas ocultas para la red:

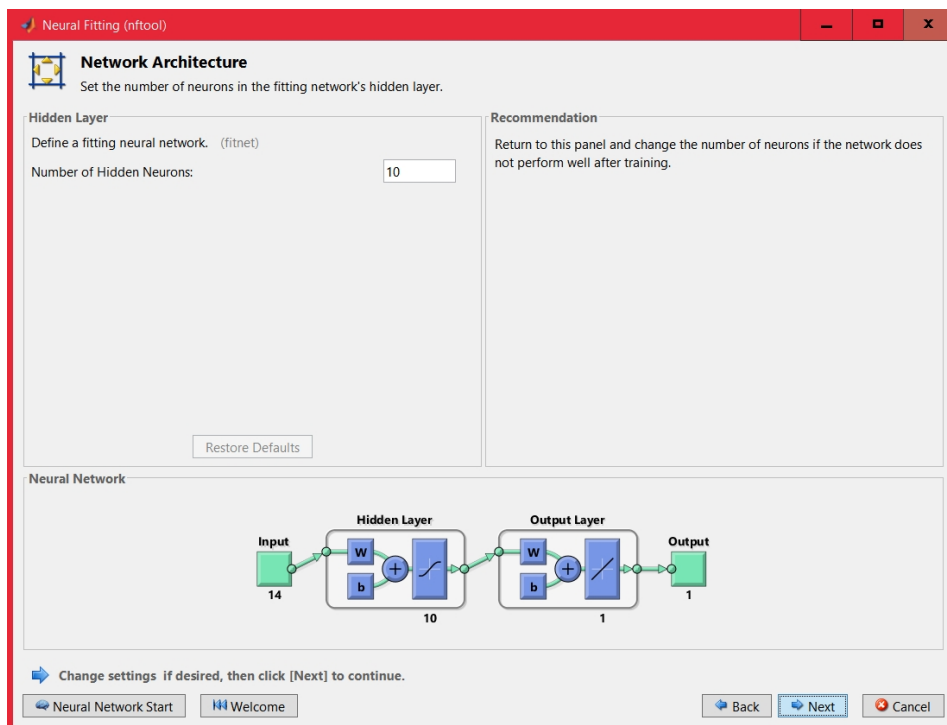
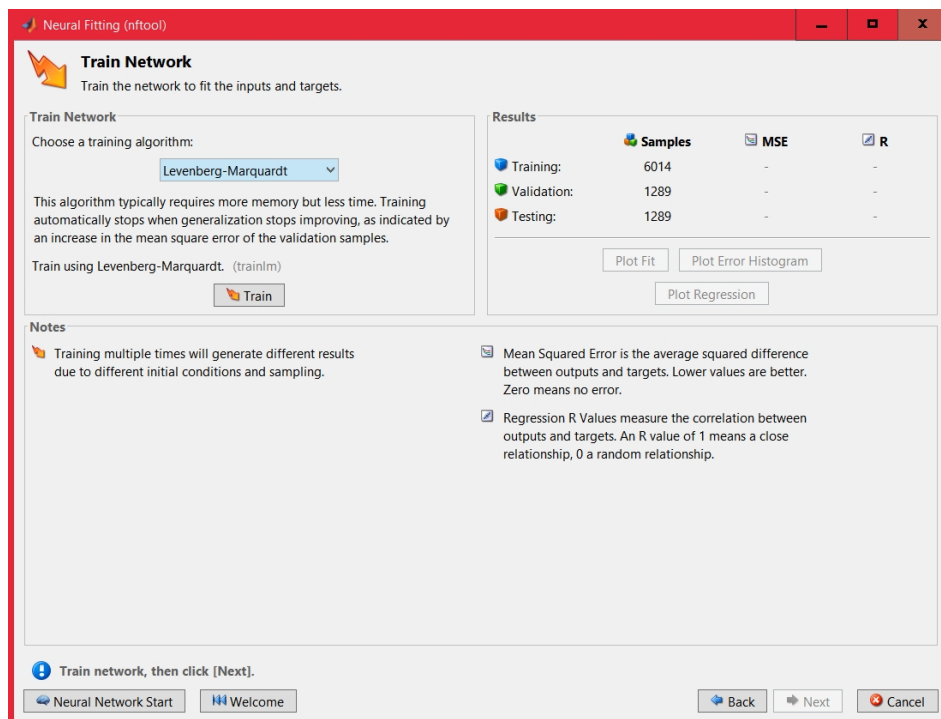


Figura 37: NNF – Selección arquitectura de la red

- Seleccionar tipo de algoritmo a usar en el entrenamiento de la red y entrenarla:



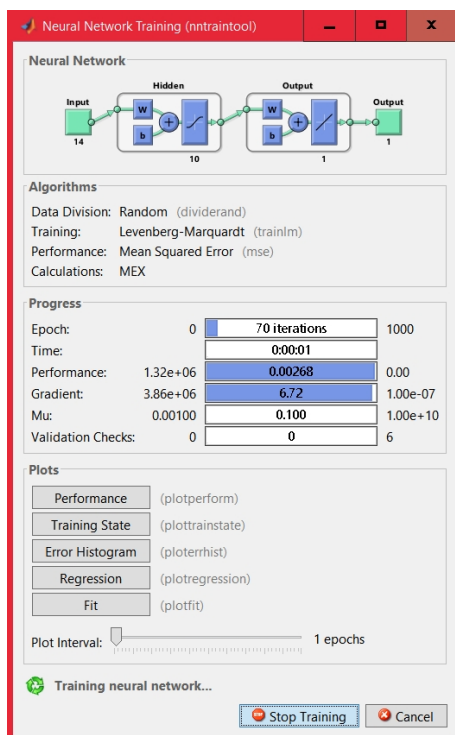


Figura 39: NNF - Entrenamiento en curso

- Evaluar la red: podemos visualizar el gráfico de regresión que nos da una idea del desempeño de la red y realizar un test (si tenemos datos para ello)

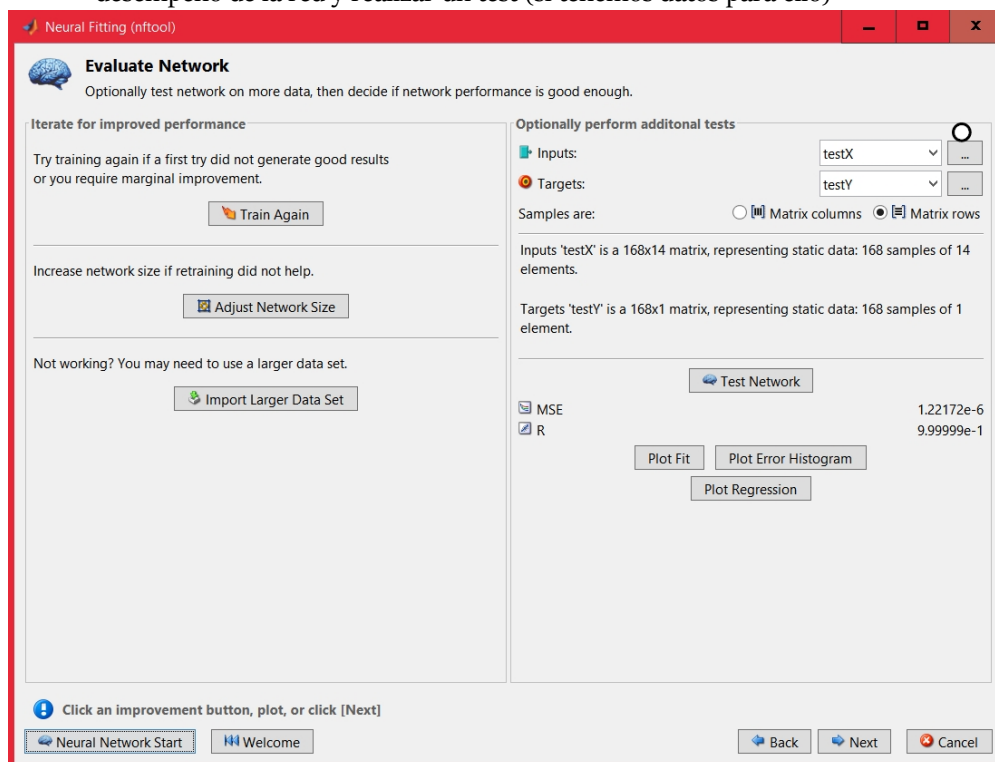


Figura 40: NNF - Evaluación red obtenida

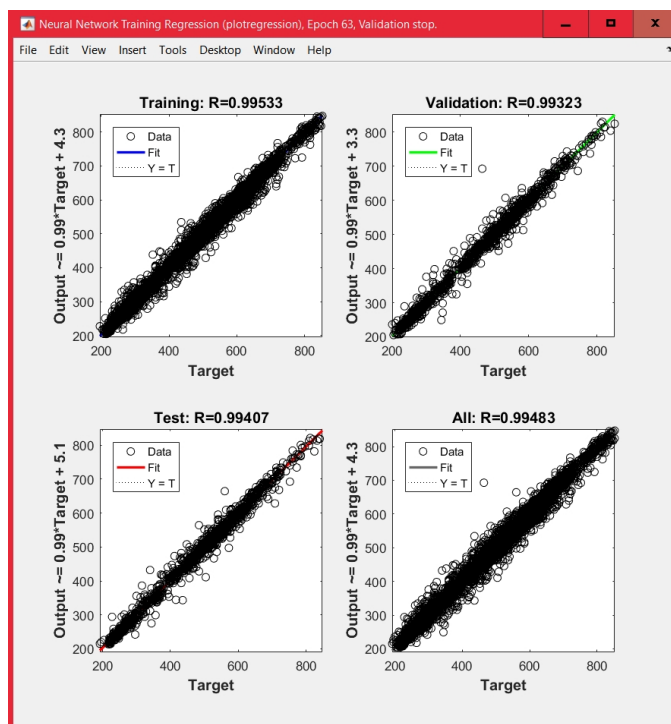


Figura 41: NNF – Gráfico de regresión

- Podemos guardar la red entrenada, generar el código o un bloque para Simulink.

