Hydrogen consumption prediction of a fuel cell based system with a hybrid intelligent approach

Juan Aurelio Montero-Sousa[a], Héctor Aláiz-Moretón[b], Héctor Quintián[a,*],
Tomás González-Ayuso[c], Paulo Novais[d], José Luis Calvo-Rolle[a]

[a]*University of A Coruña,*
Department of Industrial Engineering,
Avda. 19 de febrero s/n, 15405, Ferrol, A Coruña, Spain
[b]*Universidad de León,*
Department of Electrical and Systems Engineering,
Campus de Vegazana, 24071 León, Spain
[c]*CIEMAT,*
Departamento de Energía,
Av. Complutense 40, 28040 Madrid, Spain
[d]*University of Minho,*
Department of Informatics/Algoritmi Center,
Minho, Braga, Portugal

Abstract

Energy storage is one of the challenges of the electric sector. There are several different technologies available for facing it, from the traditional ones to the most advanced. With the current trend, it is mandatory to develop new energy storage systems that allow optimal efficiency, something that does not happen with traditional ones. Another feature that new systems must meet is to envisage the behavior of energy generation and consumption. With this aim, the present research deals the hydrogen consumption prediction of a fuel cell based system thanks a hybrid intelligent approach implementation. The work is based on a real testing plant. Two steps have been followed to create a hybrid model. First, the real dataset has been divided into groups whose elements have similar characteristics. The second step, carry out the

[*]Corresponding author

Email addresses: juan.montero.sousa@udc.es (Juan Aurelio Montero-Sousa), hector.moreton@unileon.es (Héctor Aláiz-Moretón), hector.quintian@udc.es (Héctor Quintián), tomas.gonzalez@ciemat.es (Tomás González-Ayuso), pjon@di.uminho.pt (Paulo Novais), jlcalvo@udc.es (José Luis Calvo-Rolle)

regression using different techniques. Very satisfactory results have been achieved during the validation of the model.

1. Introduction

The country's electric sector is a very complex system for a number of diverse reasons [1]. The most important feature that makes it so difficult is ensuring that energy generation must be equal to that of energy demand. The different energy generation technologies with their own characteristics, also make the system more complicated [2]. Even, things like the increase in energy plants based on renewable energies have contributed to destabilizing the system [3] [4].
Furthermore, when renewable energies or other possible sources are included in all types of buildings connected to a grid, even if the final use is not to generate energy, the network management becomes much more difficult. For this reason, it is necessary to create tools to ensure the right handling of the different energy generation and consumption points.
Therefore, the Smart-grid concept [5] [6], comes about for all of the aforementioned reasons, where it is necessary, at least, to measure the generation, the consumption, and of course, try to predict both of them, with the aim of making decisions, and then to make the overall system more efficient in every way. In any case, it is a difficult task to match the generation with the demand, which makes energy storage a desirable option in this case [7].
For all of the aforementioned reasons, it is justified that the energy storage systems are desirable in all of the different cases when working with electric power [8]. There are many technologies available for this purpose, even though some of them are relatively old and inefficient, such as pumping water into a dam [9]. However, because of this great need under the current scenario, in recent years there have been many research and development projects and works carried out [10] [11]. The electric car is one of the reasons why electric energy storage systems are being developed so quickly [12].
Of all the technologies, researchers are working especially on two of them, battery technologies and fuel cells [13]. In the case of fuel cells, which is what this work focuses on, there are also many proposals in terms of the primary fuels [14]. For numerous reasons, hydrogen was one of the first fuels

3

used. This gas is very easy to obtain through a simple hydrolysis process, and despite its storage not being as easy as for other alternatives, in general terms, this is one of the most feasible alternatives for this kind of purpose [15].

As has been commented on previously, given the current trend in the electric sector, and with the aim of optimizing the overall efficiency of the grids, it is essential to have a trustworthy forecast to make the right decisions [16]. This prediction will contribute, for instance, to purchasing or selling energy at the best price when necessary, store energy when it is more convenient, and so on. The fuel cell systems based on hydrogen and hydrogen generation itself for storage purposes, obviously needs an effective prediction.

During the modeling process, it is possible to take several different alternatives into account although the performance could be different. One of the most used techniques based on traditional regression models is the MRA (Multiple Regression Analysis) [16] [17]. There are several applications in which this method or others with small variations are used [17] [18] [19]. However the MRA technique has some limitations in several cases [20] [16] [21]. One of the reasons why its performance is not suitable is the non-linearity of the problem to be modeled. Intelligent systems are used in some different applications with very satisfactory performance in general terms [22] [23] [24] [25] [26]. Of course the non-linearity problem could be solved in many cases with the use of soft-computing techniques [27] [28] [29] [30]. The problem when the system is non-linear could be the same, even by using simple intelligent systems. When it occurs, it is possible to divide the problem using clustering techniques such as K-means [31] [32] [33] [34] [35].

In this work, the performance of a fuel cell is modeled based on hydrogen prediction, from several variables measured at a real storage system plant. The proposals deal with the method for a model creation, with the aim of predicting the amount of hydrogen consumed by the fuel cell. Two datasets from different real fuel cell-based systems were used.

Due to its non-linearity, clustering has been achieved before the regression step. To do so, a novel method based on the novel neural projection method for dimensional reduction, known as Beta Hebbian Learning (BHL), is applied to visually analyze the internal structure of the dataset for clustering purposes. The main advantage is that it is possible to take human expert knowledge into account during the creation of the groups and finally generate more simple and accurate models than by using just one general model for the whole dataset. After that, some intelligent regression techniques have

4

been carried out.

The present research work is organized as follows. After the introduction, the case being studied is explained. Then, the next section describes the proposed model approach. After that the materials and the methods taken into account for the present research are detailed. The work is continued with the experiments carried out and their results. Finally, conclusions and future works are presented.

2. Case being studied

The case study consists of an experimental system located in the "Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas" (CIEMAT) in Spain. In this specific case, the system is used to store the energy generated through two renewable sources: wind and solar. A schematic of the system is shown in Figure ??, in which the local power bus is exposed to the generation and the consumption connections of the storage system.
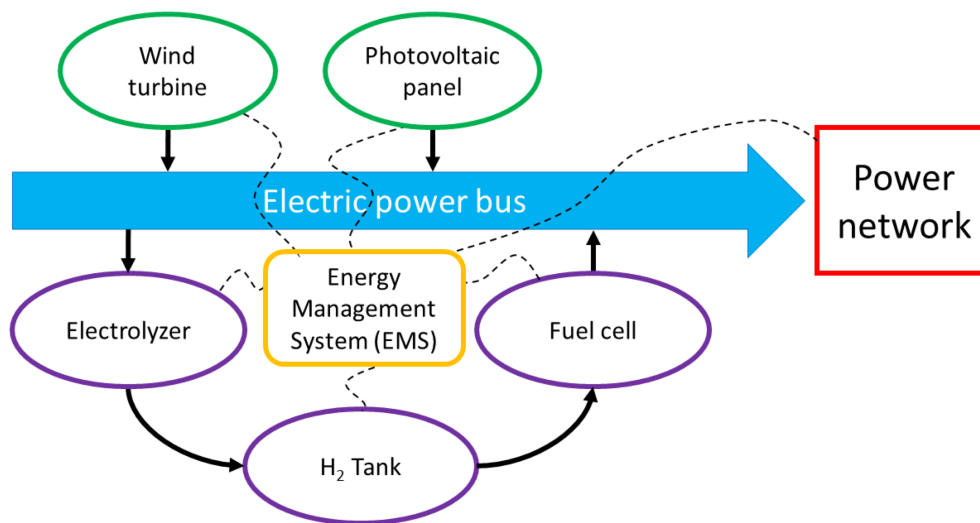

Figure 1: Energy Management System (EMS)

Both, solar and wind generation, and the fuel cell are the power input points to the power bus, while the hydrogen generation equipment is the only
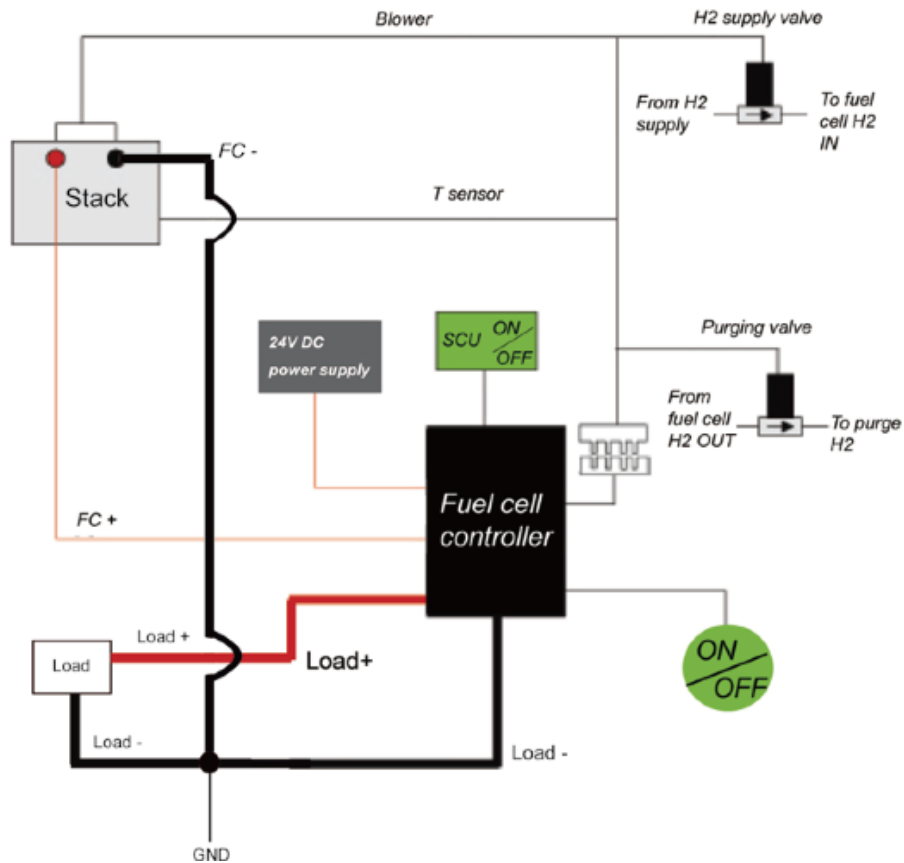
Figure 2: System configuration

consumption point shown. The energy storage system, in this case, consists of an electrolyzer, a hydrogen tank and a fuel cell. The Energy Management System decides on the operation of the whole system: how much energy is provided to the power network or to the storage system. In the periods of high generation and low demand, it would be better to store the energy so it can be used when it is necessary.

2.1. Storage System analysis

The energy storage system is analyzed to calculate the overall performance of the energy storage system.

2.1.1. Electrolyzer

The electrolyzer produces the hydrogen that feeds the fuel cell, which is in charge of producing the electric power when it is necessary.

The equipment uses H_2O; firstly, the water goes through a deionizer. The deionized water is mixed with potassium hydroxide - KOH (around 20-30%), creating the electrolyte. After that, by applying an electric current between the negative and positive electrodes in the electrolyte, the H_2 and O_2 are obtained. The operating temperature should be between 70-80°C. The O_2, in this case, is not stored and it is emitted directly to the atmosphere. The H_2 is filtered to eliminate the O_2 that may remain mixed. The hydrogen is stored in a tank for later use in the fuel cell. If the pressure increases, a safety valve opens to reduce the pressure.

To increase the efficiency, the voltage should be slightly above 1.48V, depending on the flow rate of H_2. Nitrogen is used in the electrolyzer to compensate the operating pressures.
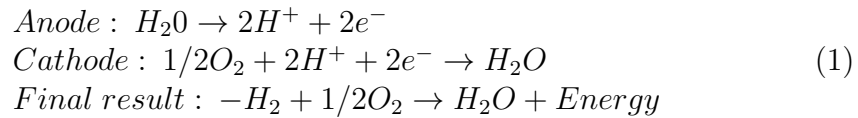
2.1.2. H_2 Tank (compressor)

Hydrogen is stored in tanks at 10 bar of pressure. The size of the tanks depends exclusively on the amount of energy to be stored; the more hydrogen stored, the more electrical energy can be produced.

