ORIGINAL RESEARCH

Computational Intelligence techniques for maximum energy efficiency of an internal combustion engine and a steam turbine of a cogeneration process

Sandra Seijo Fernández · I. del Campo · J. Echanobe · J. García-Sedano · E. Suso · E. Arbizu

Abstract This paper discusses the development of a model of a real cogeneration plant based on Computational Intelligence (CI) algorithms. In particular, two CI strategies are used: one based on artificial neural network and the other one based on a neuro-fuzzy system. Both systems are trained with a data collection from the cogeneration plant. Data mining techniques are applied to remove erroneous and redundant data, and also to obtain information about the variables and its behaviour. This task allows to select only the relevant information and is also a way to decrease the complexity of the model. In this first approach on the work, two separate subsystems of the cogeneration process are considered: an engine and a steam turbine. The obtained models are used to analyze the role of each involved variable and to derive a set of recommendations (i.e., changes in some of the input variables) to optimize the performance of the system. The recommendations applied to the models improve the behaviour of the plant providing higher energy production with a lower cost.

Introduction

Cogeneration [combined heat and power (CHP)] is defined as the simultaneous production of electricity and heat, both of which are used. The central and most fundamental principle of cogeneration is that, to maximize the many benefits that arise from it, systems should be based on the heat demand of the application [1]. Moreover, it is this use of thermal energy that makes high efficiency possible and ultimately, primary energy savings [2].

The efficiency of the combined-cycle power plants is around 60 %. As a result of this, the objective of the cogeneration is to obtain the maximum possible energy contained in the fuel. In industry, the high-temperature gases generated by engines, gas turbines or other machines are used to produce more electricity or for a process. This implies cost savings, because the amount of fuel is reduced. This fuel saving also results in a reduction of pollution. These economic and environmental factors are the reason why nowadays the number of cogeneration plants is increasing steadily.

The cogeneration plants include a large number of equipment, machinery, and instruments. Consequently, there is a huge amount of variables involved in these kinds of processes which involve very complex non-linear dynamics. Conventional methods or techniques based in mathematical models, which are derived from principles of

S. Seijo Fernández (✉) · I. del Campo · J. Echanobe
Science and Technology Faculty, University of Basque Country
UPV/EHU, Barrio Sarriena s/n, 48940 Leioa-Bizkaia, Spain
e-mail: sseijo001@ikasle.ehu.es
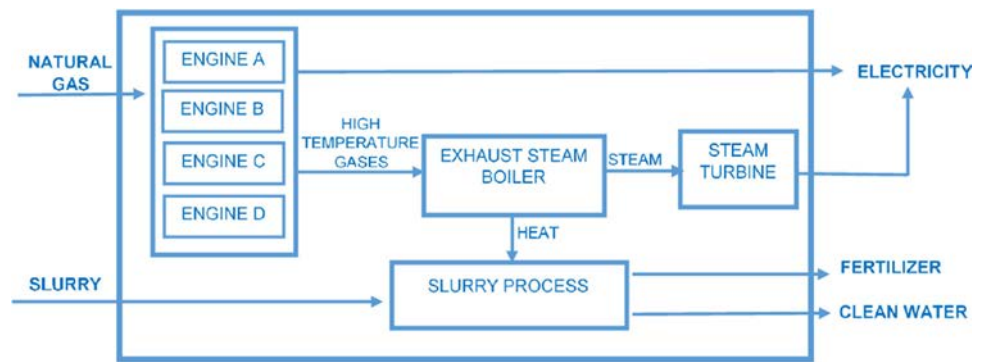
I. del Campo
e-mail: ines@we.lc.ehu.es

J. Echanobe
e-mail: javi@we.lcehu.es

J. García-Sedano · E. Suso · E. Arbizu
OPTIMITIVE S.L., Albert Einstein 15, 01510 Miñano,
Alava, Spain
e-mail: javierg@optimitive.com

E. Suso
e-mail: enriques@optimitive.com

E. Arbizu
e-mail: enekoa@optimitive.com

Fig. 1 Scheme of the cogeneration plant

physics and thermodynamics, are usually very complex. Moreover, the numerical solution of such models might be rather demanding regarding computational time. In addition, implemented in such a way that they are not suitable for real-time monitoring applications. However, they are still very valuable when an in-depth thermodynamic analysis is the main concern [3–5].

On the other hand, models based on Computational Intelligence (CI) algorithms are very useful to deal with non-linear and complex problems because they are based on the input–output behaviour of the process and consequently the model describes the overall performance [6–10], but no prior model is available.