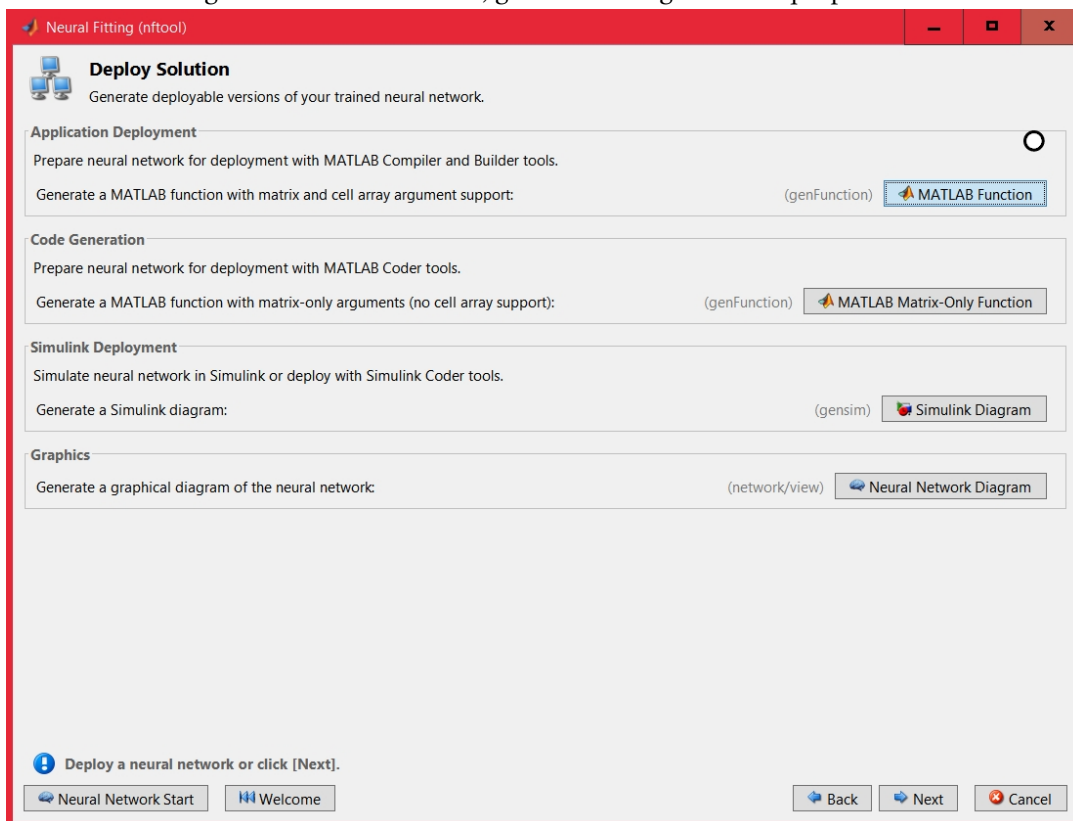


Figura 42: NNF – Despliegue de distintos bloques



- También se puede guardar el script generado

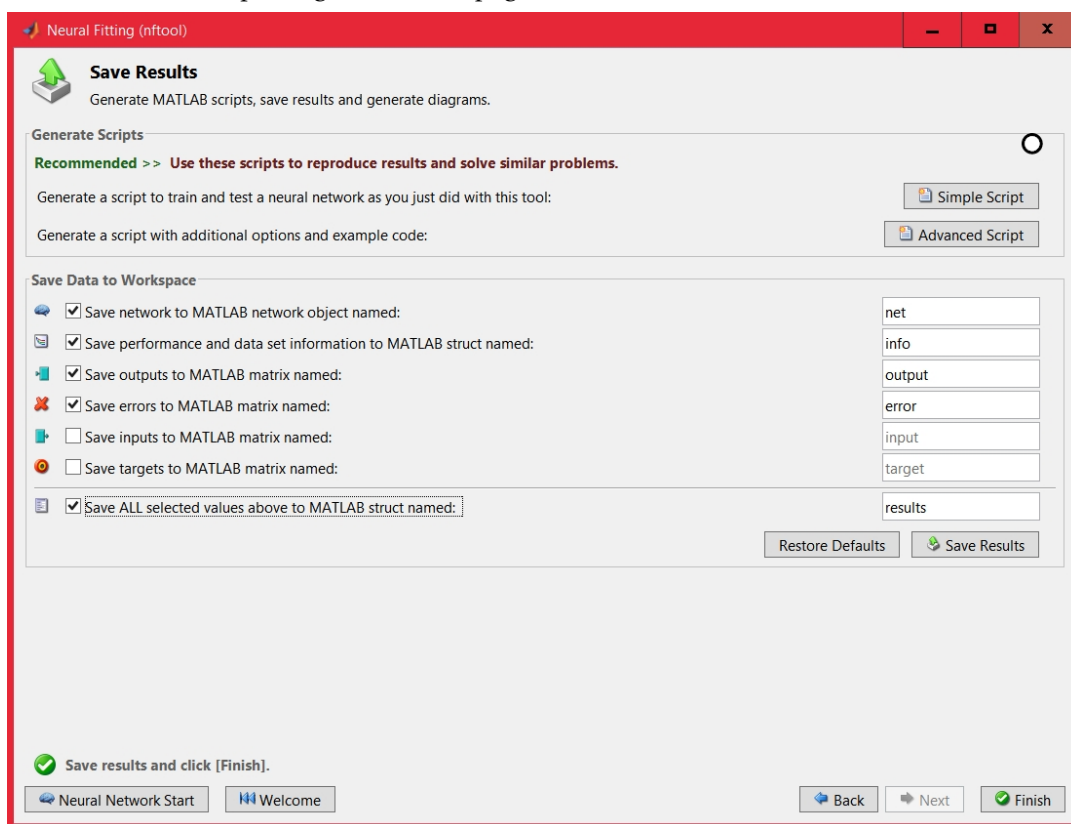


Figura 43: NNF – Guardar resultados y red obtenida

- Podemos realizar todo el proceso comentado anteriormente con el siguiente código:

```
1. net = newfit(trainX', trainY', 10);  
  
    %crear la red e indicarle las variables a usar como  
    %entradas y como salidas, número de capas ocultas  
    %("filtros").  
  
2. net.performFcn = 'mse';  
  
    % tipo de error.  
  
3. net = train(net, trainX', trainY');  
  
    % entrenar la red.  
  
4. save NNModel_PV_zgz.mat net  
  
    % guardar modelo obtenido.
```




CÓDIGO PARA USAR NEURONAL NET FITTING

```
%% Importar datos
load('DataLoad.mat')

%% Importar lista de días festivos desde un archivo Excel
[num, text] = xlsread('C:\Users\...\Data\Holidays.xls');
holidays = text(2:end,1);

%% Generar matriz de predictores
%% La función GenPredictors genera las variables del predictor usadas como
%% entradas para el modelo, para short-term se incluyen:
%% Fecha (date)
%% Hora (hour)
%% Temperatura ambiente y de rocío (Drybulb and Dew temperature)
%% Temperatura ambiente media mensual (Average temperature)
%% Precio de la electricidad (Electricity price)
%% Día de la semana (Weekday)
%% Día laborable (IsWorkingDay)
%% Carga del sistema a la misma hora la semana pasada (PrevWeekSameHourLoad)
%% Carga del sistema a la misma hora del día anterior (prevDaySameHourLoad)
%% Carga media del sistema de las últimas 24 horas (prev24HrAveLoad).

term = 'short';
[X, dates, labels] = GenPredictors(data, term, holidays);
%% Dividir los datos para entrenar la red y para realizar un test
%% En este caso ya que poseemos datos del 2007 se consideran como datos de
%% entrenamiento desde el 1 de enero hasta el 25 de diciembre (ambos
%% inclusive) mientras que los datos de testeo desde el 26 al 31 de diciembre
%% que pueden ser usados para obtener datos de comportamiento de la red.
%% Crear datos de entrenamiento:
trainInd = data.NumDate < datenum('2017-12-25');
trainX = X(trainInd,:);
trainY = data.SYSLoad(trainInd);

%% Crear datos de test y guardarlos
testInd = data.NumDate >= datenum('2017-12-25');
testX = X(testInd,:);
testY = data.SYSLoad(testInd);
testDates = dates(testInd);

save Data\testSet_zgz testDates testX testY
%% clear X data trainInd testInd term holidays dates ans num text %% borrar variables usadas

%% código para crear red neuronal
net = newfit(trainX', trainY', 30); %% crear la red e indicarle las variables a usar como entradas y como salidas,
número de capas ocultas ("filtros").
net.performFcn = 'mse'; %% interpretación del error.
net = train(net, trainX', trainY'); %% entrenar la red.
save NNetModel_zgz.mat net %% guardar modelo obtenido.

%% Usar el modelo obtenido de la Red
%% una vez creado el modelo cargamos los datos de test para probar su
%% comportamiento
cd('Data')
load testSet_zgz
cd ..
forecastLoad = sim(net, testX)';

%% Comparamos la Carga Real y la Prevista
%% creamos un plot con las 2 graficas a modo de comparación, calculamos el
%% error individual, el error porcentual, mean average error (MAE),
%% mean average percent error (MAPE) y daily peak forecast error

err = testY - forecastLoad;
fitPlot(testDates, [testY forecastLoad], err);

errpct = abs(err) ./ testY * 100;

fL = reshape(forecastLoad(1:end), 24, (length(forecastLoad))/24)';
tY = reshape(testY(1:end), 24, (length(testY))/24)';
peakerrpct = abs(max(tY,[],2) - max(fL,[],2)) ./ max(tY,[],2) * 100;

MAE = mean(abs(err));
MAPE = mean(errpct(~isinf(errpct)));

fprintf('Mean Average Percent Error (MAPE): %0.2f%% \nMean Average Error (MAE): %0.2f Wh\nDaily Peak MAPE: %0.2f%%\n', ...
        MAPE, MAE, mean(peakerrpct))

%% Examinar las distribuciones de errores
%% dibujamos la distribución de errores que nos puede dar una visión global
%% del comportamiento de la red
figure;
subplot(3,1,1); hist(err,100); title('Error distribution');
subplot(3,1,2); hist(abs(err),100); title('Absolute error distribution');
line([MAE MAE], ylim); legend('Errors', 'MAE');
subplot(3,1,3); hist(errpct,100); title('Absolute percent error distribution');
line([MAPE MAPE], ylim); legend('Errors', 'MAPE');
```



El código para obtener una red para la previsión de generación *PV* es muy parecido.

Ahora se presenta el código de la función GenPredictors usada en la creación de la matriz de predictores:

```
function [X, dates, labels] = GenPredictors(data, term, holidays)
%% Esta función genera una matriz de predictores para el modelo de previsión de carga del sistema
% Uso:
% [X, dates, labels] = GenPredictors(data, term, holidays)
%
% Entradas:
% data : un array de datos históricos del tiempo y carga del sistema
% term : 'long' or 'short' para indicar el alcance de la previsión.
% por ejemplo si seleccionamos long term no se incluyen valores de carga del día anterior para realizar la
%previsión.
% holidays: un vector de días festivos.
%
% Salidas:
% X : es una matriz de predictores en donde cada fila corresponde a un
% valor horario y cada columna corresponde a una variable.
% dates : un vector de fechas para cada observación nos ayuda a clasificar
% los datos
% labels: Un array con la descripción de cada variable de la matriz de predictores

%% Convertir las fechas en una representación numérica
if ~isfield(data, 'NumDate')
    dates = datenum(data.Date, 'yyyy-mm-dd HH:MM:SS') + (data.Hour-1)/24;
else
    dates = data.NumDate;
    if all(floor(dates)==dates) %Si las fechas no incluyen información horaria
        dates = dates + (data.Hour-1)/24;
    end
end

%% Días festivos
if nargin < 3 || isempty(holidays)
    holidays = createHolidayDates(min(dates), max(dates));
else
    holidays = datenum(holidays, 'dd/mm/yyyy');
end

%% Crear Predictores

%si se elige Short term se incluye la carga del día anterior a la misma
%hora, la carga de la semana anterior a la misma hora y la carga media de
%las últimas 24 horas.
prevDaySameHourLoad = [NaN(24,1); data.SYSLoad(1:end-24)];
prevWeekSameHourLoad = [NaN(168,1); data.SYSLoad(1:end-168)];
prev24HrAveLoad = filter(ones(1,24)/24, 1, data.SYSLoad);

%% Predictor día de la semana
dayOfWeek = weekday(dates);

% Si es día laborable
isWorkingDay = ~ismember(floor(dates), holidays) & ~ismember(dayOfWeek, [1 7]);

%% crear matriz X
if strcmpi(term, 'long', 4);
    % Long Term Forecast Predictors
    X = [daily5dayHighAve daily5dayLowAve data.Hour dayOfWeek isWorkingDay];
    labels = {'Prev5DayHighAve', 'Prev5DayLowAve', 'Hour', 'Weekday', 'IsWorkingDay', 'DryBulb'};
else
    % Short Term
    X = [data.Hour data.DryBulb data.AverageTemp dayOfWeek isWorkingDay prevWeekSameHourLoad prevDaySameHourLoad
    prev24HrAveLoad];
    labels = {'Hour', 'DryBulb', 'AverageTemp', 'Weekday', 'IsWorkingDay', 'PrevWeekSameHourLoad',
    'prevDaySameHourLoad', 'prev24HrAveLoad'};
end

function y = rep24(x) %%repetir cada 24 datos
y = repmat(x(:), 1, 24)';
y = y(:);
```



Distribución de errores

En la figura 44 se muestra la distribución de errores obtenida para la previsión del consumo:

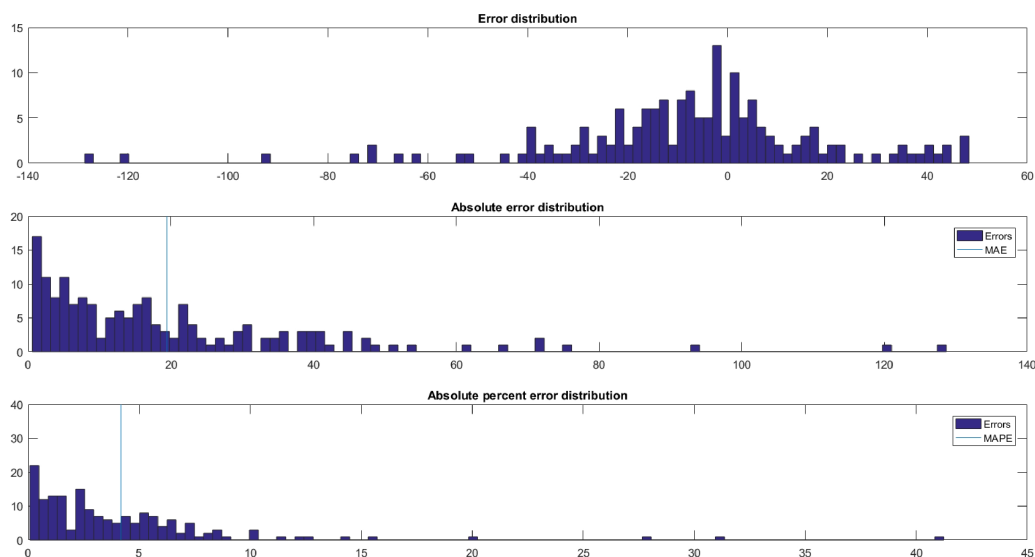


Figura 44: Distribución de errores previsión de consumo

La distribución de error posee forma de campana de Gauss centrada en cero. Éstas gráficas se han realizado con el error obtenido de restar el Consumo real y el obtenida al aplicar la red a los datos de test (la última semana del año 2017).