As the system is experimental, the size of the deposit is not relevant, but it needs a compressor to store the hydrogen at the specific pressure. This compressor uses electrical energy so, it has to be taken into account incalculating the overall performance.

2.1.3. Fuel cell

The fuel cell generates energy from H_2, which is obtained in the electrolyzer. The energy obtained is represented by the chemical reaction of the equation 1.

$$
\begin{aligned}
&Anode: \ H_2 0 \rightarrow 2H^+ + 2e^- \\
&Cathode: \ 1/2O_2 + 2H^+ + 2e^- \rightarrow H_2O \\
&Final \ result: \ -H_2 + 1/2O_2 \rightarrow H_2O + Energy
\end{aligned}
\tag{1}
$$

This reaction occurs in each cell. If some cells are connected in series, the accumulated voltage is the sum of the individual voltages of each cell. The reaction, represented in equation 1, takes place when each cell is filled with
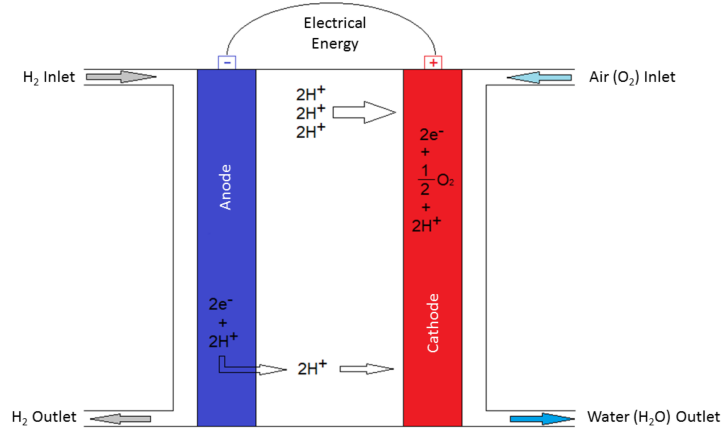
Figure 3: Schematic of a fuel cell

H_2 and O_2 (obtained from the air). A simplified scheme of the fuel cell is shown in Figure 3.

The anode is fed with hydrogen, where a catalyst accelerates the reaction, as shown in equation 1. The H_2 releases an electron of each atom and a molecule of $2H^+$ is created. This molecule will react with $1/2$ O_2 from the air that accesses the cathode. The result of this reaction is a water molecule.

For each molecule of water formed, two electrons, which provide the electrical energy are released. These reactions occur as long as H2 is supplied. Figure 4 shows the real storage system.

3. Followed steps and methods

In this section the steps followed for modeling the behaviour of the fuel cells is presented. The model used consists of a 3 step system:

- First, the dataset is procesed, outliers and wrong values are removed and the final dataset is normalized.

- Secondly, a clustering technique (the Beta Hebbian Learning algorithm) is applied to identify the internal structure of the dataset and the possible clusters.

- Finally several regression techniques are applied to the samples of each cluster in order to get the most suitable model for each cluster.
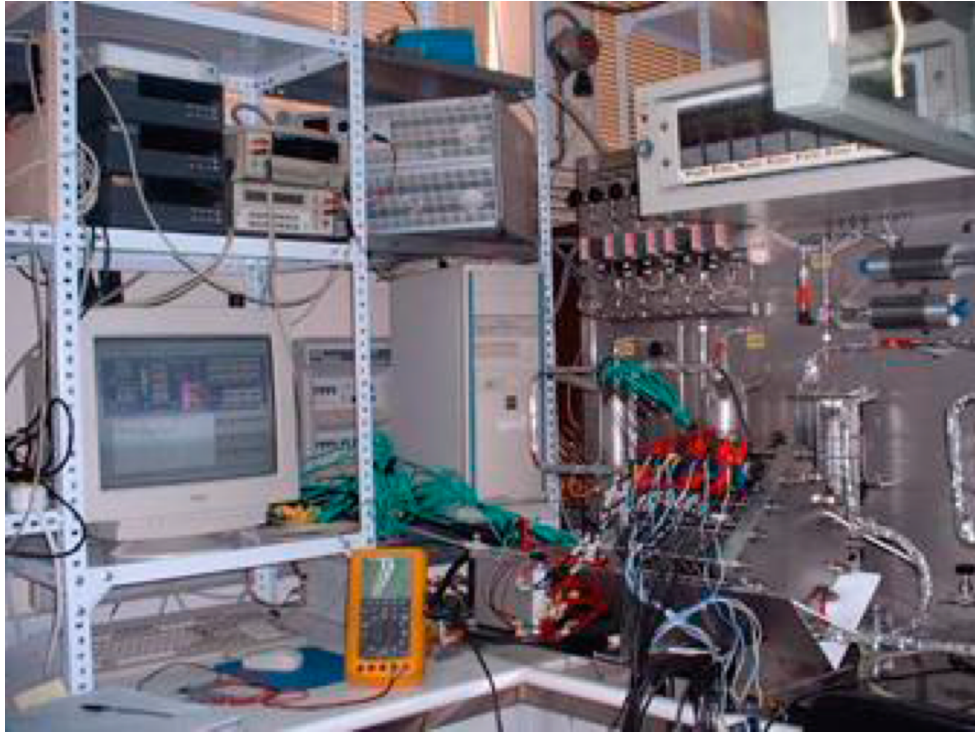
8

Figure 4: Real storage system

3.1. Preprocessing

Before using the collected dataset, a preprocessing step is applied, consisting of removing outliers and wrong data samples (for instance negative values), to finally apply a normalization of the dataset. The normalization criterion is the Maxmin Scaler [36], presented in Eq.(2):

$$\frac{X_i - min(x)}{max(x) - min(x)} \tag{2}$$

The main goal of this normalization step is to avoid the very early convergence in the first iterations, when the training process of a particular regression method begins [37].

3.2. Clustering: Beta Hebbian Learning

The Beta Hebbian Learning technique (BHL) [38] is an unsupervised neural network from the family of EPP that employs the Beta distribution

9

to update its learning rule and fit the Probability Density Function (PDF) of the residual with the distribution of a given dataset.

Thus, if the PDF of the residuals is known, the optimal cost function can be determined. By using $B(\alpha, \beta)$ parameters of the Beta distribution, the residual (e) can be drawn with the following PDF (Eq. 3):

$$p(e) = e^{\alpha-1}(1 - e)^{\beta-1} = (x - Wy)^{\alpha-1}(1 - x + Wy)^{\beta-1} \qquad (3)$$

Where α and β are used to adjust the shape of the PDF of the Beta distribution, x is the input of the network, e is the residual, W is the weight matrix, and y is the output of the network.

Then, using Eq. 4, gradient descent is performed to maximize the likelihood of the weights:

$$\frac{\partial p}{\partial W} = (e_j^{\alpha-2}(1 - e_j)^{\beta-2}(-(\alpha - 1)(1 - e_j) + e_j(\beta - 1))) =$$
$$(e_j^{\alpha-2}(1 - e_j)^{\beta-2}(1 - \alpha + e_j(\alpha + \beta - 2))) \qquad (4)$$

In the case of BHL, the learning rule allows the PDF of the residual to be fitted by maximizing the likelihood of this residual with the current distribution.

Therefore, the neural architecture for BHL is defined as follows:

$$Feedforward: y_i = \sum_{j=1}^{N} W_{ij}x_j, \forall i \qquad (5)$$

$$Feedback: e_j = x_j - \sum_{i=1}^{M} W_{ij}y_i \qquad (6)$$

$$Weights\,update: \Delta W_{ij} = \eta(e_j^{\alpha-2}(1 - e_j)^{\beta-2}(1 - \alpha + e_j(\alpha + \beta - 2)))y_i \qquad (7)$$

3.3. Regression techniques

Once the BHL has identified the different clusters present in the dataset, three well-known regression techniques are applied for each cluster to model the behaviour of the fuel cell.

3.3.1. Support Vector Machines

Support Vector Machines (SVMs) are a machine learning technique very popular for classification, regression and data analysis. SVMs work by constructing hyperplanes in high or infinite dimensional spaces, where it is possible to consider a good separation if a hyperplane has the largest distance to the nearest training data point of any class. Thus, the larger the margin, the lower the error. The SVC method (Support Vector Classifier) can be extended to solve regression problems. In contrast to SVC, Support Vector Regression (SVR) [39] depends only on a subset of the training data, because the cost function ignores any training data close to the model prediction in order to build the model. The SVR solves the following primal problem, given x training vectors and a vector $y \; \epsilon \; \{-1, 1\}^n$:

$$\min_{\omega, b, \zeta, \zeta^*} \frac{1}{2} \omega^T \omega + C \sum_{i=1}^{n} (\zeta_i + \zeta_i^*)$$
$$\text{subject to } y_i - \omega^T \phi(x_i) - b \leq \varepsilon + \zeta_i,$$
$$\omega^T \phi(x_i) + b - y_i \leq \varepsilon + \zeta_i^*,$$
$$\zeta_i, \zeta_i^* \geq 0, i = 1, ..., n$$

The parameters of Support Vector Regression are detailed below:

- Kernel function: this is the similarity function chosen for the SVR algorithm.

- C: controls the margin and misclassification. Higher C is equivalent to a small margin, and lower C is equivalent to a large margin.

- Gamma: defines how far the influence of a single training example reaches. A higher value of gamma means more influence of a training example. The 'Linear' kernel ignores this parameter.

- Degree: degree of the polynomial kernel. The other kernels do not use this hyper parameter.

3.3.2. Extra Trees Regressor

This regression technique is an ensemble method based on a meta estimator, that fits a number of randomized decision trees working with a set

of sub-samples of the dataset, to improve accuracy and control over-fitting issues later, thanks to an averaging approach [40] [41]. Therefore, this technique is able to offer the importance of a feature for establishing a feature ranking of the regression model. The parameters of the Extra Trees Regressor (ETR) are described below:

- Number of estimators: number of trees in the forest.

- Bootstrap: the number of samples used in order to build the trees.

- Max depth: maximum depth of the tree.

- Min samples leaf: minimum number of samples that are required to be at a leaf node.

- Min samples split: minimum number of samples required for splitting the internal node.

- Max features: number of features for the best split.

3.3.3. Multi-Layer Perceptron

This is a supervised learning Artificial Neural Network that is able to learn thanks to function: $Fun(\cdot) : R^n \rightarrow R0$. In this work, MLP has been implemented for regression with Python Scikit-Learn [42]. This has been trained using back-propagation containing the output layer; the linear activation function. The implemented MLP has the following parameters:

- Early Stopping: setups as True with the aim of finishing training stage sooner if the validation score is not improving.

- Learning rate: rate schedule for weight updates.

- Learning rate init: indicates how the learning rate is utilized initially. It is able to control step size when the weights are updated.

- Hidden layer sizes: number of neurons in the hidden layer.

- Nesterovs momentum: option for using Nesterov's momentum [43]. It defines how the GD (gradient descend) momentum can be implemented in two stages. First stage: make a significant jump in the direction which matches the direction of the previously calculated accumulated gradient. Second stage: make a measure of the gradient where it ended up before making the pertinent correction.

- Activation: activation function included in the hidden layer.

- Solver: defines how weight optimization is managed, with the following three options. 'LBFGS', oriented to use the quasi-Newton method family algorithm [44]. 'SGD' for using the stochastic gradient descent algorithm. [45]. And 'ADAM', oriented to the stochastic gradient-based optimizer algorithm [46]

- Warm start: when its value is True it implements a reusing of the solution made in the previous call, for fitting as initialization.

- Batch size: mini-batches for stochastic optimizer size.

3.3.4. Tuning hyper-parameters techniques
. With the aim of encapsulating the transformations in the raw input data, the *Pipeline* tool from *Python Scikit-Learn* is used [47]. This technique makes it easier to define a set of steps for pre-processing the raw dataset and carry out the training and validation test in the validation model step.

Pipeline is a Scikit-Learn tool for carrying several steps, these being, handled together within a *Ten-fold Grid Search Cross Validation* process. In this way, the cross validation is able to train the algorithms explained previously in 3.3 with different parameters, choosing a combination of the best ones when wishing to obtain the best regression model [48].

The set of parameters combined and tested in each iteration of Grid Search Cross Validation are called hyper-parameters [49].