This work proposes the development of a model of a real cogeneration plant, based on CI algorithms, with the aim of increasing the efficiency of the process [11–13] using a database from the cogeneration plant. The database contains values from variables read every minute in a 15-day period.

The significance of this work is to improve the efficiency of the cogeneration process using the models generated using CI algorithms for the engine and the steam turbine. These models are used to generate recommendations (changes) in some chosen input variables. These changes in the chosen input variables allow a reduction of fuel in the engines and provide a power increase in the steam turbine. As a result of this, the efficiency of the cogeneration process is improved.

Cogeneration plant

The cogeneration plant being evaluated is located in Huesca, in the north of Spain. It generates electrical and heat energy, with four internal combustion engines and a steam turbine. The rated electrical power of the plant is 14.8 MW and the estimated production is 118.400 MWy. The four engines are identical, with the same characteristics and the nominal power of each one being 3,700 kW. The fuel used to feed the engines is natural gas. Therefore, the engines

generate electrical energy and high-temperature gases. Subsequently, the electrical power generated is sold and the high-temperature gases go to an exhaust steam boiler. Moreover, the exhaust steam boiler generates steam for the turbine and hot water for a slurry process. After that, the steam turbine uses steam from the exhaust steam boiler to generate more electricity, with 1,000 kW of nominal power. In addition, as in the engines the power generated for the steam turbine is sold.

The slurry drying process uses the slurry from nearby farms, more than 60,000 tonnes of slurries are processed per year. After being processed by the plant it becomes fertilizer (see Fig. 1).

In the plant a lot of variables are measured to control the process. Therefore, 211 variables are selected from the cogeneration process (from engines, the steam turbine and the recovery boiler) and also from the slurry drying process. These variables are mainly temperatures, levels, pressures, flows and powers. Further study is used to decide which variables are used in the models. On the other hand, the values of the variables are measured every minute and are stored in a database.

Data mining

Data mining is the process of discovering in databases interesting and useful information. In this first approach, a dataset with 211 variables has been used from the whole cogeneration process (engines, steam turbine, exhaust steam boiler, slurry process, etc.) collected in a 15-day period. During this period, the values of each variable were read every minute, obtaining nearly 18,000 samples. Due to the large number of variables and the complexity of the process, the work has started with two separate subsystems: one for an engine and another one for the steam turbine.

From all dataset variables (211), the number of variables read from the engine is large enough (40), so the rest of the process variables are not considered for the engine model. However, the number of read variables from the steam

turbine is only four, so initially all the process variables have been taken into account.

Firstly, a data mining process has carried out using Rapid Miner® software [14]. Each variable is represented to analyze its behaviour, to detect anomalous data and remove a non-informative variable (e.g. constant variables, zero variables or redundant variables). After this, the objective variable is chosen for each subsystem: for the engine the gas flow to feed it (to minimize the gas flow) and for the steam turbine, the power generated (to maximize it). To reduce the number of variables, Covariance Matrix and Principal Component Analysis (PCA) are applied to each subsystem. A Covariance Matrix is a matrix whose elements give the covariance (a measure of how much two random variables change together) between all the variables. Principal Component Analysis uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components [15].

In particular, the most related variables with a total of 85 % of covariance with the objective variable are selected. As a result, a total of 10 variables are chosen for the engine (5 temperatures, 1 pressure, 2 valves position, 1 power and the natural gas flow), and 8 variables for the steam turbine (2 temperatures, 2 flows, 2 pressures, 1 frequency of rotation and the power generated in the steam turbine), including the objective variable in both cases. Finally, bad data such as outliers or missing data are removed using outlier detection, filters and smoothing of signal.

Algorithms

Computational Intelligence (or Soft Computing) comprises a variety of techniques or methods for dealing with problems which are very hard to solve with standard methods, due to complexity, non-linearities, high dimensionality, the absence of an analytical model, etc. CI techniques are able to handle information that is imprecise, vague, incomplete or ambiguous. The most representative methods in CI are fuzzy systems, able to perform imprecise reasoning; neural networks, which can learn from samples; and the genetic algorithms, which make a search in the space of solutions. Furthermore, many systems combine two or more of these techniques such as, for example, neuro-fuzzy systems. CI techniques have been used successfully in problems where ordinary tools fail to provide a solution [16, 17].