En la figura 45 se muestra la distribución de errores obtenida para la previsión de la generación PV:

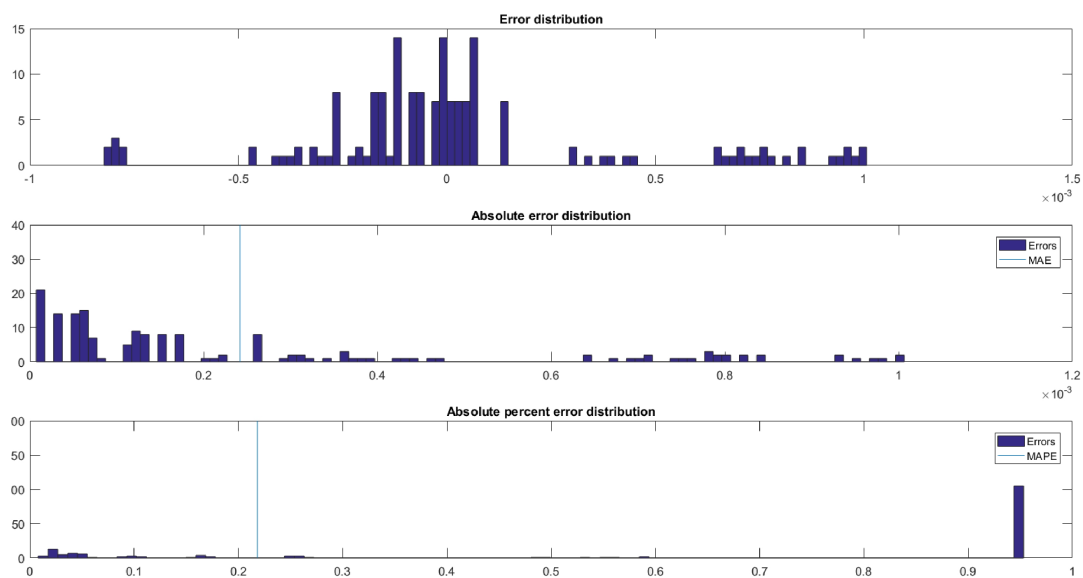


Figura 45: Distribución de errores prevision de generación PV

Los errores tienen orden de magnitud inferiores a los de la previsión del consumo debido a que esta red neuronal ha tomado valores medios mensuales replicados según el número de días de cada mes (véase *tabla Anexo I*).

ANEXO V. MODELO DEL SISTEMA IMPLEMENTADO EN SIMULINK

A continuación, se detalla el sistema implementado en Simulink al cual usa como entradas: el consumo de la red (Wh), la generación fotovoltaica (Wh), el precio de la electricidad (Eur/Wh).

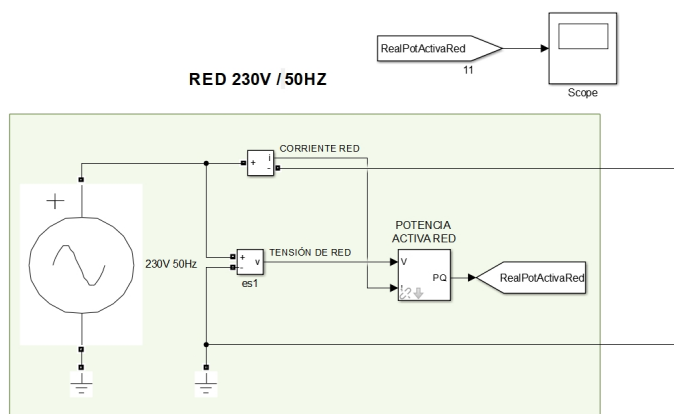


Figura 46: Simulink – Representa la red eléctrica

- El consumo de la carga del sistema (figura 47) se representa con un generador de corriente.

La referencia se multiplica por la constante:

$-(\frac{\sqrt{2}}{230 \cdot 1h})$, ya que el vector de consumo se introduce en Wh.

Se mide la tensión, corriente y se registra los valores de potencia activa para visualizarlos y comprobar que se ha generado correctamente el consumo durante el tiempo de la simulación. En la figura 48 se muestra el resultado de potencia requerido por el consumo en 1 día:

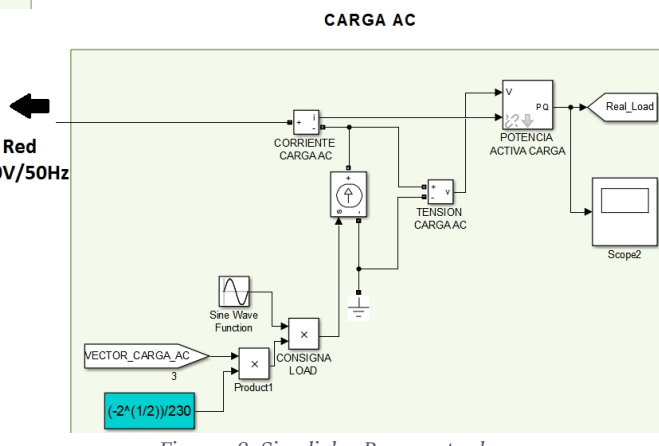


Figura 48: Simulink – Representa el consumo

Figura 47: Simulink – Potencia requerida por el consumo en 1 día

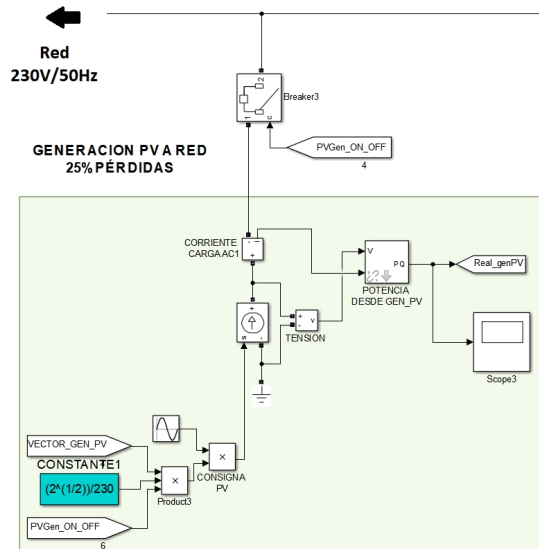


Figura 49: Simulink – Representa Inversor DC-AC

- La generación *PV* del sistema (figura 49) requiere un inversor Dc-Ac y se representa de la siguiente manera:

La referencia se multiplica por la constante:

$\left(\frac{\sqrt{2}}{230 \cdot 1h} \right)$, ya que el vector de generación *PV* se introduce en Wh.

Se mide la tensión, corriente y se registra los valores de potencia activa para visualizarlos y comprobar la generación *PV* durante el tiempo de la simulación.

La figura 50 muestra el resultado de potencia que aporta la generación *PV* a la red en 1 día:

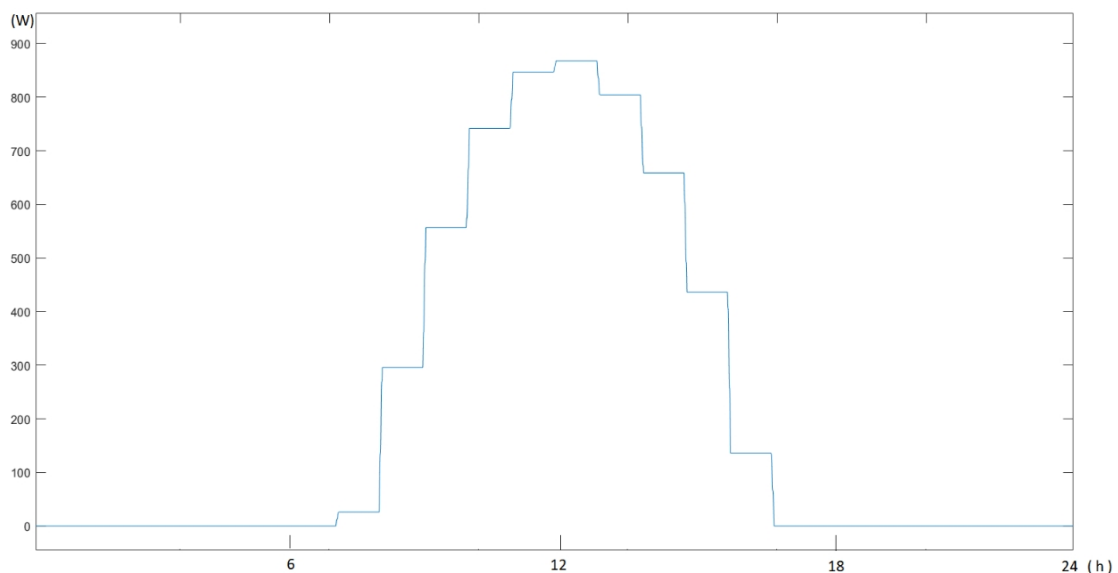


Figura 50: Potencia generada por el sistema fotovoltaico en 1 día

- El inversor/rectificador (figura 51) está representado por el siguiente subsistema el cual tiene como referencia la potencia que cede o consume la batería:

De la misma manera se multiplica por la constante:

$-\left(\frac{\sqrt{2}}{230 \cdot 1h} \right)$, dado que la corriente de descarga de la batería se ha tomado con el signo negativo y la corriente de carga con signo positivo.

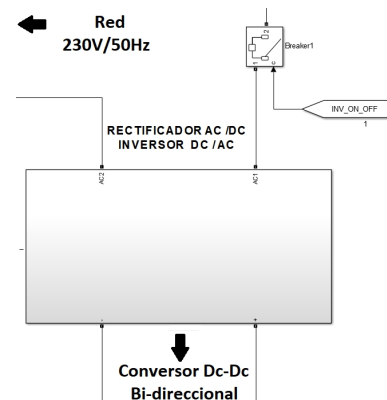


Figura 51: Simulink – Representa Inv DC-AC bidireccional

- Para la carga y descarga de la batería se hace necesario el uso de un convertidor DC-DC Buck-Boost con carácter Bidireccional [13],[18].El esquema propuesto en[13]se muestra en la figura 52.

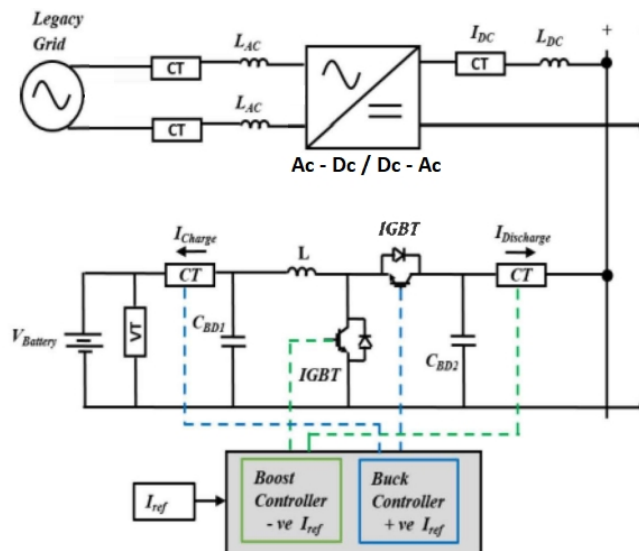


Figura 52: Esquema propuesto en [13]

Los valores de los componentes del convertor son: $L=9\text{mH}$, $C_{BD1}=1200\mu\text{F}$, $C_{BD2}=1200\mu\text{F}$ [13].

La frecuencia de conmutación $f_{\text{switching}}=5\text{KHz}$ para evitar que sean muy notorias las pérdidas por conmutación y por calentamiento [13].