3.3.5. Error metrics
The following, error metrics for evaluating the quality of the models are explained:

- Mean Absolute Error (MAE). This metric is oriented to measure the difference between the real value and the predicted value. This error measurement has several advantages over other error measurement [50], equation 8.

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |Y_i - \hat{Y}_i| \tag{8}$$

where Y_i is the real value and $\hat{Y}_i$ is the predicted value.

- LMLS (Least Mean Log Squares). This error measure can be used as a regression loss function in the training process or as a validation error measurement [51], equation 9.

$$LMLS = \frac{1}{n} \sum_{i=1}^{n} \log \left(1 + \frac{1}{2} \left(Y_i - \hat{Y}_i \right)^2 \right) \tag{9}$$

where Y_i is the real value and $\hat{Y}_i$ is the predicted value.

- sMAPE (Symmetric Mean Absolute Percentage Error). This metric is oriented to show relative errors due to the percentage of use [52], equation 10.

$$sMAPE = \frac{2}{n} \sum_{i=1}^{n} \frac{|Y_i - \hat{Y}_i|}{Y_i + \hat{Y}_i} \tag{10}$$

where Y_i is the real value and $\hat{Y}_i$ is the predicted value.

- MASE (Mean Absolute Scaled Error). This is a typical measure of the accuracy of regressions [53], equation 11.

$$MASE = \frac{1}{n} \sum_{i=1}^{n} \left(\frac{|Y_i - \hat{Y}_i|}{\frac{1}{n-1} \sum_{i=2}^{n} |Y_i - Y_{i-1}|} \right) \tag{11}$$

where Y_i is the actual value and $\hat{Y}_i$ is the predicted value.

- MSE (Mean Squared Error). This metric measure is able to include the variance of error, it is applicable to a wide variety of problems [54] equation 12.

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2 \tag{12}$$

where Y_i is the real value and $\hat{Y}_i$ is the predicted value.

- MAPE (Mean Absolute Percentage Error). This metric is one of the most popular measurements in forecasting accuracy [55], equation 13.

$$MAPE = \frac{100\%}{n} \sum_{i=1}^{n} \frac{|Y_i - \hat{Y}_i|}{Y_i} \tag{13}$$

where Y_i is the real value and $\hat{Y}_i$ is the predicted value.
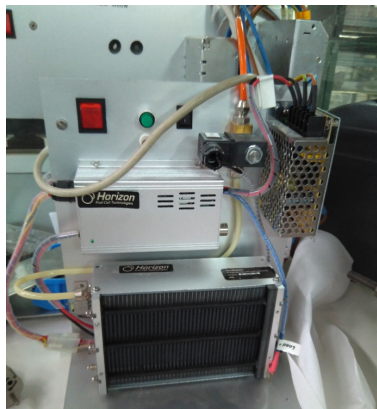
4. Datasets

Three different fuel cells (A, B and C) over the same configuration system (see Fig.1), have been used to perform different experiments and to validate our proposal, and the same parameters for both fuel cell were measured. Fuel cells A and B are from the same manufacturer (Horizon) and C from a different one (MES-DEA).

| Characteristic	Fuel Cell A	Fuel Cell B	Fuel Cell C
Type of fuel cell	PEM	PEM	PEM
Number of cells	120	48	60
Rated Power	5000W	200W	1500W
Performance	72V@70A	28V@7.2A	63V@45A
H2 Supply valve voltage	12V	12V	12V
Purging valve voltage	12V	12V	12V
Blower voltage	24V	12V	–
Flow rate at max output	65 L/min	2.6 L/min	20 L/min
Start up time	$\leq$30S	Immediate	Immediate
Efficiency of stack	40%@72V	40%@72V	40%@72V
Low voltage shut down	60V	24V	56V
Over current shut down	90A	12A	95A
Over temp. shut down	$65°C$	$65°C$	$67°C$
External power supply	24V($\pm1V$), 8A 12A	13V ($\pm1V$), <5A	9.6-12V, <2.5A

Table 1: Main characteristics fuel cells A, B and C

(a) Fuel cell (Horizon 5000W)

(b) Fuel cell B (Horizon 500W)
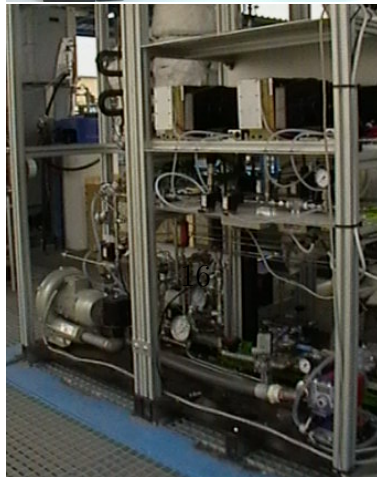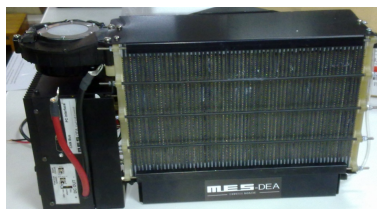
(c) Fuel cell (MES-DEA 1500W)

These datasets consists of a total of 281 (fuel cell A), 471 (fuel cell B) and 945 (fuel cell C) samples with 6 input variables and 1 output variable, with a sample rate of 1 min:

- Input variables:

  - Process temperature
  - Voltage
  - Current
  - Power
  - Anode pressure
  - Cathode pressure

- Output variable:

  - H_2 generated

In all cases the procedure carry out to create the dataset consist on sample the system following these steps:

- All fuel cells were operating with controlled output current.

- Output current was changing by steps starting from 0A to 65A, 7A and 35 for fuel cells A, B and C respectively, with increments of 5A, 1A and 0.5A for each one.

- Following, output current was decreasing to 0A by steps of 5A (fuel cell A), 1A (fuel cell B) and 0.5A (fuel cell C) every step.

- Steps were lasting approximately 10 minutes.

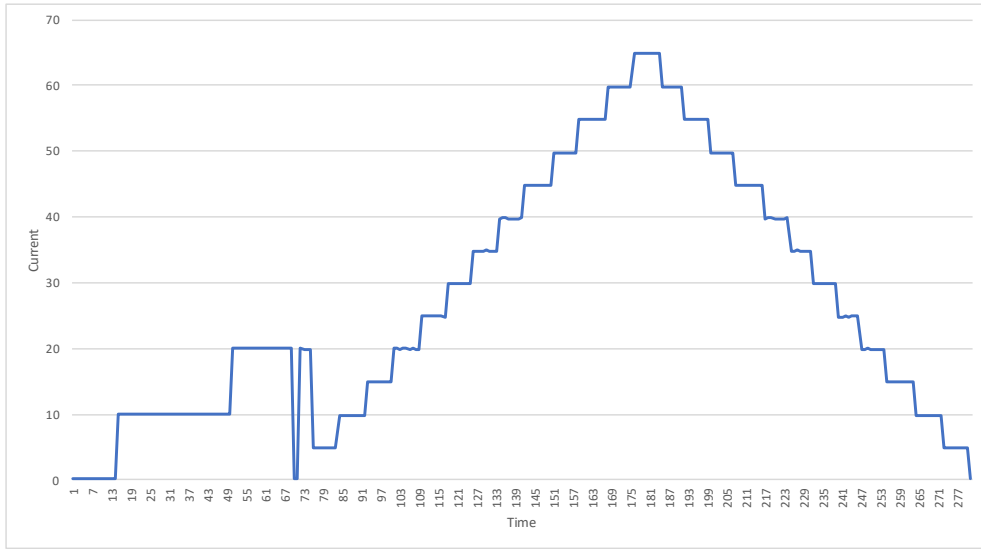Figures 6, 7 and 8 show the previously described process for fuel cells A, B and C.
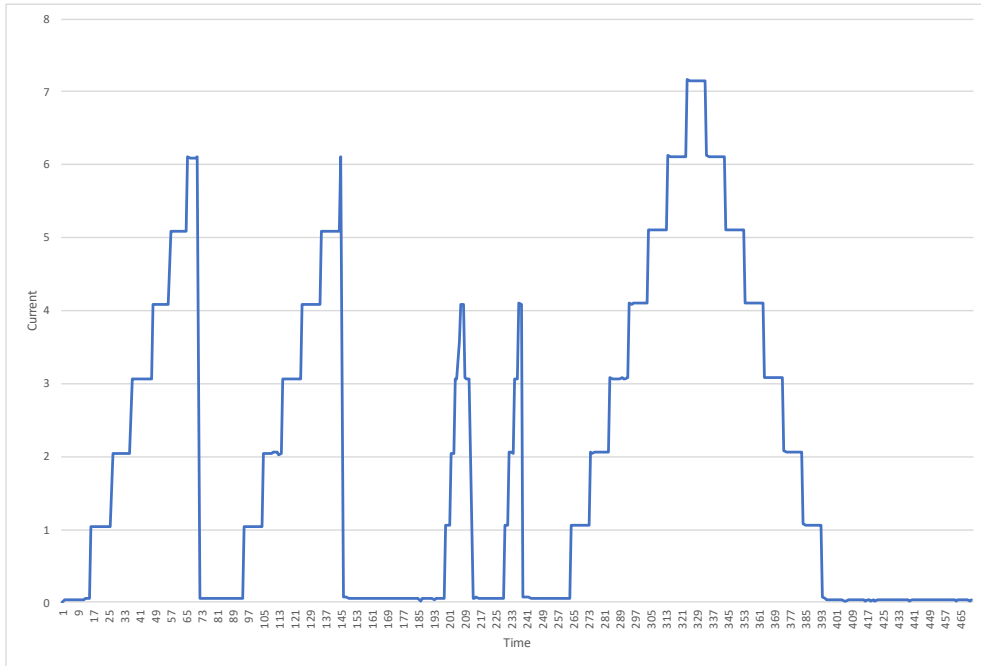
Figure 6: Fuel cell A output current

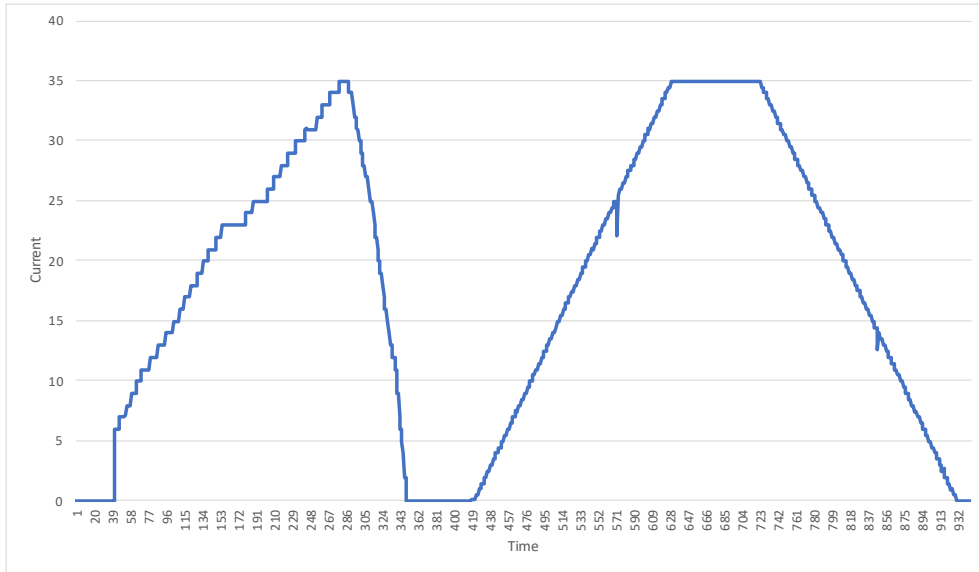Figure 7: Fuel cell B output current

18

Figure 8: Fuel cell C output current

5. Experiments and Results

This section addresses how the set of experiments are implemented, and how the grid search is associated to each one of them. After that, the parameters used for each experiment and the results are detailed.

5.1. Experiments

In a first step, the BHL algorithm is applied as a clustering technique to identify the internal structure of the datasets and clusters present in the datasets. The values of the BHL parameters were chosen in an experimental process of trial and error. The tuning of the parameters is a task that is very dependent on the dataset to be used, several initial experiments were carried out with a range of combinations of these parameter values.