Neural networks

Inspired by biological nervous system, ANNs are based on a large number of interconnected processing elements called neurons which are structured into layers (i.e. the input layer, one or more hidden layers and output layer). In this study, one of the most common ANN has been used: the Multiple Layer Perceptron (MLP). In the MLP a neuron in a particular layer is connected to all neurons in the next layer as Fig. 2 shows.

A neuron is a single computational element in the network that receives one or more inputs (x_j) either from a neuron in previous layer or (when it is situated in the input layer) from the outside world (input variables read from the process) are multiplied by weights (ω_{ij}) and then added together. After that, they are passed through to an activation function (φ), making the activation of the neuron continuously valued. Figure 3 provides the output of the neuron, where n is the number of inputs.

$$o_j = \varphi \left(\sum_{j=1}^{n} w_{ij} x_j \right) \tag{1}$$

When the neuron fires, this means that its output has a non-zero value and the output of the neuron is transmitted to neurons in the next layer. When a neuron is situated in the output layer, its output is (along with other outputs from neurons in the same layer) the output of the network. From the structure of a neural network it can be stated that the information is stored in the weights.

Neural networks can be trained by means of a collection of sample data using different learning algorithms, like for example, the gradient descend method (GDM). During the training process the weights are updated in such a way that the square sum of the difference between real output (desired output) and the output of the network is minimized. Equation (2) shows how a weight is updated in every iteration of the GDM algorithm, where $E = \sum_{i=1}^{k} (y_i - y'_i)^2$ is the mean square error, y_i and y'_i are respectively the actual and the desired ith outputs and η denotes a *learning rate*.

$$w_{ij}^k = w_{ij}^{k-1} - \eta \frac{\mathrm{d}E}{\mathrm{d}w_{ij}} \tag{2}$$

Fuzzy systems

Fuzzy systems are based on the fuzzy sets theory proposed by Zadeh [18] in the 1960s. They are systems which are able to deal with imprecision, vagueness or incomplete information. They have a rule inference mechanism where the rules are of the IF-THEN type:

IF x_1 is A_1 AND x_2 is A_2 ... AND x_n is A_n THEN y is C

where $x_1, x_2 . . x_n$ are the inputs, $A_1, A_2 . . A_n$ are linguistic values (e.g. 'low', 'medium', 'high') defined by fuzzy sets, y is the output and C is defined as a crisp value, called consequent.

Fig. 2 Example of a MLP neural network with a single hidden layer
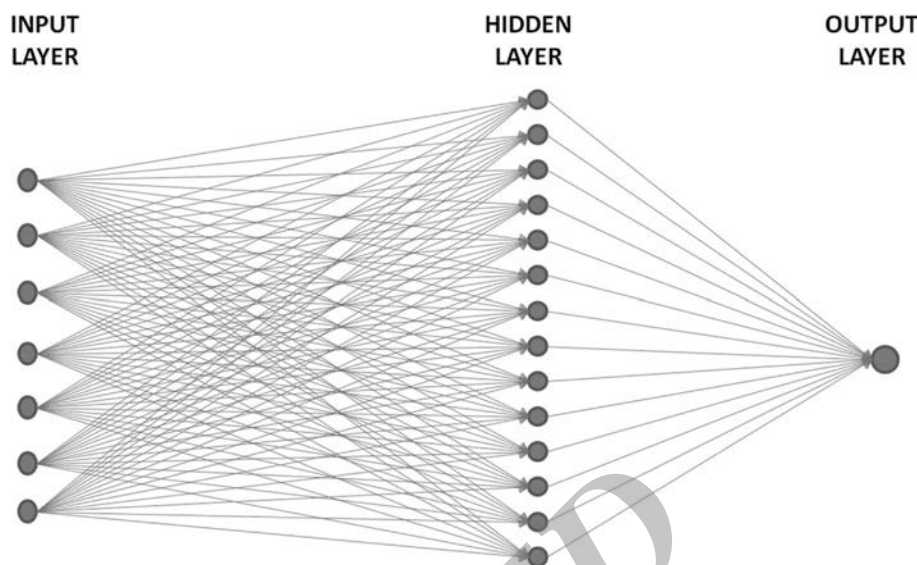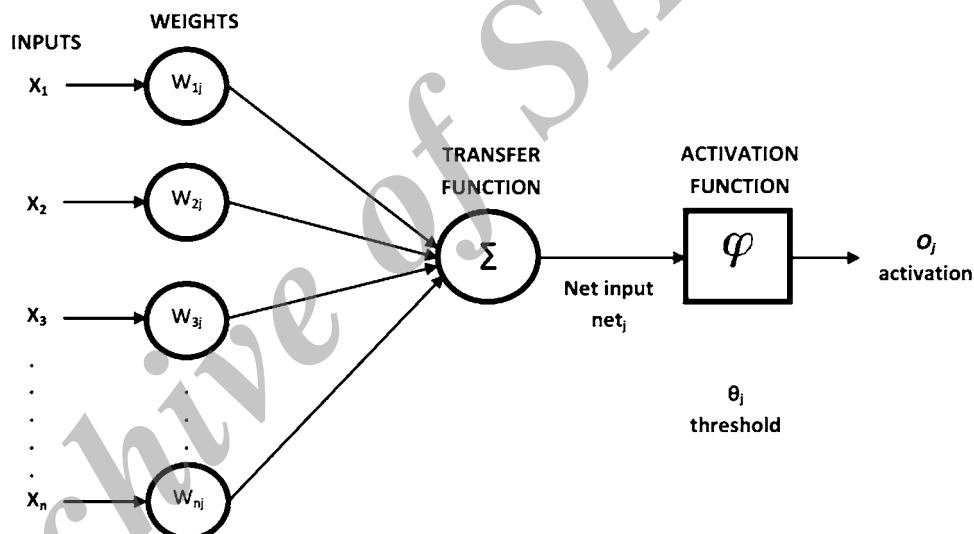