La figura 53 muestra el modelo del convertor DC-DC bidireccional implementado en Simulink [14]:

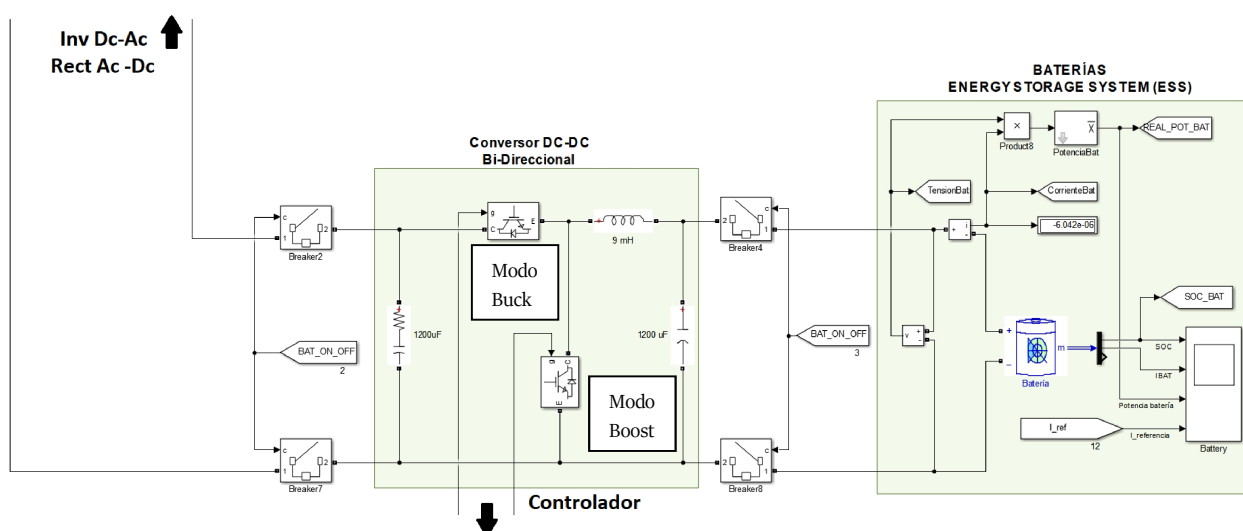


Figura 53: Simulink - Convertor Buck-Boost Bidireccional y Batería

El convertor bidireccional trabaja entre dos tensiones: la tensión de la batería y la tensión de salida del rectificador Ac-Dc.



La batería elegida es una de Plomo ácido (Lead-Acid) que simulink ya incorpora en sus librerías. Los parámetros de configuración que introducimos son:

- Tipo de batería: Lead-Acid, Lithium-Ion, Nickel-Cadmium, Nickel-Metal-Hydride.
- Voltaje nominal de la batería (48 V).
- Capacidad Nominal (72 Ah).
- Estado inicial de carga SOC (70 %).
- Tiempo de respuesta de la batería (30 s).

Los parámetros de descarga se calculan por defecto a partir de los parámetros nominales introducidos anteriormente, pero existe la posibilidad de modificarlos.

Se debe señalar que las descargas profundas y cargas excesivas afectan negativamente a la vida útil de la batería, por lo que se hace necesario imponer límites de SOC para la carga y descarga para intentar extender dicha vida útil. La corriente medida a la salida/entrada de la batería (CorrienteBat) es la que se realimenta al controlador, mientras que el SOC es monitorizado en todo momento por el HEMS.

La figura 54 muestra la curva de descarga de la batería de nuestro modelo de 72Ah - 48V (3456Wh).

Si el consumo es 7.2 A, la descarga total se lleva a cabo en 10 horas., si este consumo es del doble (14,4A) se lleva a cabo en la mitad del tiempo, es decir, en aproximadamente 5 horas.

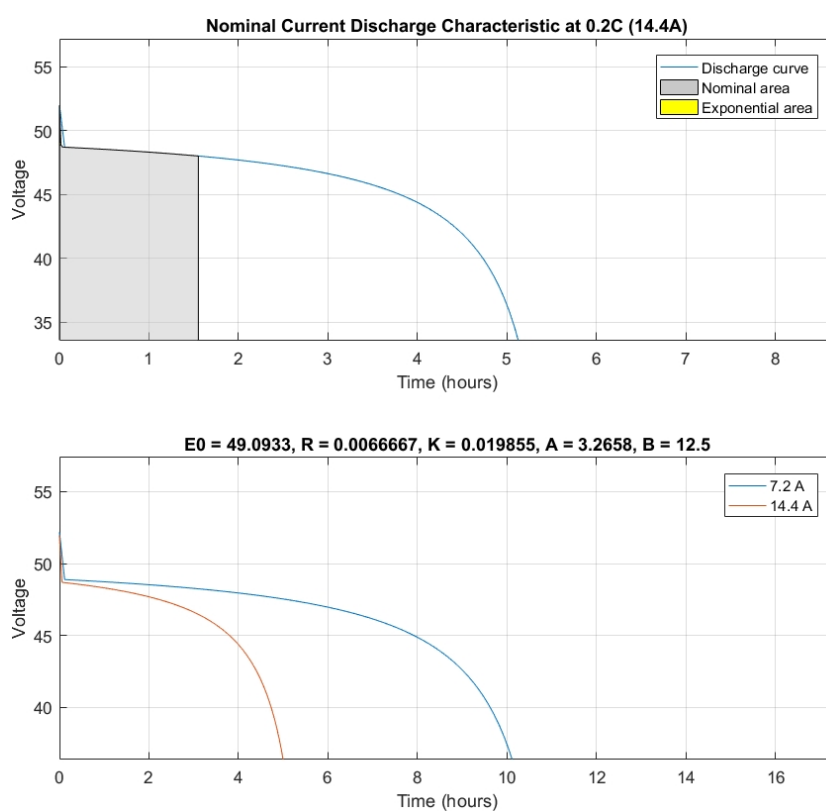


Figura 54: Gráfica de descarga de la batería

- La figura 55 muestra el modelo del control de la conmutación de los IGBT del convertor Bidireccional Buck-Boost implementado en Simulink [14]:

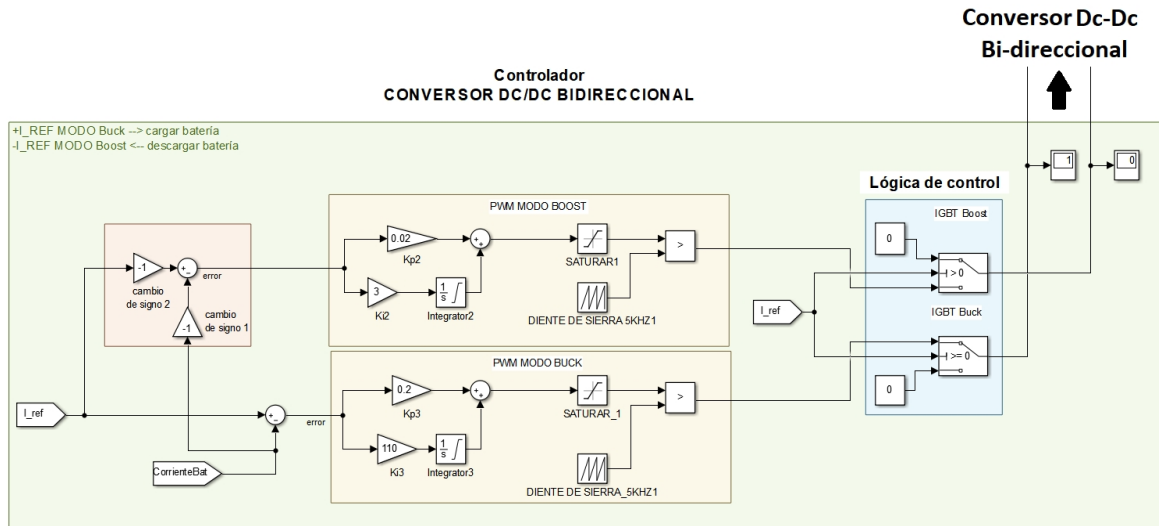


Figura 55: Simulink - Controlador PI convertor DC-DC bidireccional

La corriente de referencia es calculada por el HEMS, la realimentación es la que medimos en la salida/entrada de batería. Se han diferenciado los dos modos de funcionamiento:

- Modo Buck: carga de la batería (referencia de corriente Positiva)
- Modo Boost: descarga de la batería (referencia de corriente Negativa)

El control proporcional integral PI [14], fue implementado para el modo carga (Buck) con los valores de las constantes $K_p = 0.2$ y $K_i = 110$, y para el modo de descarga (Boost) los valores de las constantes $K_p = 0.02$ y $K_i = 3$.

A continuación, se relata el método aplicado para ajustar dichas constantes [14]:

La forma en la que se ha ajustado las constantes del controlador fue poner $K_i = 0$ e introducir un valor de $K_p < 1$ e ir comprobando la respuesta del sistema con una entrada tipo escalón, cuando se obtiene un valor muy próximo al de la referencia (debido al error en régimen permanente) se incrementa los valores de K_i hasta conseguir reducir ese error. A veces puede que haya que encontrar un equilibrio e ir cambiando los valores de ambas variables de acuerdo con la respuesta del sistema.

La forma ideal de realizar el ajuste de dichas constantes es una vez obtenida la función de transferencia ajustar el control PI acorde a ésta. Pero un sistema bidireccional como éste no es lineal, en otras palabras, la configuración circuital cambia conforme la conmutación de los dispositivos y con ello la función de transferencia. Algunos autores han introducido el concepto de circuito medio equivalente para elaborar una función de transferencia media que represente el sistema, aunque planteado para los modos buck y boost por separado.

El controlador PI funciona de la siguiente manera: La realimentación de la corriente de la batería es comparada con la referencia calculada por el HEMS, obtiene el error y éste se multiplica por las distintas

constantes K_p y K_i para calcular la acción proporcional e integral y luego son sumadas. Con estos se obtiene el duty cycle que debe estar entre 0 y 1. Finalmente se compara con una señal en diente de sierra que varía entre 0 y 1 a una frecuencia de conmutación $f_{switching} = 5\text{Khz}$ para evitar que sean muy notorias las pérdidas por conmutación de los IGBT y por calentamiento.

Para el modo boost, dado que la referencia y por tanto realimentación son negativas, se han incorporado bloques de ganancia -1 para trabajar con valores positivos de constantes K_p y K_i . También se incorporan bloques de saturación entre 0 y 0.9 ya que un duty cycle de 1 requiere que el IGBT esté conectado el 100% del tiempo lo que en la realidad puede causar daños al propio componente y al circuito en general [14].

Por último, se incorpora un bloque lógico que decide si el duty cycle calculado se debe aplicar al IGBT del modo Buck o Boost. Simplemente compara si la referencia es positiva deja pasar la señal PWM calculada por el controlador del modo Buck, en el caso contrario en el que la referencia sea negativa deja pasar la señal PWM calculada por el controlador Boost. Gracias a este bloque, en cualquier caso, no pueden estar ambos IGBT en modo ON.

- En un bloque Matlab_Function se introduce el algoritmo del HEMS.

En la figura 56 se muestra dicho bloque que posee como entradas la lectura del Consumo, Generación_PV, Precio de la electricidad, SOC, SOC_{max} , SOC_{min} , Tensión de la Batería, entre otras y como salidas incorpora distintos interruptores de control como el del inversor, de los paneles fotovoltaicos y baterías, así como la referencia de corriente para la carga/descarga de la batería que será usada por el controlador PI. Ésta referencia pasa por un bloque de saturación en el cual debemos fijar los valores máximos admisibles para las corrientes de carga y descarga de la batería.