In a second step three regression techniques were applied on each of the clusters identified in the previous step by BHL, with the aim of obtaining the best regression model for each cluster. For each experiment a split into two slices was made. The first split retains 80% of the preprocessed dataset for training purposes, which has the purpose of obtaining the error values and finding out which is the best model. The second split with the remaining 20% of the preprocessed data is used to test the model and show the final error values.

19

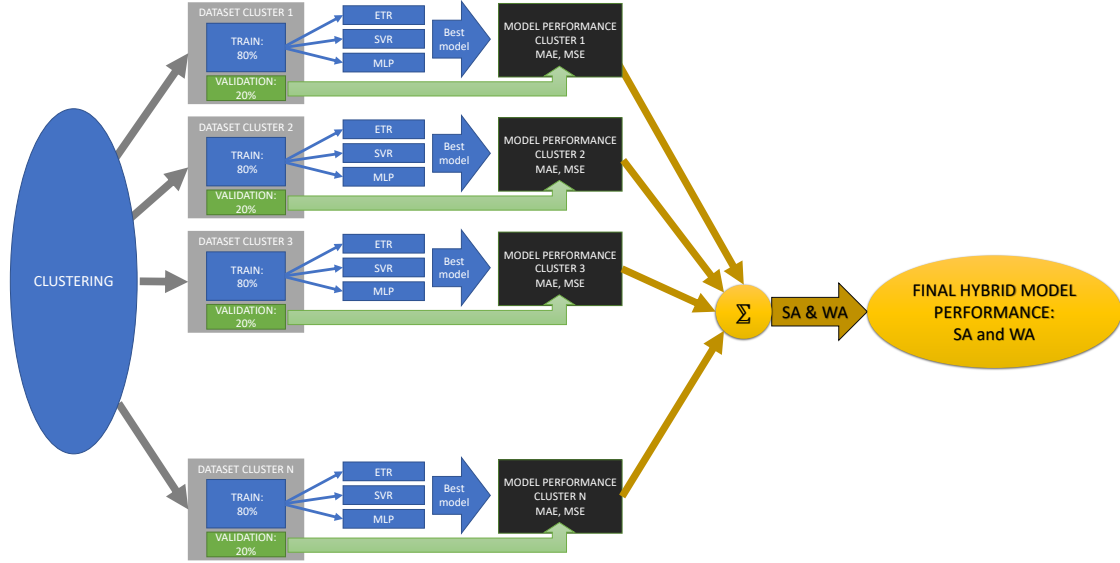Figure 9, presents the structure of the hybrid system, and the steps followed to obtain the final model.

Figure 9: Hydrid system structure.

Each of the three techniques addressed in section 3.3 has been implemented to be trained with each cluster, Therefore, each machine learning regression technique and its associated own pram-grid is applied to each cluster identified by BHL.

The Cross-validation technique has been implemented by applying *Ten fold* with the goal of getting the best combination of parameters and hence, the best model. The measurement error chosen for optimizing the test-split within the Grid Search as Cross-Validation has been the *Mean Absolute Error* (MAE). the best combinations of the param-grid found, will have the minor MAE, obtaining the best model. The hyper-parameter set of the three implemented machine learning techniques are described at [56], [57], [58] and [59].

One model was created for each fuel cell using the obtained datasets after a monitorization process. The testing procedure consist on dividing the dataset A into training and validation (80% and 20%) and for training k-folds technique were applied (training-testing). Validation was performed over validation dataset (never previously used in training). Training procedure

was repeated 10 times (10 folds, training-testing) and average errors were obtained (in order to avoid bias measures). For dataset B and C, the same process was followed independently from the other datasets. In figure 10 the training and validation procedure is summarized.
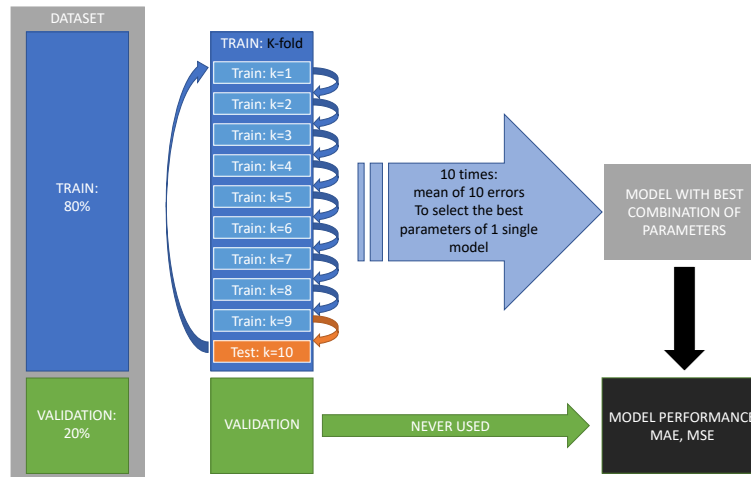

Figure 10: Training and validation procedure.

5.2. Results

5.2.1. Clustering results

Figures 11, 13 present the best projection of BHL algorithm by using the parameters presented in table 2. Four main groups are clearly presented for fuel cell A and B and three groups for fuel cell C. Clusters obtained by the projection of BHL are presented in compact groups and well separated from each other. These groups correspond to the different working states of the process, and therefore demonstrating different behaviours, so different models are needed in order to obtain the best approach.

As can be seen in the three figures, the internal structure of each dataset is different due to their corresponding to distinct fuel cells.

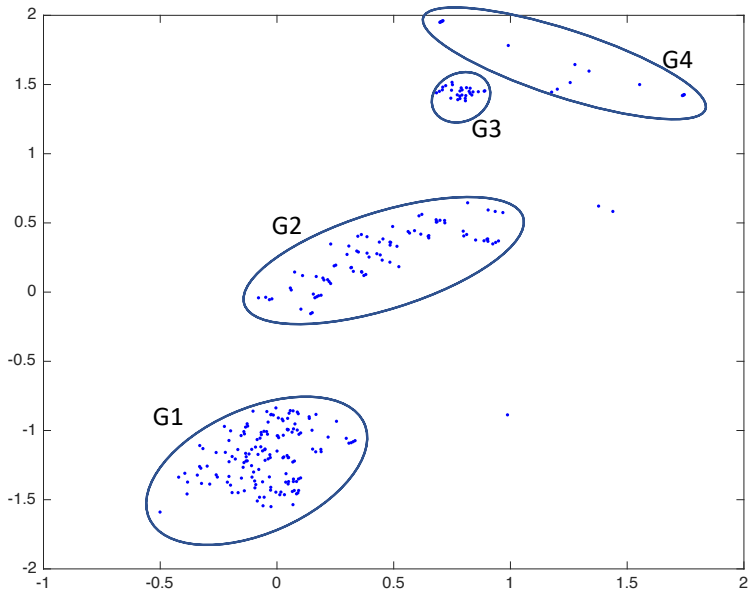| BHL (dataset A)	iters=1000, lrate=0.05, $\alpha = 3.2$, $\beta = 4.3$
BHL (dataset B)	iters=10000, lrate=0.01, $\alpha = 3$, $\beta = 3$
BHL (dataset C)	iters=50000, lrate=0.05, $\alpha = 3.5$, $\beta = 4.1$

Table 2: BHL parameters for dataset A, B and C
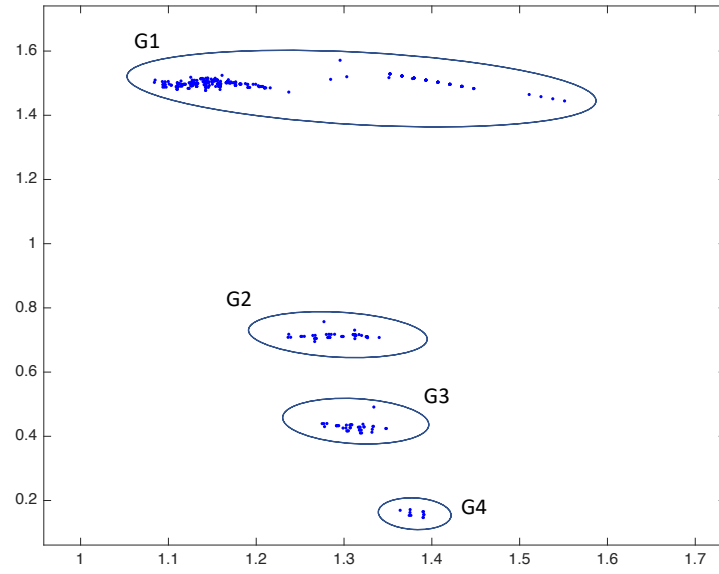

Figure 11: BHL projection for dataset A
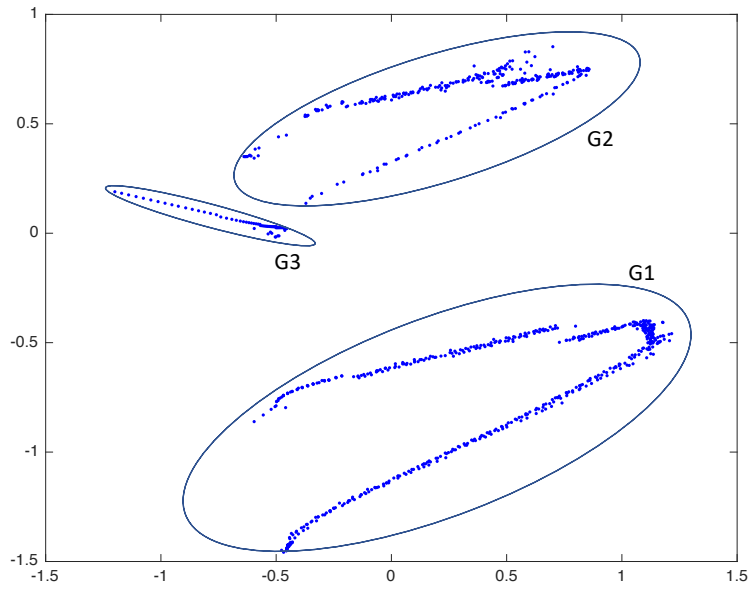
Figure 12: BHL projection for dataset B

Figure 13: BHL projection for dataset C.

Dataset A

The combination of parameters (param-grid) 361 tested for the ETR, SVR and MLP regression techniques are shown in tables 3, 4, 5.

Number estimators:	[50, 100, 200, 400, 600, 700, 800, 900]
Bootstrap:	[False,True]
Maximum depth:	[5,10,20,40,50, 60, 70]
Minimum samples leaf:	[2,3,5,7]
Minimum samples split:	[2-6]

Table 3: Paramgrid for Extra Trees Regressor method

Kernel	Linear	Poly	RBF
Tol	0.1, 0.01, 0.001, 0.0001	0.1, 0.01, 0.001, 0.0001	0.1, 0.01, 0.001, 0.0001
Gamma	-	1 ,0.1 ,0.01 ,0.001 ,0.0001	1, 0.1, 0.01, 0.001, 0.0001
Degree	-	2, 3, 4, 5, 6, 7.	- .
C	1, 10, 100, 1000, 10000, 100000	1, 10, 100, 1000, 10000, 100000	1, 10, 100, 1000, 10000, 100000

Table 4: Paramgrid for SVR method

Early Stopping:	[False, True]
Hidden layer sizes:	[5-20]
Nesterov momentum:	[False, True]
Solver:	[*lbfgs, sgd, adam*]
Learning rate init:	[0.9, 0.5, 0.01, 0.15]
Activation:	[*relu, tanh*]
Warm start:	[*True, False*]
Batch size	[5, 10, 15, 20]

Table 5: Paramgrid for MLP Regressor method

The best combination of parameters for the best regression technique extracted from the Ten-fold Grid Search Cross Validation are summarized in table 6:

CLUSTER 1:	ETR, MAE: 1.0451, Number estimators : 600, Bootstrap : False, Maximum depth: 20, Minimum samples leaf: 2, Minimum samples split: 2
CLUSTER 2:	ETR, MAE: 1.2814, Number estimators : 50, Bootstrap : False, Maximum depth: 20, Minimum samples leaf: 2, Minimum samples split: 5
CLUSTER 3:	SVR, MAE: 1.2184, Kernel: RBF, Tol: 0.1, Gamma: 0.01, C: 1000
CLUSTER 4:	MLP, MAE: 0.0103, Early Stopping: True, Hidden layer sizes: 19, Nesterovs momentum: True, Solver: lbfgs, Learning rate init: 0.9, Activation: tanh, Warm start: False, Batch size: 5

Table 6: Parameters for the best model for each cluster for dataset A

The results are defined shaped like the most common error measurement of the twelve experiments. These are grouped by regression techniques applied to four clusters. Tables 7, 8 and 9 show the results.