Fig. 3 Performance of a neuron

The output of the system is provided by computing Eq. (3), being r the number of rules of the system and $w_j = \prod_{i=1}^{n} A_i^j(x_i)$ the activation of the jth rule providing that the product is the operation for the AND connector.

$$y = \frac{\sum_j^r w_j c_j}{\sum_j^r w_j} \tag{3}$$

The centre of gravity method is adopted to transform fuzzy sets into crisp values (i.e. defuzzification algorithm).

In Fig. 4 an example of the inference mechanism is shown. The advantage of these systems is that they can represent knowledge in a linguistic way. Therefore it can be easily interpreted by humans. However, FSs have no way of learning or acquiring the knowledge.

Neuro-fuzzy

The aim of NF is to combine the advantages of both approaches: neural networks and fuzzy systems. The knowledge about the system is expressed as linguistic fuzzy relationships and the learning schemes of neural networks, capable of learning non-linear mappings of numerical data, are used to train the system. In addition, a neuro-fuzzy system is capable of extracting fuzzy knowledge from numerical data. In particular, in this work, the well-known adaptive neuro-fuzzy inference system (AN-FIS) has been used [19], which is trained with a hybrid learning algorithm composed by a GDM process and a least squared estimator (LSE) process.
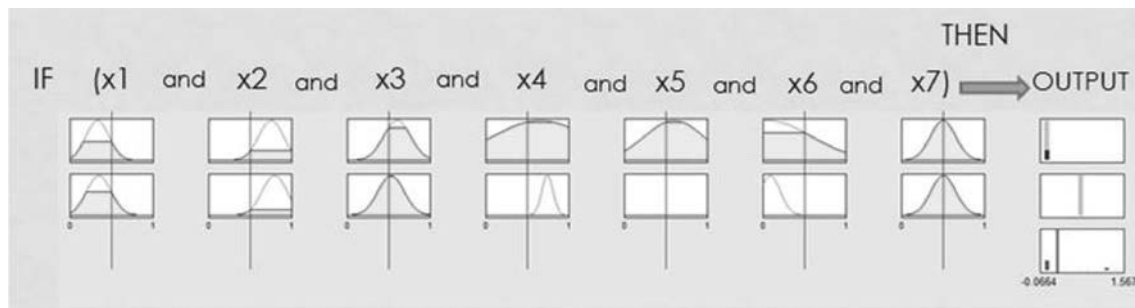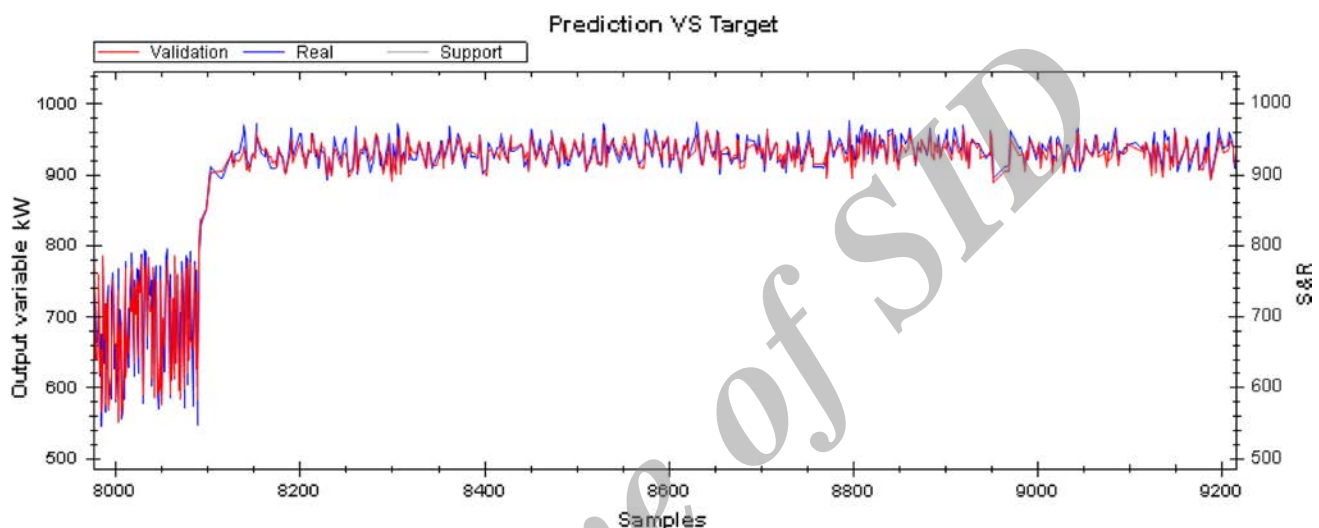
Fig. 4 Fuzzy rule scheme