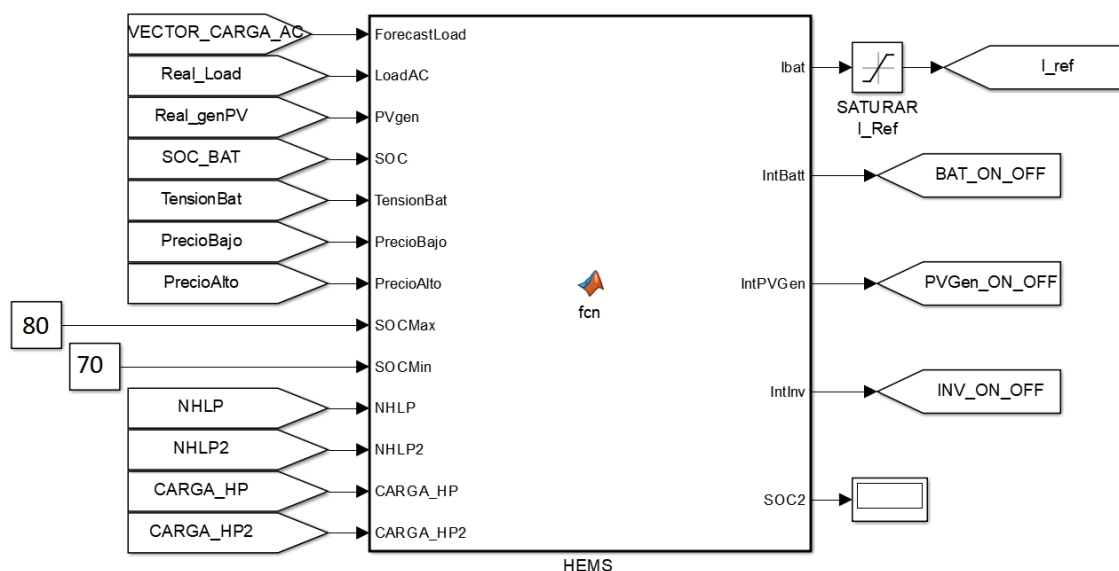


Figura 56: Simulink - Controlador HEMS



ANEXO VI. EJEMPLO DE CÓDIGO FUENTE PARA LA GENERACIÓN DE CONSIGNAS (PRECIO)

```
function y = fcn(u,time)
%%codegen                                %%vector de 24 valores
Price= [0.1353                          %%primer día, Eur/kWh
0.1337
0.1318
0.1272
0.1229
0.1204
0.1181
0.1177
0.1162
0.1156
0.1238
0.1254
0.1250
0.1210
0.1110
0.0985
0.0973
0.1056
0.1134
0.1210
0.1240
0.1303
0.1300
0.1228
]/1000;

a=time/168;                             %%tiempo de simulación /(7días*24horas)
if((u > 0)&&(u < a))                      %%mostrar el valor según la hora del día
    y = Price(1);
elseif((u >= a)&&(u < 2*a))
    y = Price(2);
elseif((u >= 2*a)&&(u < 3*a))
    y = Price(3);
elseif((u >= 3*a)&&(u < 4*a))
    y = Price(4);
elseif((u >= 4*a)&&(u < 5*a))
    y = Price(5);
elseif((u >= 5*a)&&(u < 6*a))
    y = Price(6);
elseif((u >= 6*a)&&(u < 7*a))
    y = Price(7);
elseif((u >= 7*a)&&(u < 8*a))
    y = Price(8);
elseif((u >= 8*a)&&(u < 9*a))
    y = Price(9);
elseif((u >= 9*a)&&(u < 10*a))
    y = Price(10);
elseif((u >= 10*a)&&(u < 11*a))
    y = Price(11);
elseif((u >= 11*a)&&(u < 12*a))
    y = Price(12);
elseif((u >= 12*a)&&(u < 13*a))
    y = Price(13);
elseif((u >= 13*a)&&(u < 14*a))
    y = Price(14);
elseif((u >= 14*a)&&(u < 15*a))
    y = Price(15);
elseif((u >= 15*a)&&(u < 16*a))
    y = Price(16);
elseif((u >= 16*a)&&(u < 17*a))
    y = Price(17);
elseif((u >= 17*a)&&(u < 18*a))
    y = Price(18);
elseif((u >= 18*a)&&(u < 19*a))
    y = Price(19);
elseif((u >= 19*a)&&(u < 20*a))
    y = Price(20);
elseif((u >= 20*a)&&(u < 21*a))
    y = Price(21);
elseif((u >= 21*a)&&(u < 22*a))
    y = Price(22);
elseif((u >= 22*a)&&(u < 23*a))
    y = Price(23);
elseif((u >= 23*a)&&(u < 24*a))
    y = Price(24);
else
    y = 0;
end
```



ANEXO VII. COMPARACIÓN DE GRÁFICAS A DIFERENTES TIEMPOS DE MUESTREO

A continuación, se adjuntan las siguientes gráficas en las cuales se pueden apreciar el notorio cambio al realizar la medida de la potencia activa consumida en un sistema real.

El archivo en formato Excel se ha descargado de la dirección web [15] y posee datos del consumo en W/min a lo largo de varios años, es decir:

$$60 \frac{\text{datos}}{1 \text{ hora}} \quad 24 \frac{\text{horas}}{1 \text{ día}} \quad 365 \frac{\text{días}}{1 \text{ año}} = 525600 \frac{\text{datos}}{\text{año}}$$

Por ejemplo, para este primer análisis usamos los datos del primer día (1 de enero -> consumo atípico)

Consumo minutal: En un día se han tomado 1440 medidas y se corresponde con la figura 57:

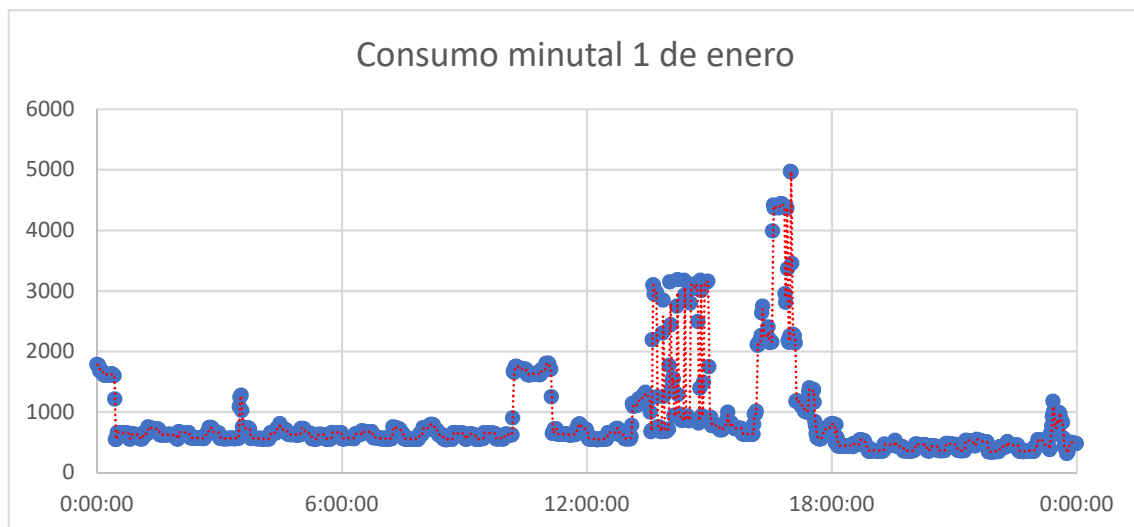


Figura 57: Consumo minutal 1 de enero

Podemos ver grandes picos de consumo rozando los 3000W - 5000W que se prolongan varios minutos y una línea de tendencia permanente en el consumo en torno a los 600 W.

Consumo 5-minutal: Ahora comprobaremos el cambio en la gráfica con respecto a la anterior en el caso de haber tomado datos cada 5 minutos, por lo tanto, en 1 día se habrán tomado 288 medidas y se corresponde con la figura 58:

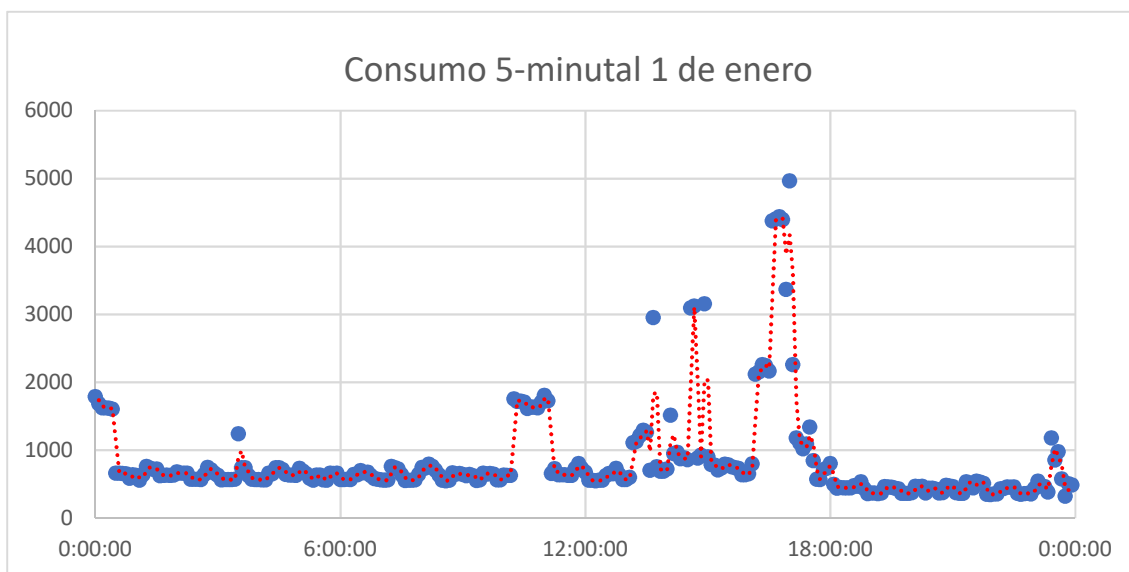


Figura 58: Consumo 5-minutal 1 de enero

Los picos de consumo se mantienen visibles, aunque con menor frecuencia, la línea de tendencia de consumo permanente se sigue manteniendo en torno a los 600 W.

Consumo 15-minutal: En el caso de realizar medidas de consumo cada 15 minutos ya empieza a ser muy notorio el cambio, ya que solo se tienen 96 medidas en 1 día y se corresponde con la figura 59:

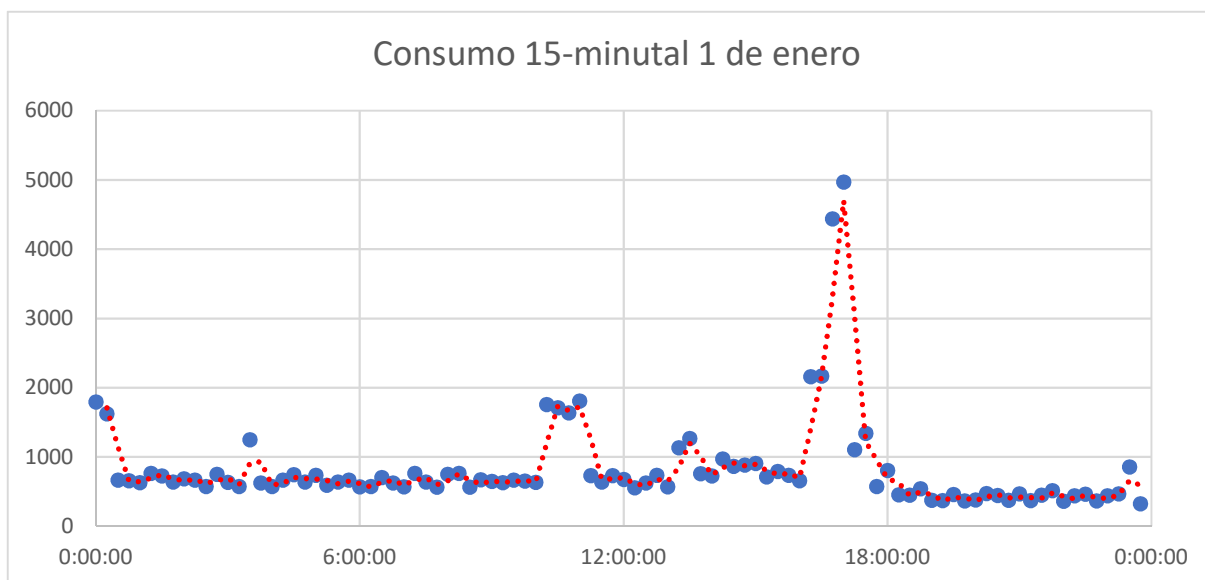


Figura 59: Consumo 15-minutal 1 de enero

Los picos de consumo casi han desaparecido, y el consumo permanente se mantiene casi intacto cerca de los 600 W.



Consumo Horario: Por último, se presenta el consumo horario que es la base de este proyecto ya que el uso los datos y resultados obtenidos se han obtenido con desglose horario. Con 24 medidas en un día se corresponde con la figura 58:

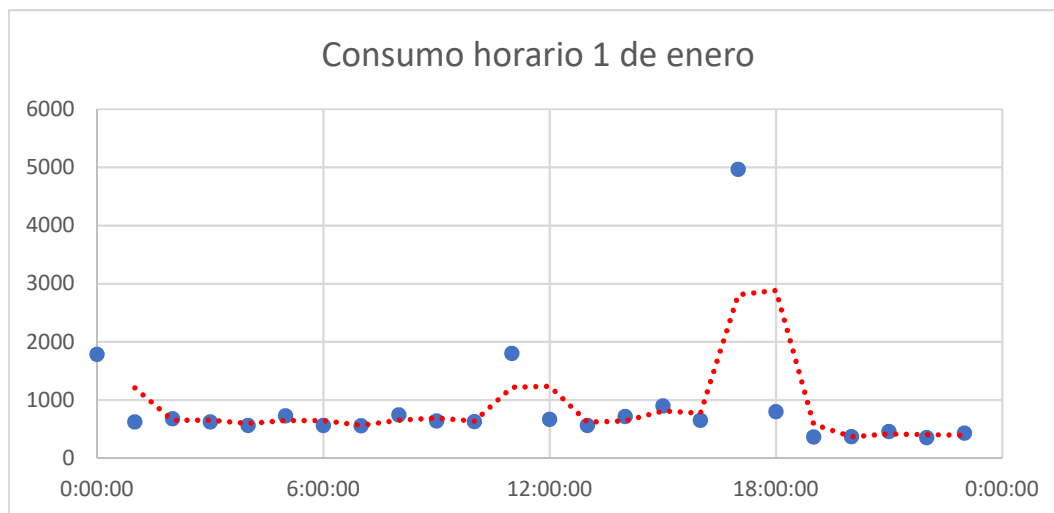


Figura 60: Consumo horario 1 de enero

Solo se mantienen los picos que coincidieron con el periodo de muestreo de 1 hora por lo que, a diferencia de la gráfica del minutario original, ésta gráfica solo posee 2 picos de consumo. El consumo permanente se ha visto poco influenciado por este periodo de muestreo.