$Cluster$	1	2	3	4
LMLS	0.4771	0.6266	0.6857	0.0003
MSE	1.5253	2.9704	2.2844	0.0006
MAPE	0.0600	0.1231	0.2293	Inf
MAE	1.0451	1.2814	1.4021	0.0164
SMAPE	0.0589	0.1219	0.2037	1.0574
MASE	0.0896	0.2457	1.8671	0.0091

Table 7: Extra Tree Regression error measures for dataset A

Cluster	1	2	3	4
LMLS	1.1093	0.8485	0.6212	0.0018
MSE	9.2785	6.9840	2.0567	0.0036
MAPE	0.0640	0.0273	0.0281	Inf
MAE	2.2364	1.7867	1.2184	0.0522
SMAPE	0.1221	0.1833	0.1784	1.5023
MASE	0.200	0.4180	1.9207	0.0228

Table 8: SVR error measures for dataset A

Cluster	1	2	3	4
LMLS	1.6354	1.6509	0.9118	0.0001
MSE	33.6201	9.7880	3.2027	0.0002
MAPE	0.1768	0.3291	0.2851	Inf
MAE	3.8177	2.9828	1.7331	0.0103
SMAPE	0.1672	0.2839	0.2492	1.0237
MASE	0.3234	0.5290	83.417	0.0057

Table 9: MLP error measures for dataset A

Graphical representation as real output (dashed line blue) versus predicted output (in red) per each experiment are presented in Figures 14, 16, 15 and 17. The "Y" axis represents the Hydrogen value while the "X" axis represents each data sample of 20% cases of the final validation data split for each cluster.
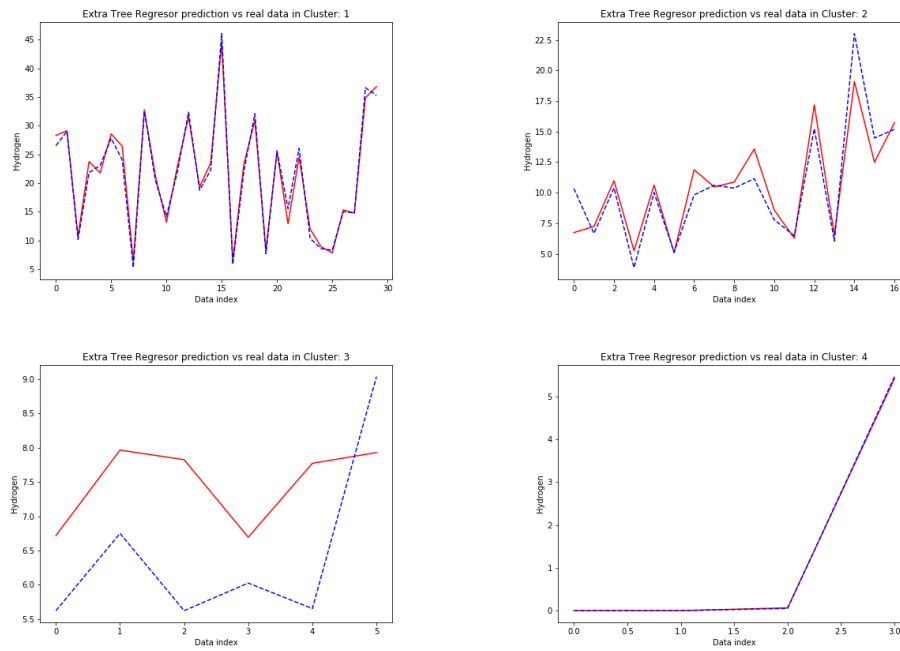
Figure 14: Real data vs. Extra Tree Regressor prediction for dataset A
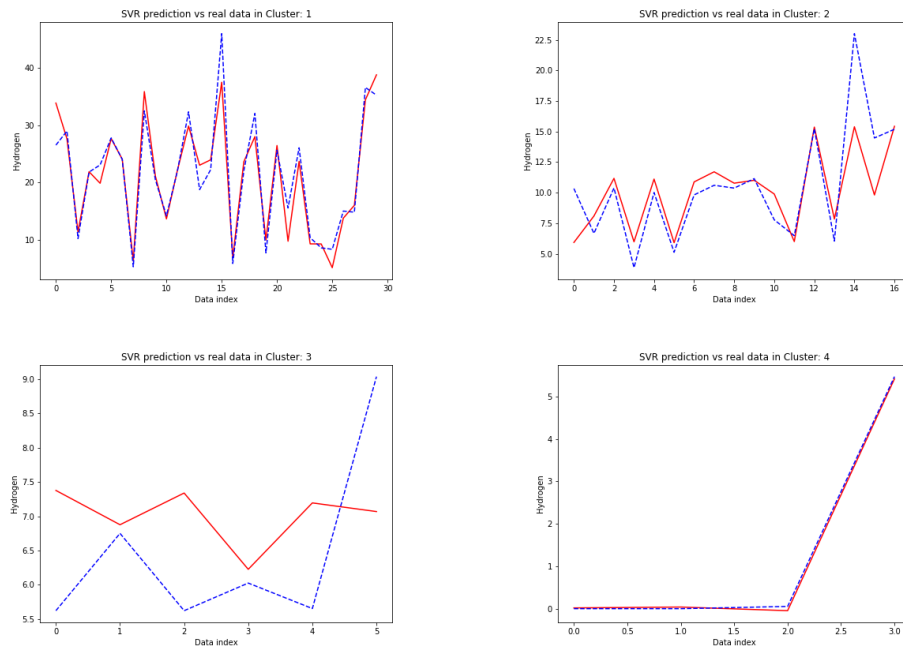
Figure 15: Real data vs. SVR prediction for dataset A

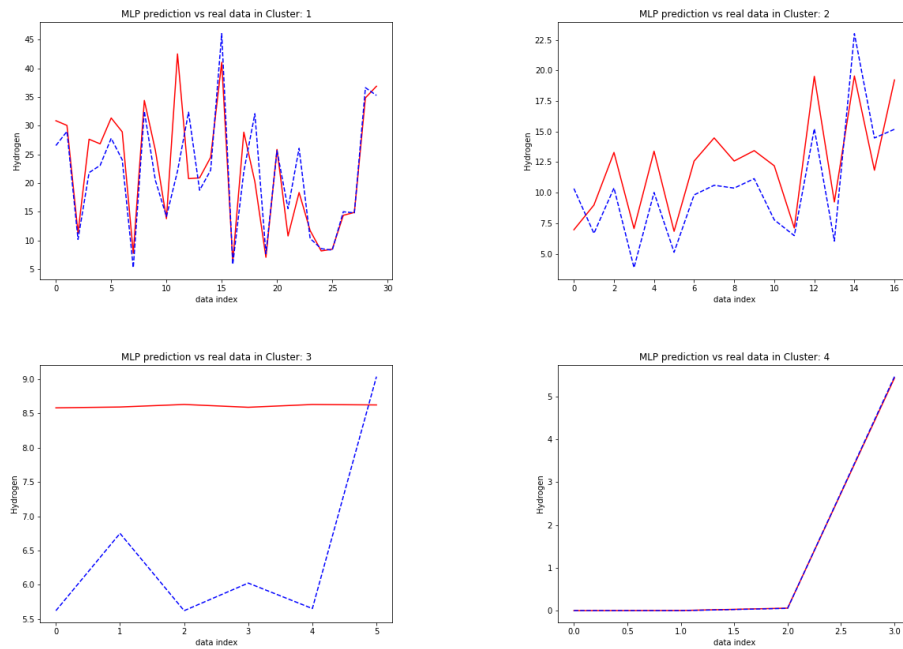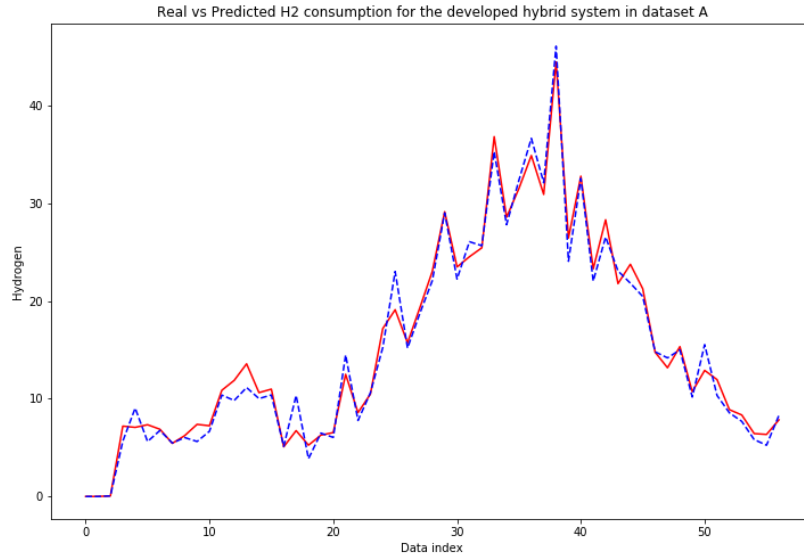Figure 16: Real data vs. MLP predictions for dataset A

Figure 17: Real vs Predicted output for the developed hybrid system in dataset A

Dataset B

For the dataset of the second fuel cell B followed the same procedure as in the case of the dataset for fuel cell A. After that, the best regression technique for each cluster is shown, as well as the MAE and MSE measures for each cluster (Table 10):

CLUSTER 1:	ETR, MAE: 0.0444, MSE:0.004791, Number estimators: 200, Bootstrap: True, Maximum depth: 20, Minimum samples leaf: 3, Minimum samples split: 2
CLUSTER 2:	ETR, MAE: 0.0348, MSE: 0.002409, Number estimators: 900 , Bootstrap : True, Maximum depth: 10, Minimum samples leaf: 2, Minimum samples split: 5
CLUSTER 3:	ETR, MAE: 0.0442, MSE: 0.002785, Number estimators: 700 , Bootstrap : False, Maximum depth: 5, Minimum samples leaf: 2, Minimum samples split: 5
CLUSTER 4:	ETR, MAE: 0.0608, MSE: 0.005693, Number estimators: 50 , Bootstrap : False, Maximum depth: 5, Minimum samples leaf: 2, Minimum samples split: 2

Table 10: Parameters for the best model for each cluster fuel cell B

The results for both datasets are defined shaped like the most common error measures of experiments. These are grouped by regression techniques applied to four clusters. Tables 11, 12 and 13 show the results for fuel cell B.:

Cluster	1	2	3	4
LMLS	0.0024	0.0012	0.0014	0.0028
MSE	0.0048	0.0024	0.0028	0.0057
MAPE	0.1097	0.0375	0.0206	Inf
MAE	0.0444	0.0343	0.0442	0.0609
SMAPE	0.0809	0.0364	0.0206	0.0290
MASE	0.0668	0.0471	0.2324	0.1822

Table 11: Extra Tree Regression error measures fuel cell B.

Cluster	1	2	3	4
LMLS	0.0046	0.0026	0.0032	0.0052
MSE	0.0093	0.0053	0.0065	0.0104
MAPE	0.2358	0.0675	0.0364	Inf
MAE	0.0738	0.0627	0.0731	0.1018
SMAPE	0.1770	0.0671	0.0361	0.0524
MASE	0.1111	0.0872	0.3348	0.3478

Table 12: SVR error measures fuel cell B.

Cluster	1	2	3	4
LMLS	0.0023	0.0046	0.0061	0.0419
MSE	0.0047	0.0093	0.0123	0.0868
MAPE	0.1768	0.3291	0.2851	Inf
MAE	0.0453	0.0777	0.1053	0.2550
SMAPE	0.0619	0.0681	0.0515	0.1322
MASE	0.0674	0.1142	1.0561	0.3238

Table 13: MLP error measures fuel cell B.