Fig. 5 Real power vs. power for the steam turbine model using ANN

Modelization

For each subsystem, i.e. the steam turbine and one of the four engines, two CI strategies are used: one based on ANN and the other based on NF. All the experiments, modelization and recommendations are carried out with Optibat® Trainer tool [20].

Neural network

In both steam turbine and engine, some parameters related with the algorithm are initially fixed by means of different preliminary experiments. Subsequently, the following architectures are chosen:

In *the steam turbine* ANN model, the number of inputs selected is seven, as explained in "Data mining". The output variable is the power generated by the steam turbine, in kW; the objective is to maximize it. The number of hidden layers selected is only one because different experiments showed good results, and hence, the simplest option is selected. The number of neurons in the hidden layer is 14, because this is enough to generate a good model. Note that the larger the number of neurons, the longer the time required to train the ANN.

In the *engine* ANN model, the number of inputs selected is nine, as explained in "Data mining". The output variable chosen is the natural gas flow feeding the engine, in Nm^3/h; the objective is to minimize it. The engine is operating at nominal load as a system constraint. Therefore the variability of the variables is small and the learning process is more difficult. This means that the number of input variables is larger than in the steam turbine and the complexity of the architecture of the neural network is greater too. Different experiments, with different numbers of hidden layers and with different number of neurons in the hidden layers, are made. The best model is selected. The number of hidden layers is 3 and the number of neurons in the hidden layers is 10.