En la siguiente tabla se muestra el resultado de sumar los consumos y su representación en Wh:

Muestreo	Consumo diario particular	Consumo diario
Minutal	1223766 Wmin	20396,1 Wh
5-minutal	240408 W5min	20034 Wh
15-minutal	79518 W15min	19879,5 Wh
Horario	21274 Wh	21274 Wh

Tabla 15: Tabla comparativa, Consumo a diferentes tiempos de muestreo - 1 de enero

Se observa que no hay grandes diferencias en cuanto al orden de magnitud de los resultados obtenidos en los distintos casos, todos ellos en torno a los 20 kWh.

Se puede extraer como conclusión general que no está muy alejado de la realidad el trabajar con valores horarios, aunque en cierta medida sería más realista trabajar con datos minutales, pero por contra es más complejo el tomar, almacenar y gestionar gran cantidad de datos.



A continuación, se realizará el mismo análisis para otro día, por ejemplo, para el 1 de abril:

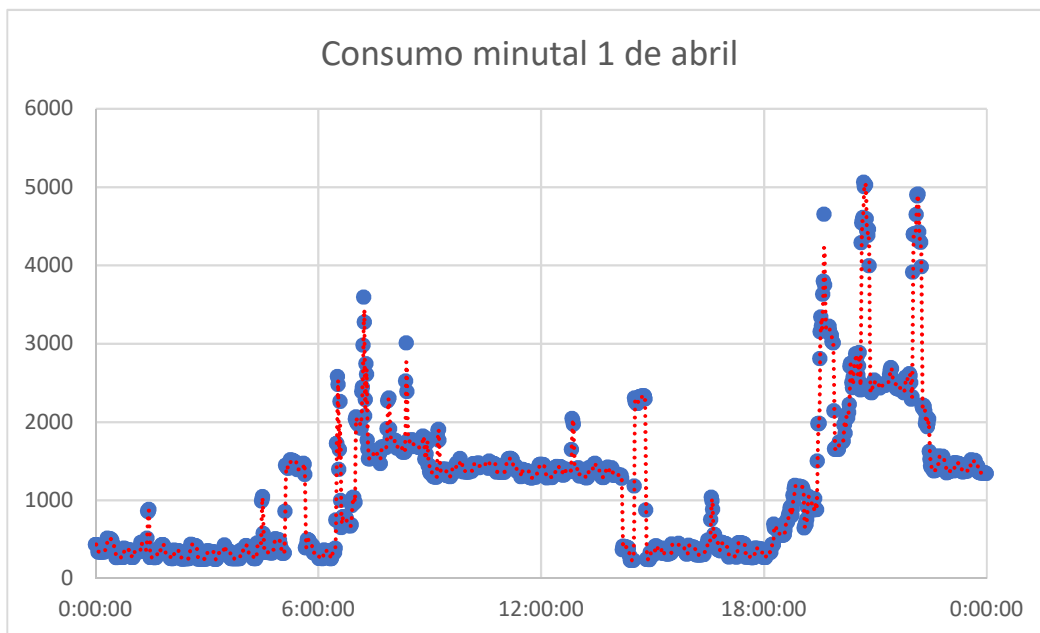


Figura 61: Consumo minutal 1 de abril

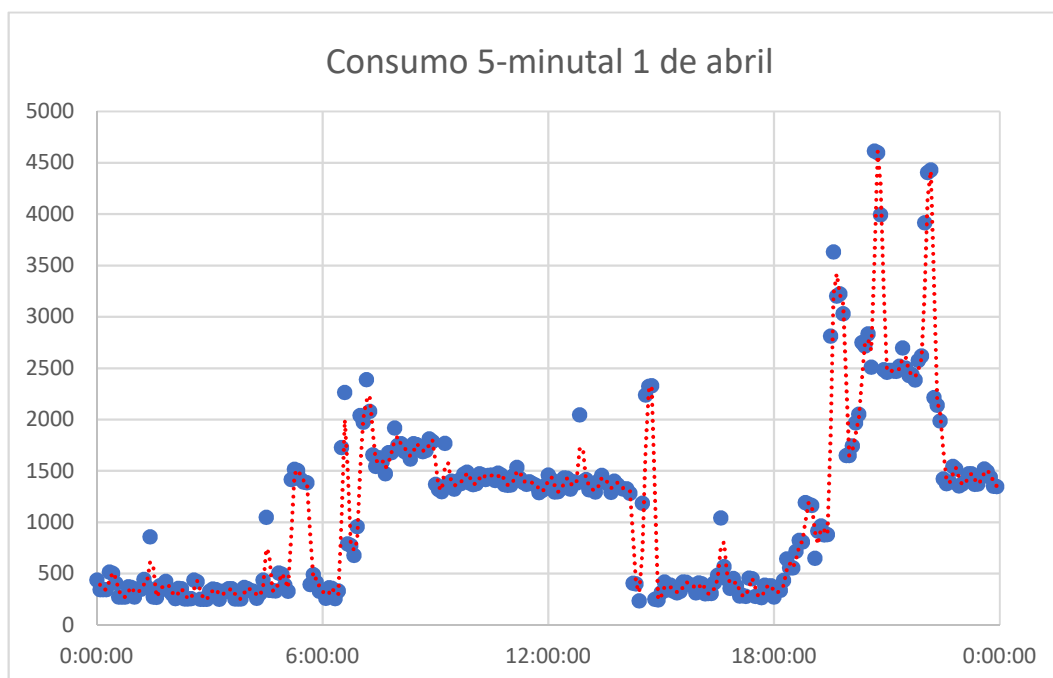


Figura 62: Consumo 5-minutal 1 de abril

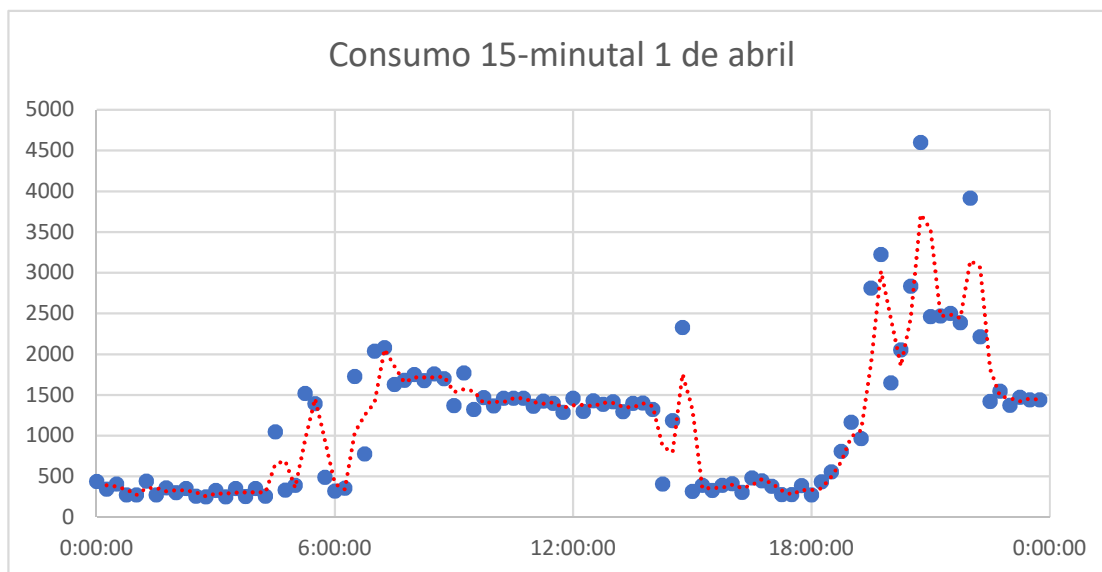


Figura 63: Consumo 15-minutal 1 de abril

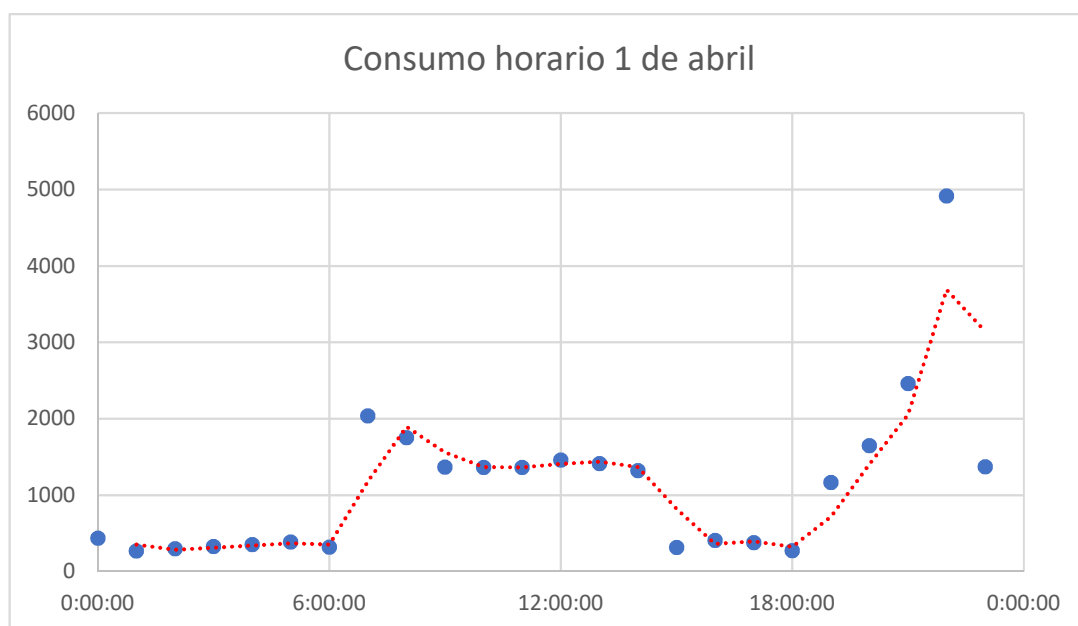


Figura 64: Consumo horario 1 de abril

Y los resultados de los consumos se muestran a continuación:

Muestreo	Consumo diario particular	Consumo diario
Minutal	1687440 Wmin	28124 Wh
5-minutal	335558 W5min	27963 Wh
15-minutal	112366 W15min	28091,5 Wh
Horario	27446 Wh	27446 Wh

Tabla 16: Tabla comparativa, Consumo a diferentes tiempos de muestreo - 1 de abril

Observamos nuevamente que todas poseen un orden de magnitud similar, en torno a los 27 kWh.