Graphical representation as real output (dashed line blue) versus predicted output (in red) per each experiment in dataset B, are presented in Figures 18, 20, 19 and 21. The "Y" axis represents the Hydrogen value while the "X" axis represents each data sample of 20% cases of the final validation data split for each cluster.
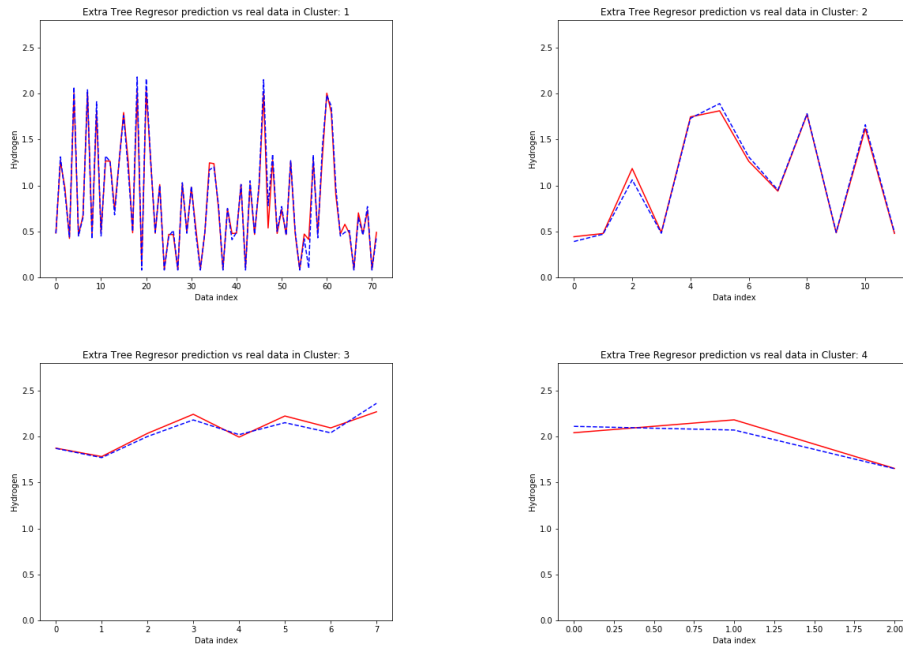
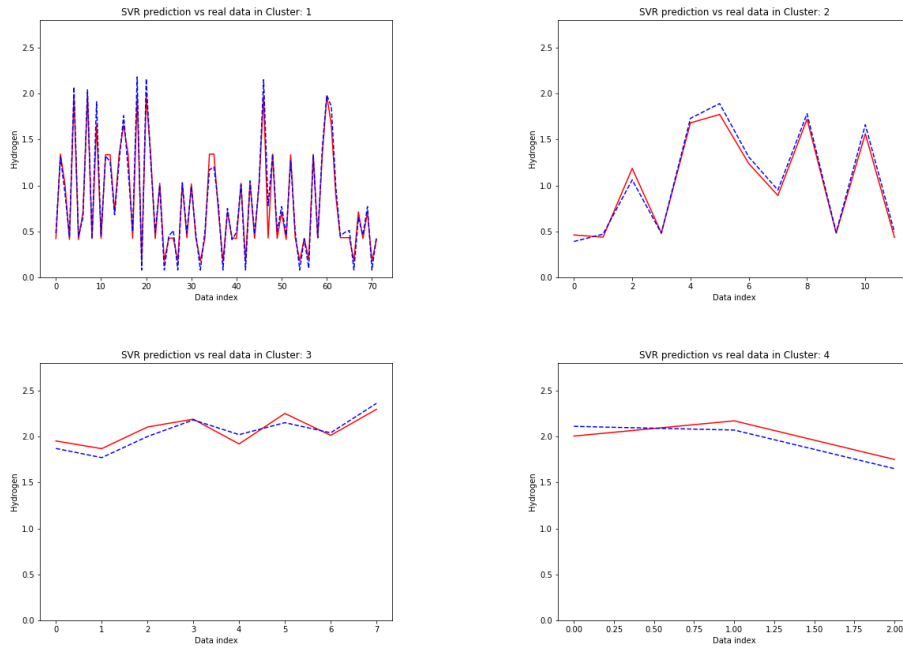Figure 18: Real data vs. Extra Tree Regressor prediction fuel cell B.

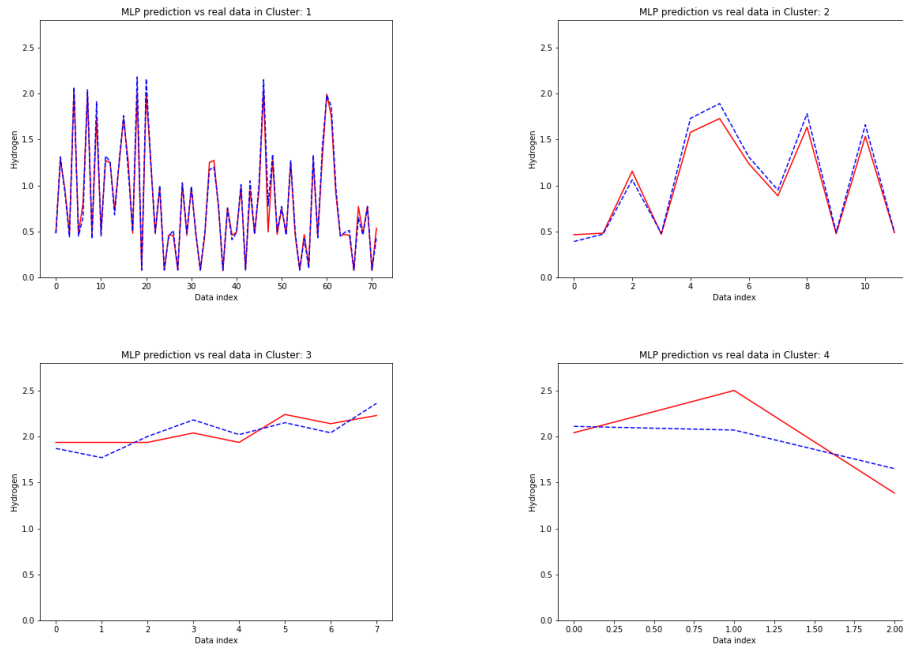Figure 19: Real data vs. SVR prediction fuel cell B.

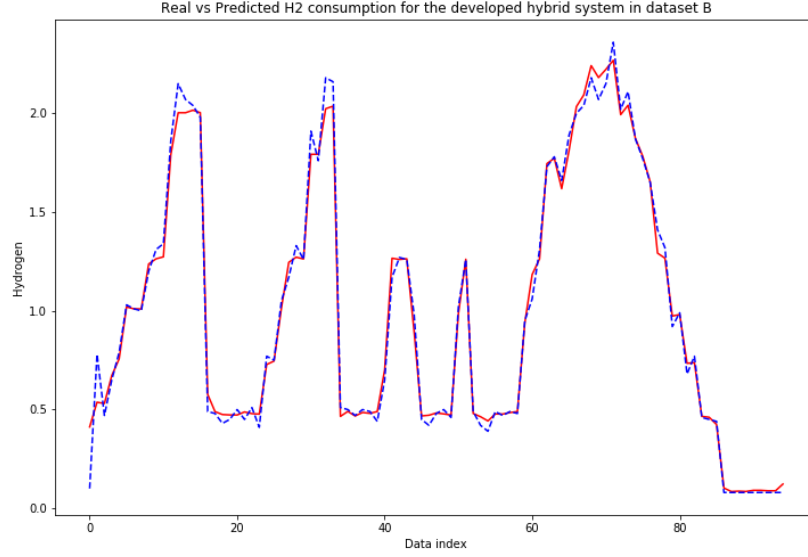Figure 20: Real data vs. MLP predictions fuel cell B.

Figure 21: Real vs Predicted output for the developed hybrid system in dataset B

Dataset C

In the experimental procedure for this dataset of the third fuel cell (C) it has been followed same steps for fuel cell A and B. Following, the best regression technique for each cluster is showed, as well as the MAE and MSE measures for each cluster (Table 14):

CLUSTER 1:	ETR, MAE: 0.3385, MSE:0.2610, Number estimators : 100, Bootstrap: True, Maximum depth: 20, Minimum samples leaf: 3, Minimum samples split: 2
CLUSTER 2:	ETR, MAE: 0.8161, MSE: 2.1536, Number estimators : 200 , Bootstrap : True, Maximum depth: 10, Minimum samples leaf: 2, Minimum samples split: 2
CLUSTER 3:	SVR, MAE: 0.1176, Kernel: RBF, Tol: 0.01, Gamma: 1, C: 1000

Table 14: Parameters for the best model for each cluster fuel cell C

Following most common error measures of experiments are grouped by regression techniques applied to three clusters. Tables 15, 16 and 17 show the results for fuel cell C.:

Cluster	1	2	3
LMLS	2.6372	2.3981	0.5939
MSE	0.2610	2.1536	0.1852
MAPE	0.5140	Inf	Inf
MAE	0.3385	0.8161	0.1648
SMAPE	0.4441	0.5114	Inf
MASE	0.9413	01.0175	0.8199

Table 15: Extra Tree Regression error measures fuel cell C.

Cluster	1	2	3
LMLS	2.6295	2.4722	0.54642
MSE	0.2851	3.6985	0.0494
MAPE	0.5114	Inf	Inf
MAE	0.3777	1.3088	0.1176
SMAPE	0.4420	0.5275	1.9683
MASE	0.9547	1.0156	0.8900

Table 16: SVR error measures fuel cell C.

Cluster	1	2	3
LMLS	2.6406	2.3787	0.5320
MSE	0.2624	2.1822	0.1036
MAPE	0.5151	Inf	Inf
MAE	0.3525	0.8922	0.1273
SMAPE	0.4452	0.5075	1.97215
MASE	0.9475	0.9644	0.9465

Table 17: MLP error measures fuel cell C.

Graphical representation like real output (dashed line blue) versus predicted output (in red) per each experiment, can be seen in Figures 22, 24, 23 and 25. The "Y" axis represents the Hydrogen value while the "X" axis represents each data sample of 20% cases of the final validation data split for each cluster.
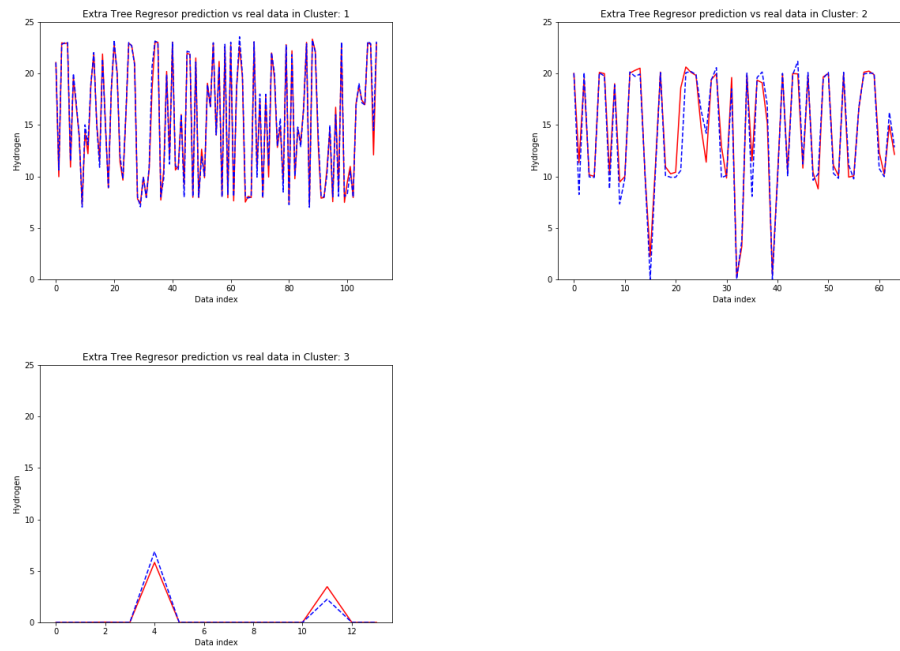
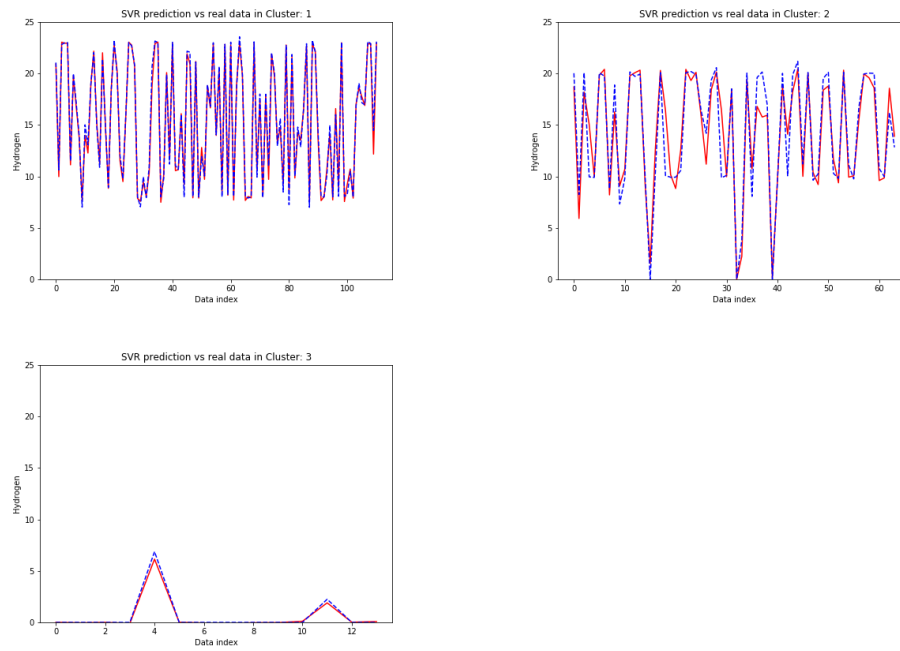Figure 22: Real data vs. Extra Tree Regressor prediction fuel cell C.

Figure 23: Real data vs. SVR prediction fuel cell C.
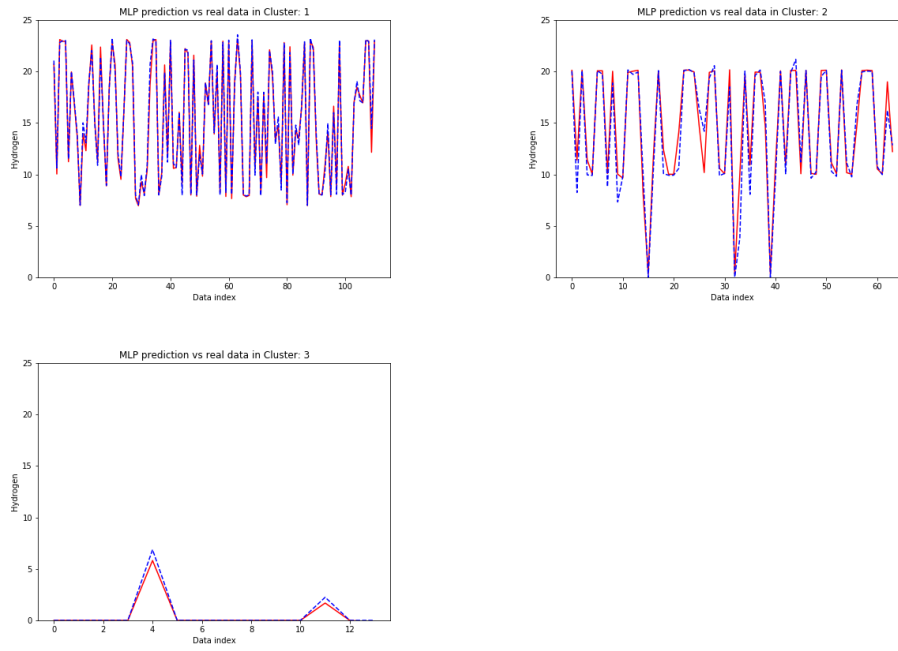
Figure 24: Real data vs. MLP predictions fuel cell C.
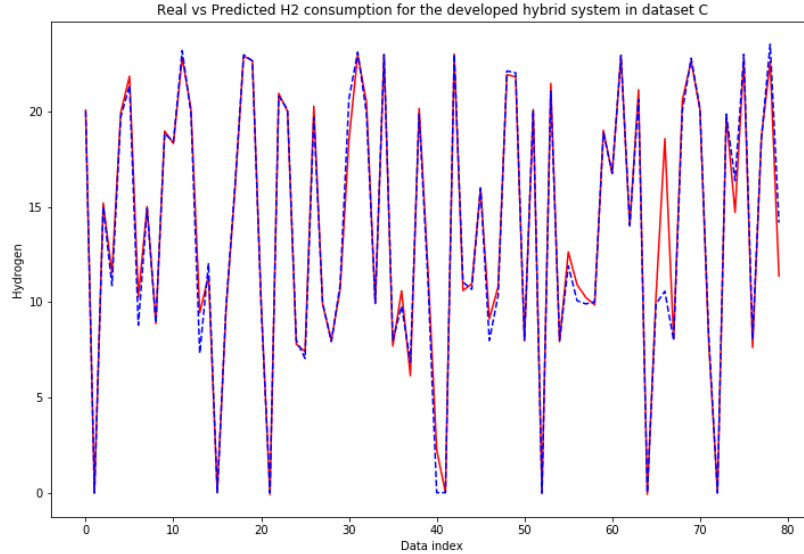
Figure 25: Real vs Predicted output for the developed hybrid system in dataset C

5.2.3. Comparative Results

Finally in tables 18, 19 and 20, it is shown the MAE and MSE results for:

- General global model without clustering process for each regression technique (ETR, MLP, SVR).

- Local model defined by the cluster process using the same regression technique (ETR, MLP or SVR) for all clusters and showing the simple average and weighted average for each algorithm.

- Local best model defined by the clustering algorithm, using the best regression technique for each cluster and showing the simple average (SA) and weighted average (WA).

Such combination of MAE and MSE errors for each cluster has been performed in two different ways:

- Simple average (SA): Each cluster has the same weight so the average MAE and MSE for each cluster is calculating as the sum of each MAE

41

and MSE divided by total number of clusters, 4 in Cell Fuel A and B, and 3 in fuel Cell C.

- Weighted average (WA): Each cluster has a weight, proportional to the total size of the cluster, C1 (53.7%), C2 (30%), C3 (10.5%) and C4 (5.8%) for Cell Fuel A. C1 (76.82%), C2 (12.44%), C3 (8.36%) and C4 (2.36%) for Cell Fuel B. And, C1(58.86%), C2(33.75%) and C3(7.4%) for Fuel Cell C. So the average is calculated as the sum of each MAE or MSE weighted by its size (in %).

$Model$	MAE	MSE
GLOBAL ETR	2.195	7.916
GLOBAL SVR	2.203	10.797
GLOBAL MLP	2.467	10.810
LOCAL BEST MODEL (SA)	0.888	1.638
LOCAL BEST MODEL (WA)	1.074	1.925

Table 18: Comparison of global and local models based on MAE and MSE for fuel cell A

$Model$	MAE	MSE
GLOBAL ETR	0.0652	0.01768
GLOBAL SVR	0.1339	0.07739
GLOBAL MLP	0.0830	0.02124
LOCAL BEST MODEL (SA)	0.0459	0.00434
LOCAL BEST MODEL (WA)	0.0435	0.00391

Table 19: Comparison of global and local models based on MAE and MSE for fuel cell B

$Model$	MAE	MSE
GLOBAL ETR	0.5712	1.0976
GLOBAL SVR	1.0714	3.8509
GLOBAL MLP	0.6669	0.9707
LOCAL BEST MODEL (SA)	0.4240	0.8213
LOCAL BEST MODEL (WA)	0.4833	0.8842

Table 20: Comparison of global and local models based on MAE and MSE for fuel cell C

Based on MAE and MSE measures summarize in Tables 18, 19 and 20, as the most representative from among the measures used in this research, detailed in Table 7, 8, 9, 11, 12, 13, 15, 16 and 17.

It can be concluded that the best solution is based on a procedure where a clustering algorithm is applied first, for apply after, a set of regression techniques to each cluster, in order to get a robust hybrid model with the best regression technique for cluster.

Table Table 6 shows MAE and MSE for each cluster as well as the best combination of regression techniques (ETR for clusters 1 and 2, SVR for cluster 3 and MLP for cluster 4) for the fuel cell A. This approach with a MAE of 0.888(SA) & 1.074(WA) and a MSE of 1.638(SA) & 1.925(WA) outperforms the global approach based on ETR global model as show Table 13, where the best global model has a MAE of 2.195 and a MSE of 7.916, being worse than local model approach.

Results for fuel cell B illustrate that the local approach where the ETR is applied to each of the clusters optimized the MAE measure. That is true that in the cluster 1 a better result of MSE equal to 0.0047 can be obtained applying MLP technique in cluster 1. However the difference using ETR or MLP is minimum (0.0001), being the MSE using ETR in cluster 1 equal to 0.0048, while the difference from MAE point of view is higher (0.0009), when MLP is used in cluster 1 (0.0453) versus to use ETR in cluster 1 (0.0444). For this reason, ETR will be chosen like regression technique for all cluster for implementing the local model approach in the cell fuel B.

This way, the local model solution where ETR is utilized over the four clusters gets a MAE of 0.045(SA) & 0.0435(WA) and a MSE of 0.00434(SA) & 0.00391(WA) while the global model using ETR without clustering, deliver a results with a MAE of 0.0652 and a MSE of 0.01768 (Table 14). It is necessary to consider that the difference of both approaches from MAE and MSE point of view, might seem minimum in this experiment, however, the fuel cell B works in very small range of values compared with the fuel cell A.

In case of fuell cell C, the hydrid model outperforms the results obtained by the global model, by using ETR for cluster 1 and 2 and SVR for cluster 3, it is obtained a reduction in the MAE of 25,7% with respect to the best global model (ETR) and a reduction of MSE of 15.4%.

Figures 17, 21 and 25 show how the final hybrid model composed by the four and three local models works. It can see how in the majority of cases

the solution purposed in this work has an optimal behavior.

Considering all reflections previously addressed in this section. It can be said that the use of combination of local models in order to get a robust final hybrid model is a better approach, that the global model without the implementation of clustering procedure. Even if the training process is more complex for the local approach compared to the global one.

6. Conclusions and future works

In this paper, a combination of clustering and modelling techniques are applied to a real dataset for predicting the consumption of H_2 by a fuel cell. As a result, the combination of both techniques outperforms the result of a single general model.

Due to the complexity in obtaining a parametric model of a fuel cell system to find out the amount of hydrogen consumed based on main parameters such as voltage, current, etc., the use of intelligent techniques seems to be a suitable option for predicting the amount of hydrogen demanded by means of experimental data. However, due to the nonlinear behaviour of fuel cells, the use of only one overall model could not achieve as good results as those obtained from several local models.

Local models based on clustering and regression have been successfully applied in previous works over different industrial systems [32, 35], especially for non-linear systems and with well differentiated states of behaviour. In particular, when human knowledge of the system's behaviour is relevant, the use of projectionist techniques such as BHL has demonstrated good results, as it allows this human expertise to be included during the clustering phase, prior to the application of regression techniques [60, 61, 62].

Therefore in this research, we have successfully applied local models outperforming the results achieved by a general model during the task of predicting the consumption of hydrogen by a fuel cell. The results showed that the predicted output of hydrogen consumed, provide a good idea of the real output of the hydrogen consumed.

It is also important to highlight that different fuel cells present different behaviours and therefore it is necessary to generate one specific model for each one based on experimental datasets.

Thus, in order to validate the proposed approach and generalize its applicability to any fuel cell, 3 different fuel cells located in different systems have been tested. In both cases, the results obtained are good, allowing a model

44

to be obtained for predicting the consumption of hydrogen when a charge is demanding energy from the fuel cell.

Based on the results obtained, it can be concluded that the proposed methodology outperforms some other classic methods in the task of predicting the hydrogen consumed in the fuel cell.