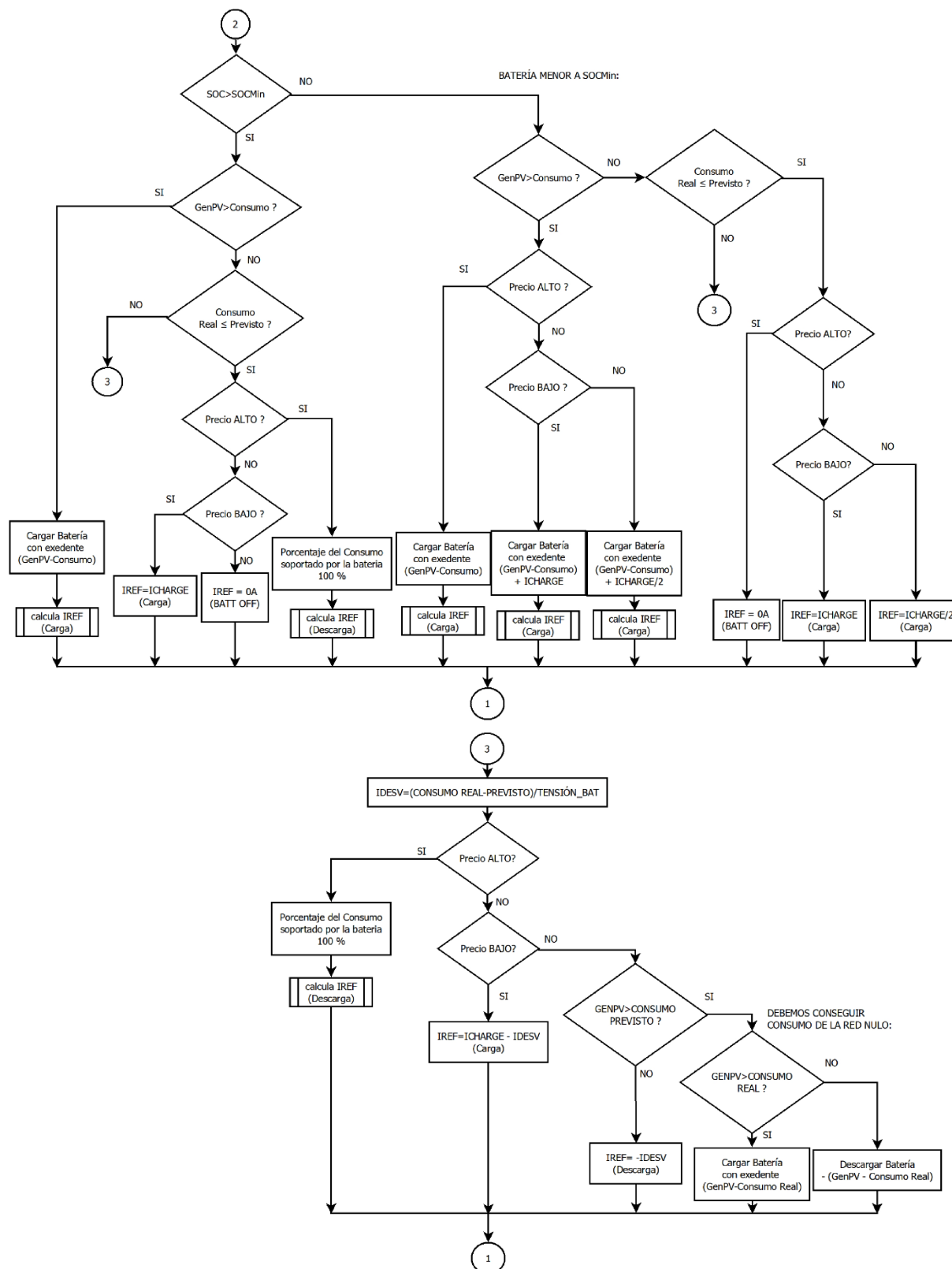


Figura 66: Diagrama de flujo caso "real" (previsión de consumo ≠ consumo real). Continuación



Código fuente implementado en el bloque *HEMS Controller*:

```
function [Ibat,IntBatt,IntPVGen,IntInv,SOC2] =  
fcn(ForecastLoad,LoadAC,PVgen,SOC,TensionBat,PrecioBajo,PrecioAlto,SOCMax,SOCMin,NHLP,NHLP2,CARGA_HP,CARGA_HP2)  
%% Valores de las salidas si no se cumple ningún caso  
IntPVGen=1;  
IntInv=1;  
Ibat=0;  
IntBatt=0;  
  
%% Calcular límite SOCAuxiliar: SOCmin + 1/4(SOCmax-SOCmin)  
SOC2=SOCmin+(1/4)*(SOCMax-SOCmin);  
  
%% Corriente óptima de carga según el número de horas precio bajo y load forecast  
%%Referencia positiva cargar batería, referencia negativa: descarga  
batería%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%55  
ICargaBat = (CARGA_HP-(IntPVGen*CARGA_HP2))/((NHLP-(IntPVGen*NHLP2))*TensionBat);%(restar potencia horas high_price y  
horas low_price)=cuando cuando GenPV mayor a Consumo  
  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%% Caso SOC mayor a SOCmax, batería cargada  
if(SOCMax<=SOC)  
if(PrecioAlto)&&(LoadAC-PVgen>0) %%precio elevado y Carga mayor a GenPV  
IntBatt=1; %%encender baterías  
Ibat=(100/100)*(-(LoadAC-PVgen))/(TensionBat);%descarga 100% de la carga consumida  
else  
if(SOC<99)  
if((LoadAC-PVgen)<0) %%generaciónPV mayor a carga  
IntBatt=1; %%encender baterías  
Ibat= -(LoadAC-PVgen)/(TensionBat);%%corriente a la batería: excedente  
else  
if(LoadAC<=ForecastLoad) %%caso ideal  
IntBatt=0;  
Ibat=0; %%apaga baterías  
else %%hemos fallado en la previsión ya que es menor al consumo real  
if(PrecioAlto) %%si precio alto  
IntBatt=1; %%encender baterías  
Ibat=(100/100)*(-(LoadAC-PVgen))/(TensionBat);%corriente de la batería: 100% de la necesaria  
elseif (PrecioBajo) %%si precio bajo  
Ibat=ICargaBat-((LoadAC-ForecastLoad)/TensionBat);  
%%corriente a la batería óptima menos variación carga real y prevista  
IntBatt=1; %%encender baterías  
else %%si precio normal  
if(PVgen>ForecastLoad) %%necesito 0  
if(LoadAC>PVgen) %%consumo real mayor a PVgen  
Ibat=-(LoadAC-PVgen)/TensionBat; %DESCARGA  
IntBatt=1; %%enciende baterías  
else %%PVgen mayor a consumo real  
Ibat= -(LoadAC-PVgen)/(TensionBat);%CARGA excedente  
IntBatt=1; %%enciende baterías  
end  
else %%necesito que la potencia sea igual que la del caso ideal  
Ibat=-(LoadAC-ForecastLoad)/TensionBat;%descarga lo equivocado  
IntBatt=1; %%enciende baterías  
end  
end  
end  
else  
IntBatt=0;  
Ibat=0; %%apaga baterías  
end  
end  
end  
  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
%% caso SOC entre SOC2 y SOCMax  
if(SOCMax>SOC)&&(SOC>=SOC2)  
if((LoadAC-PVgen)<0) %%generaciónPV mayor a carga  
IntBatt=1; %%encender baterías  
Ibat= -(LoadAC-PVgen)/(TensionBat);%corriente a la batería: excedente  
else %%carga mayor a GenPV  
if(LoadAC<=ForecastLoad) %%caso ideal  
if(PrecioAlto) %%si precio alto  
IntBatt=1; %%encender baterías  
Ibat=(100/100)*(-(LoadAC-PVgen))/(TensionBat);%corriente de la batería: 100% de la necesaria  
elseif (PrecioBajo) %%si precio bajo  
Ibat=ICargaBat;  
IntBatt=1; %%encender baterías  
else %%si precio normal  
Ibat=0;  
IntBatt=0; %%apaga baterías  
end  
else %%hemos fallado en la previsión ya que es menor al consumo real
```



```
if(PrecioAlto)                %%si precio alto
    IntBatt=1;                %%encender baterías
    Ibat=(100/100)*(-(LoadAC-PVgen))/(TensionBat);%%corriente de la batería: 100% de la necesaria

elseif (PrecioBajo)           %%si precio bajo
    Ibat=ICargaBat-((LoadAC-ForecastLoad)/TensionBat);
    %%corriente a la batería óptima menos variación carga real y prevista
    IntBatt=1;                %%encender baterías

else                           %%si precio normal
    if(PVgen>ForecastLoad)
        if(LoadAC>PVgen)
            Ibat=-(LoadAC-PVgen)/TensionBat; %DESCARGA
            IntBatt=1;
        else
            Ibat= -(LoadAC-PVgen)/(TensionBat);%CARGA excedente
            IntBatt=1;
        end
    else
        Ibat=-(LoadAC-ForecastLoad)/TensionBat;%%descarga lo equivocado
        IntBatt=1;
    end
end
end
end

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% caso SOC entre SOC2 y SOCmin
if(SOC2>SOC)&&(SOC>=SOCmin)
    if((LoadAC-PVgen)<0)        %%generaciónPV mayor a carga
        IntBatt=1;            %%encender baterías
        Ibat= -(LoadAC-PVgen)/(TensionBat);%%corriente a la batería: excedente
    else
        %%carga mayor a genPV
        if(LoadAC<=ForecastLoad) %%caso ideal
            if(PrecioAlto)        %%si precio alto
                IntBatt=1;        %%encender baterías
                Ibat=(100/100)*(-(LoadAC-PVgen))/(TensionBat);%%corriente de la batería 100% de la necesaria

            elseif (PrecioBajo)    %%si precio bajo
                IntBatt=1;        %%encender baterías
                Ibat=ICargaBat;    %%corriente a la batería óptima menos variación carga real y prevista

            else
                %%si precio normal
                Ibat=0;
                IntBatt=0;
            end
        else
            %%hemos fallado en la previsión ya que es menor al consumo real
            if(PrecioAlto)        %%si precio alto
                IntBatt=1;        %%encender baterías
                Ibat=(100/100)*(-(LoadAC-PVgen))/(TensionBat);%%corriente de la batería: 100% de la necesaria

            elseif (PrecioBajo)    %%si precio bajo
                Ibat=ICargaBat-((LoadAC-ForecastLoad)/TensionBat);
                %%corriente a la batería óptima menos variación carga real y prevista
                IntBatt=1;        %%encender baterías

            else
                %%si precio normal
                if(PVgen>ForecastLoad)
                    if(LoadAC>PVgen)
                        Ibat=-(LoadAC-PVgen)/TensionBat; %DESCARGA
                        IntBatt=1;
                    else
                        Ibat= -(LoadAC-PVgen)/(TensionBat);%CARGA excedente
                        IntBatt=1;
                    end
                else
                    Ibat=-(LoadAC-ForecastLoad)/TensionBat;%%descarga lo equivocado
                    IntBatt=1;
                end
            end
        end
    end
end
end

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% caso SOC menor a SOCmin, batería descargada
if(SOCmin>SOC)
    if((LoadAC-PVgen)<0)        %%generaciónPV mayor a carga

        if(PrecioAlto)        %%si precio alto
            IntBatt=1;        %%encender baterías
            Ibat= -(LoadAC-PVgen)/(TensionBat);%%corriente a la batería: excedente

        elseif (PrecioBajo)    %%si precio bajo
            Ibat=ICargaBat+(-(LoadAC-PVgen))/(TensionBat);%%corriente a la batería ex +carga óptima
            IntBatt=1;        %%encender baterías

        else
            %%si precio normal
        end
    end
end
```

```
Ibat=(ICargaBat/2)+(-(LoadAC-PVgen))/(TensionBat);%corriente a la batería ex +carga opt a medio gas
IntBatt=1; %encender baterías

else
end

    %%carga mayor a genPV
    if(LoadAC<=ForecastLoad) %caso ideal
        if(PrecioAlto) %si precio alto
            IntBatt=0; %apaga batería
            Ibat=0;

        elseif (PrecioBajo) %si precio bajo
            Ibat=ICargaBat; %corriente a la batería óptima
            IntBatt=1; %encender baterías

        else %si precio normal
            Ibat=(ICargaBat/2); %corriente a la batería óptima a medio gas
            IntBatt=1; %encender baterías
        end
    else %hemos fallado en la previsión ya que es menor al consumo real
        if(PrecioAlto) %si precio alto
            IntBatt=1; %encender baterías
            Ibat=(100/100)*(-(LoadAC-PVgen))/(TensionBat);%corriente de la batería: 100% de la necesaria

        elseif (PrecioBajo) %si precio bajo
            Ibat=ICargaBat-((LoadAC-ForecastLoad)/TensionBat);
            %corriente a la batería óptima menos variación carga real y prevista
            IntBatt=1; %encender baterías

        else %si precio normal
            if(PVgen>ForecastLoad)
                if(LoadAC>PVgen)
                    Ibat=-((LoadAC-PVgen)/TensionBat);%DESCARGA
                    IntBatt=1;

                else
                    Ibat= -((LoadAC-PVgen))/(TensionBat);%CARGA excedente
                    IntBatt=1;

                end
            else
                Ibat=-((LoadAC-ForecastLoad))/TensionBat;%descarga lo equivocado
                IntBatt=1;

            end
        end
    end
end

end
end
```




ANEXO IX. RESULTADOS OBTENIDOS PARA EL CASO C

A continuación, en la figura 67 se muestra el vector de precios generado para los 7 días de la semana con los niveles de precio alto y precio bajo y la señal obtenida al cruzar dichos límites:

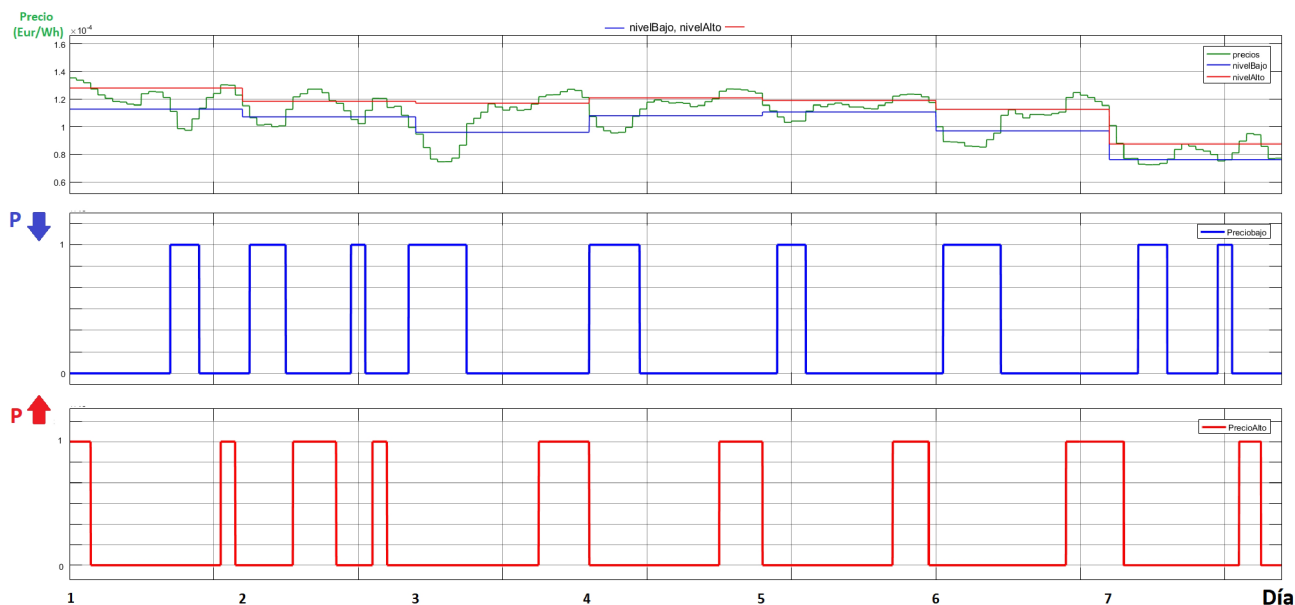


Figura 67: 1: Vector de precios para los 7 días

2. -Franjas horarias de precio bajo 3.- Franjas horarias de precio elevado

La figura 67 se muestran los niveles del precio de la electricidad alto y bajo, éstos se calculan para cada día realizando la media aritmética del vector de precios de todo el día $\pm 20\%$ para el precio alto y bajo respectivamente. Estos pulsos serán usados por el controlador para saber si el precio de actual de la electricidad se encuentra en niveles mínimos o máximos.

En este caso no se ha conectado la generación fotovoltaica por lo que el algoritmo intentará almacenar la energía necesaria que se consumirá en las horas con precio elevado.

En la figura 68 se muestran los resultados obtenidos en la simulación durante los 7 días, en este orden:

- SOC de la batería (%)
- Pulsos Precio alto y Precio Bajo (0 - 1)
- Corriente de la batería (Carga: $I_{Ref} > 0$; Descarga: $I_{Ref} < 0$)
- Potencia del inversor (W) (Potencia cedida > 0 ; Potencia consumida < 0)

La corriente calculada por el algoritmo será la óptima que necesite el sistema para cargar el ESS para luego abastecer el consumo en periodos de precio elevado (ya que se toma en cuenta la previsión del consumo), así evitaremos almacenar más energía de la necesaria. Es por ello, que los niveles del estado de carga SOC se mantendrán en valores muy similares al inicio y al final de cada día.



ANEXOS

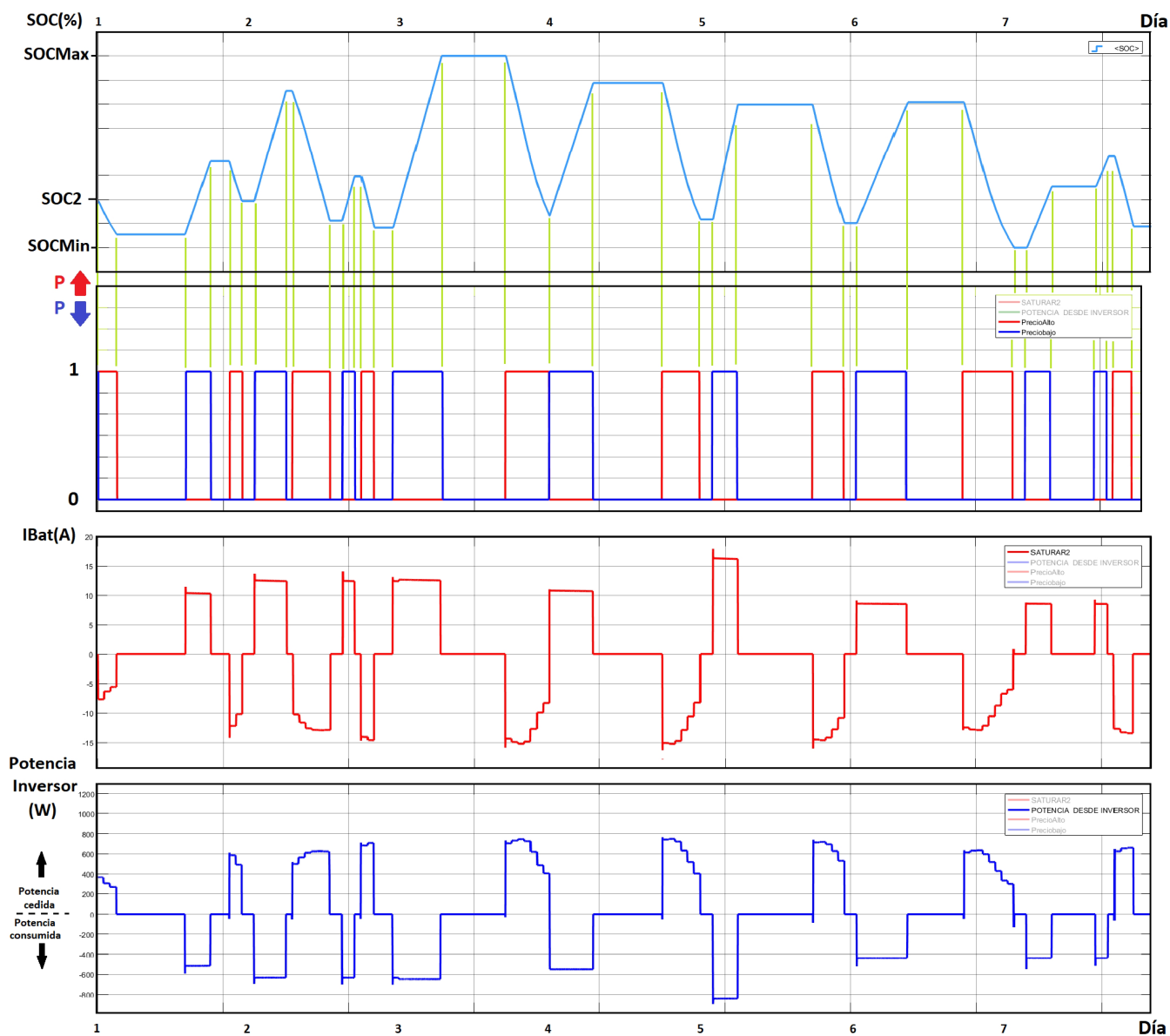


Figura 68: Resultados obtenidos caso C

- 1.- Estado de carga (SOC); 2.-Franjas horarias de precio elevado y bajo
- 3.-Corriente de la Batería; 4.-Potencia del inversor de la batería

La gráfica del consumo de potencia asumida por la red eléctrica se muestra en el caso C del Capítulo IV.



ANEXO X. RESULTADOS OBTENIDOS PARA EL CASO D

Este es el caso en el que el algoritmo propuesto se implica a fondo ya que se encuentran activados todos los elementos del sistema. En la figura 69, se muestran los resultados obtenidos en la simulación durante los 7 días, en este orden:

- Consumo (de la Carga_AC) y Generación PV (W)
- SOC de la batería (%)
- Pulsos Precio alto y Precio Bajo (0 - 1)
- Corriente de la batería (Carga: $I_{Ref} > 0$; Descarga: $I_{Ref} < 0$)
- Potencia del inversor (W) (Potencia cedida > 0 ; Potencia consumida < 0)

Se observa que siempre que la Generación PV supera el consumo esa potencia extra se almacena en la batería, por el contrario ($Gen_{PV} < Consumo$) en periodos de precio bajo se compra y almacena Energía eléctrica y en periodos de precio alto se cede esa energía para reducir al máximo el consumo desde la red eléctrica.

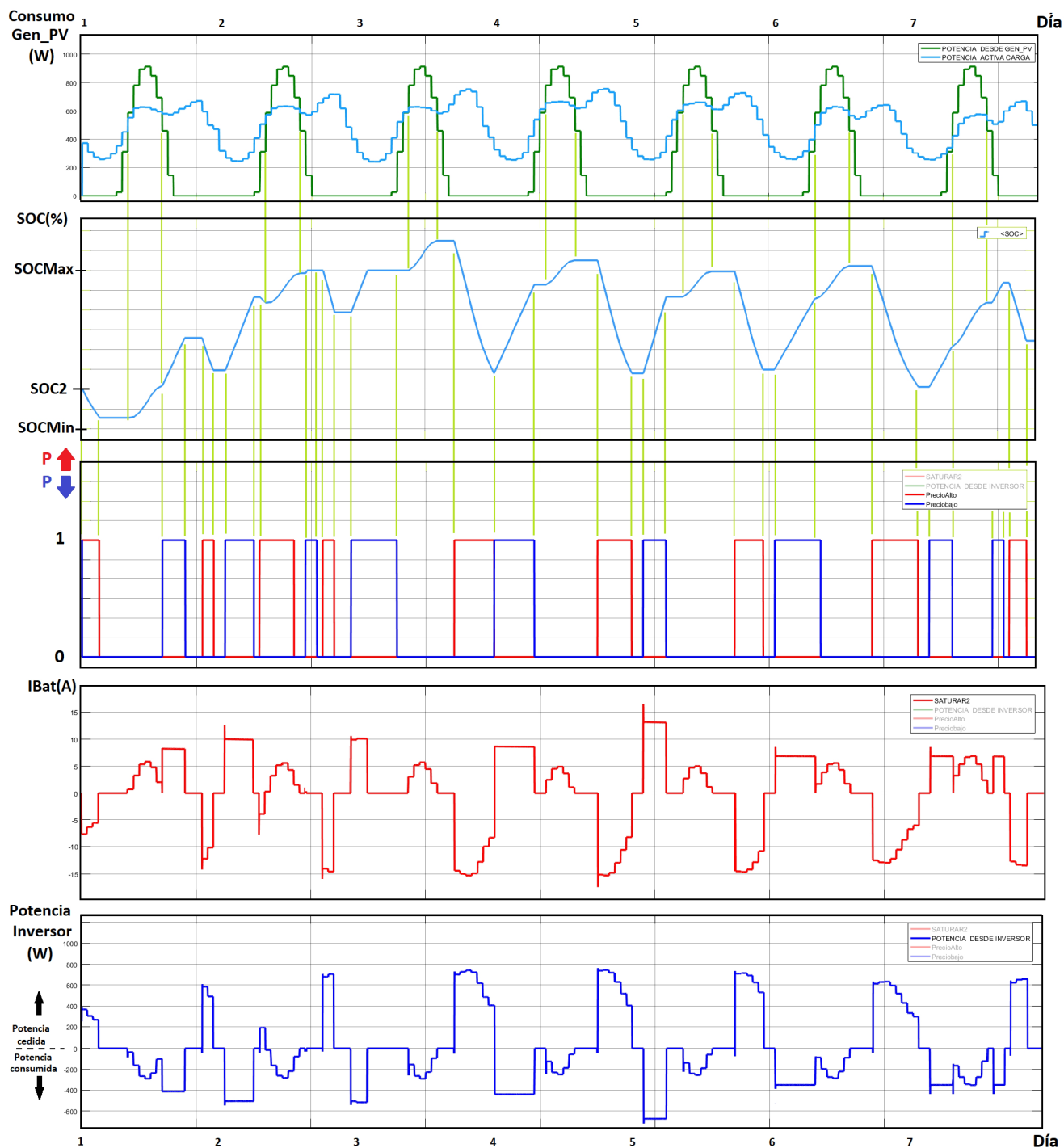


Figura 69: Resultados obtenidos caso D

1.- Gen PV y Consumo 2.- Estado de carga (SOC); 3.-Franjas horarias de precio elevado y bajo
4.-Corriente de la Batería; 5.-Potencia del inversor de la batería

La gráfica del consumo de potencia asumida por la red eléctrica se muestra en el caso D del Capítulo IV.