To achieve a good performance of the predicted system it is important to obtain a complete dataset containing information on all working states of the system. This means that the system should be pass through all the different states of behaviour, with different combinations of measured parameters (current, voltage, power, etc.). Therefore a systematic procedure, consisting of increasing the current demanded up to the maximum and decreasing it again to zero, should be done in order to generate more accurate and simple models.

Due to the possible existence of small clusters (with a few samples), it is possible to get very small error prediction values for such clusters, however the contribution of these small errors are weighted by its size, so the final errors are not very influenced by them.

The use of the methodology presented allows the hydrogen consumed for any real fuel cell system to be predicted, as the procedure can be applied using different measured variables, such as new temperatures points, pressure in different parts of the system, hydrogen flow, etc.

As general conclusion, in this research a methodology to predict hydrogen consumption through hybrid intelligent system that can be applied to any fuel cell is presented and validated. Therefore, the generated model can be used in simulation processes, without the need to use the real system, to carry out different experiments.

Future works will be divided into two objectives: First, generated inverse models to predict the current generated based on the hydrogen consumed. Second, develop more sophisticated models using complex prediction techniques such as deep learning models and other clustering techniques.

7. References

References

[1] J.-A. Montero-Sousa, J.-L. Casteleiro-Roca, J.-L. Calvo-Rolle, Evolution of the electricity sector after the 2nd world war, DYNA 92 (2017) 280–284.

[2] J. A. Montero-Sousa, J. L. Casteleiro-Roca, J. L. Calvo-Rolle, The electricity sector since its inception until the second world war, DYNA 92 (2017) 43–47.

[3] S. Chen, F. Zhu, H. Long, J. Yang, Energy footprint controlled by urban demands: How much does supply chain complexity contribute?, Energy 183 (2019) 561 – 572.

[4] G. Carrosio, I. Scotti, The 'patchy' spread of renewables: A socio-territorial perspective on the energy transition process, Energy Policy 129 (2019) 684 – 692.

[5] M. D. de Souza Dutra, M. F. Anjos, S. L. Digabel, A general framework for customized transition to smart homes, Energy 189 (2019) 116138.

[6] M. Amin, Smart grid, Public Utilities Fortnightly (2015).

[7] M. Nizami, A. Haque, P. Nguyen, M. Hossain, On the application of home energy management systems for power grid support, Energy 188 (2019) 116104.

[8] N. Good, E. A. M. Ceseña, C. Heltorp, P. Mancarella, A transactive energy modelling and assessment framework for demand response business cases in smart distributed multi-energy systems, Energy 184 (2019) 165 – 179. Shaping research in gas-, heat- and electric- energy infrastructures.

[9] C.-J. Yang, R. B. Jackson, Opportunities and barriers to pumped-hydro energy storage in the united states, Renewable and Sustainable Energy Reviews 15 (2011) 839–844.

[10] E. Hache, A. Palle, Renewable energy source integration into power networks, research trends and policy implications: A bibliometric and research actors survey analysis, Energy Policy 124 (2019) 23 – 35.

[11] B. Dunn, H. Kamath, J.-M. Tarascon, Electrical energy storage for the grid: a battery of choices, Science 334 (2011) 928–935.

[12] M. H. Westbrook, The Electric Car: Development and future of battery, hybrid and fuel-cell cars, IET Digital Library, 2001.

[13] P. J. Hall, E. J. Bain, Energy-storage technologies and electricity generation, Energy policy 36 (2008) 4352–4355.

[14] H. Tao, J. L. Duarte, M. A. Hendrix, Line-interactive ups using a fuel cell as the primary source, IEEE Transactions on Industrial Electronics 55 (2008) 3012–3021.

[15] A. Dicks, D. A. J. Rand, Fuel cell systems explained, Wiley Online Library, 2018.

[16] C. W. Potter, A. Archambault, K. Westrick, Building a smarter smart grid through better renewable energy information, in: Power Systems Conference and Exposition, 2009. PSCE'09. IEEE/PES, IEEE, pp. 1–5.

[17] S. A. Raza, N. Shah, A. Sharif, Time frequency relationship between energy consumption, economic growth and environmental degradation in the united states: Evidence from transportation sector, Energy 173 (2019) 706 – 720.

[18] R. Jin, K. Cho, C. Hyun, M. Son, Mra-based revised CBR model for cost prediction in the early stage of construction projects, Expert Systems with Applications 39 (2012) 5214–5222.

[19] J. Osborn, D. Guzman, F. de Cos Juez, A. Basden, T. Morris, E. Gendron, T. Butterley, R. Myers, A. Guesalaga, F. S. Lasheras, et al., Open-loop tomography with artificial neural networks on canary: on-sky results, Monthly Notices of the Royal Astronomical Society 441 (2014) 2508–2514.

[20] R. Mishra, T. K. Saha, Combined control of stand-alone energy conversion scheme for distributed sources: Development and performance analysis, International Journal of Electrical Power and Energy Systems 115 (2020) 105480.

[21] Y. Cho, H. B. Awbi, A study of the effect of heat source location in a ventilated room using multiple regression analysis, Building and Environment 42 (2007) 2072–2082.

[22] N. Basurto, Ángel Arroyo, R. Vega, H. Quintián, J. L. Calvo-Rolle, Álvaro Herrero, A hybrid intelligent system to forecast solar energy production, Computers and Electrical Engineering 78 (2019) 373 – 387.

[23] J. A. Rincon, V. Julian, C. Carrascosa, A. Costa, P. Novais, Detecting emotions through non-invasive wearables, Logic Journal of the IGPL 26 (2018) 605–617.

[24] A. Moreno-Fernandez-de Leceta, J. M. Lopez-Guede, L. Ezquerro Insagurbe, N. Ruiz de Arbulo, M. Graña, A novel methodology for clinical semantic annotations assessment, Logic Journal of the IGPL 26 (2018) 569–580.

[25] F. Segovia, J. M. Górriz, J. Ramírez, F. J. Martinez-Murcia, M. García-Pérez, Using deep neural networks along with dimensionality reduction techniques to assist the diagnosis of neurodegenerative disorders, Logic Journal of the IGPL 26 (2018) 618–628.

[26] S. Wojciechowski, A comparison of classification strategies in rule-based classifiers, Logic Journal of the IGPL 26 (2018) 29–46.

[27] T. S. Mahmoud, B. S. Ahmed, M. Y. Hassan, The role of intelligent generation control algorithms in optimizing battery energy storage systems size in microgrids: A case study from western australia, Energy Conversion and Management 196 (2019) 1335 – 1352.

[28] J. Nieves-Acedo, I. Santos-Grueiro, P. Garcia-Bringas, Enhancing the prediction stage of a model predictive control systems through meta-classifiers, Dyna 88 (2013) 290–298.

[29] A. S. Sánchez, P. R. Fernández, F. S. Lasheras, F. J. de Cos Juez, P. G. Nieto, Prediction of work-related accidents according to working conditions using support vector machines, Applied Mathematics and Computation 218 (2011) 3539–3552.

[30] O. Fontenla-Romero, J. L. Calvo-Rolle, Artificial intelligenge in engineering: past, present and future, DYNA 93 (2018) 350–352.

[31] Y. Peng, J. Li, S. Park, K. Zhu, M. M. Hassan, A. Alsanad, Energy-efficient cooperative transmission for intelligent transportation systems, Future Generation Computer Systems 94 (2019) 634 – 640.

[32] D. Martínez-Rego, O. Fontenla-Romero, A. Alonso-Betanzos, Efficiency of local models ensembles for time series prediction, Expert Syst. Appl. 38 (2011) 6884–6894.

[33] M. H. Ghaseminezhad, A. Karami, A novel self-organizing map (SOM) neural network for discrete groups of data clustering, Appl. Soft Comput. 11 (2011) 3771–3778.

[34] J.-L. Casteleiro-Roca, J. L. Calvo-Rolle, J. A. Méndez Pérez, N. Roqueñí Gutiérrez, F. J. de Cos Juez, Hybrid intelligent system to perform fault detection on bis sensor during surgeries, Sensors 17 (2017) 179.

[35] H. Quintián, J. L. Calvo-Rolle, E. Corchado, A hybrid regression system based on local models for solar energy prediction, Informatica 25 (2014) 265–282.

[36] Scikit-learn, Min max scaler, 2018.

[37] A. Géron, Hands-On Machine Learning with Scikit-Learn and Tensor-Flow: Concepts, Tools, and Techniques for Building Intelligent Systems, O'Reilly Media, 2017.

[38] H. Quintián, E. Corchado, Beta hebbian learning as a new method for exploratory projection pursuit, Int. J. Neural Syst. 27 (2017) 1–16.

[39] Y. Bengio, I. J. Goodfellow, A. Courville, Deep learning, Nature 521 (2015) 436–444.

[40] P. Geurts, D. Ernst, L. Wehenkel, Extremely randomized trees, Machine learning 63 (2006) 3–42.

[41] D. Sunter, P. Berkeley, D. Kammen, City-integrated photovoltaics sustainably satisfy urban transportation energy needs, WIT Transactions on Ecology and the Environment 204 (2016) 559–567.

[42] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt, G. Varoquaux, API design for machine learning software: experiences from the scikit-learn project, in: ECML PKDD Workshop: Languages for Data Mining and Machine Learning, pp. 108–122.

[43] S. Ruder, An overview of gradient descent optimization algorithms, CoRR abs/1609.04747 (2016).

[44] P. Moritz, R. Nishihara, M. Jordan, A linearly-convergent stochastic l-bfgs algorithm, in: Artificial Intelligence and Statistics, pp. 249–258.

[45] L. Bottou, Large-scale machine learning with stochastic gradient descent, in: Proceedings of COMPSTAT'2010, Springer, 2010, pp. 177–186.

[46] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, CoRR abs/1412.6980 (2014).

[47] Scikit-Learn, Pipeline, 2018.

[48] Scikit-Learn, Grid search cross validation, 2018.

[49] K. Duan, S. S. Keerthi, A. N. Poo, Evaluation of simple performance measures for tuning svm hyperparameters, Neurocomputing 51 (2003) 41–59.

[50] C. J. Willmott, K. Matsuura, Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance, Climate research 30 (2005) 79–82.

[51] A. M. Campoy, F. Rodríguez-Ballester, R. O. Carot, Using dynamic, full cache locking and genetic algorithms for cache size minimization in multitasking, preemptive, real-time systems, in: International Conference on Theory and Practice of Natural Computing, Springer, pp. 157–168.

[52] R. J. Hyndman, A. B. Koehler, Another look at measures of forecast accuracy, International journal of forecasting 22 (2006) 679–688.

[53] P. H. Franses, A note on the mean absolute scaled error, International Journal of Forecasting 32 (2016) 20–22.

[54] Z. Wang, A. C. Bovik, Mean squared error: Love it or leave it? a new look at signal fidelity measures, IEEE signal processing magazine 26 (2009) 98–117.

[55] S. Kim, H. Kim, A new metric of absolute percentage error for intermittent demand forecasts, International Journal of Forecasting 32 (2016) 669–679.

[56] H. Alaiz-Moreton, J. Aveleira-Mata, J. Ondicol-Garcia, A. L. Muñoz-Castañeda, I. García, C. Benavides, Multiclass classification procedure for detecting attacks on mqtt-iot protocol, Complexity 2019 (2019).

[57] Scikit-Learn, Extra trees regressor, 2018.

[58] Scikit-Learn, Random forest regressor, 2018.

[59] Scikit-Learn, Svr, 2018.

[60] R. V. Vega, H. Quintián, C. Cambra, N. Basurto, Á. Herrero, J. L. Calvo-Rolle, Delving into android malware families with a novel neural projection method, Complexity 2019 (2019) 6101697:1–6101697:10.

[61] R. V. Vega, H. Quintián, J. L. Calvo-Rolle, Á. Herrero, E. Corchado, Gaining deep knowledge of android malware families through dimensionality reduction techniques, Logic Journal of the IGPL 27 (2019) 160–176.

[62] E. Jove, J. L. Casteleiro-Roca, H. Quintián, J. A. M. Pérez, J. L. Calvo-Rolle, A fault detection system based on unsupervised techniques for industrial control loops, Expert Systems 36 (2019).