After training, satisfactory models in both subsystems are obtained, with a 0.94 % error (mean square error) in the steam turbine model (see Fig. 5) and a 2.75 % error in the
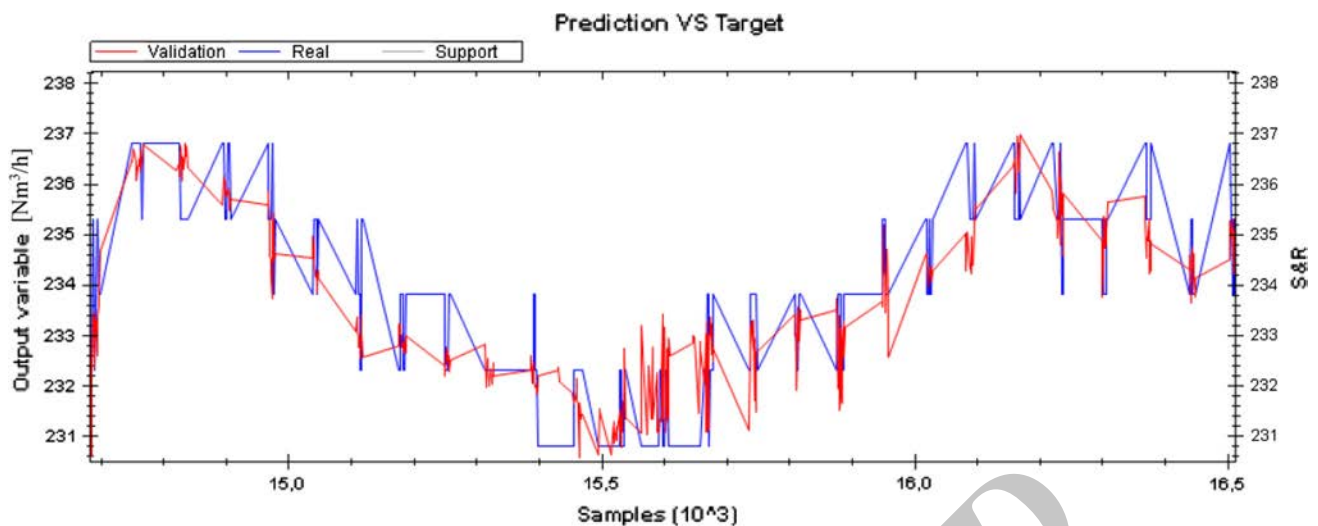
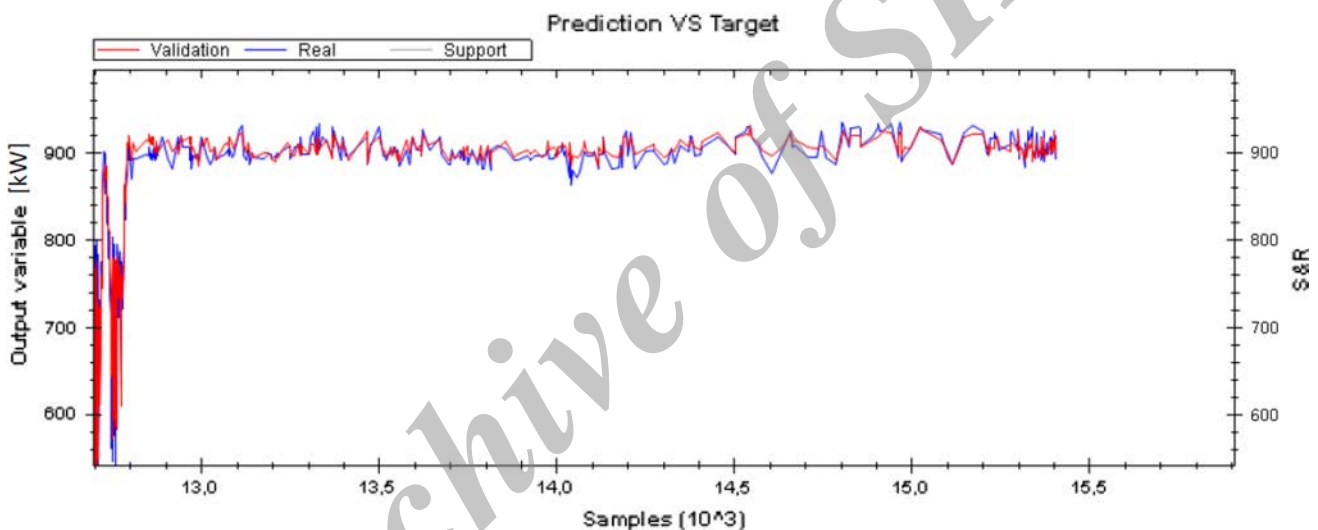**Fig. 6** Real gas flow vs. gas flow for the engine model using ANN

Fig. 7 Power real vs. power for the steam turbine model using ANFIS

engine (see Fig. 6). The engine model error is higher than the steam turbine model due to the load restrictions.

In both models the blue lines show the real output and the red lines show the model output for the steam turbine and for the engine, respectively.

ANFIS

As in the ANN, some parameters related with the algorithm are fixed after some previous experiments. After that, different models are created by changing some characteristics of the ANFIS structure. The best models for the engine and the steam turbine created with the fuzzy rules are selected and subsequently train.

The best steam turbine model and the best engine model are chosen. The steam turbine model error is 0.94 % with 20

fuzzy rules (Fig. 7), and the engine model error is 4.71 % with 25 fuzzy rules generated (Fig. 8). Consequently, the results show that ANN provides better results for the engine with a lower error than NF. On the other hand, the error for the steam turbine is the same with both algorithms.

As in the neural network graphics, the blue lines show the real output values and the red lines show the model output values.

Recommendations and results

The dependence between the input variables and the objective variables is studied to change some input variables to improve the global performance; i.e. maximize power generated by the steam turbine and minimize the gas
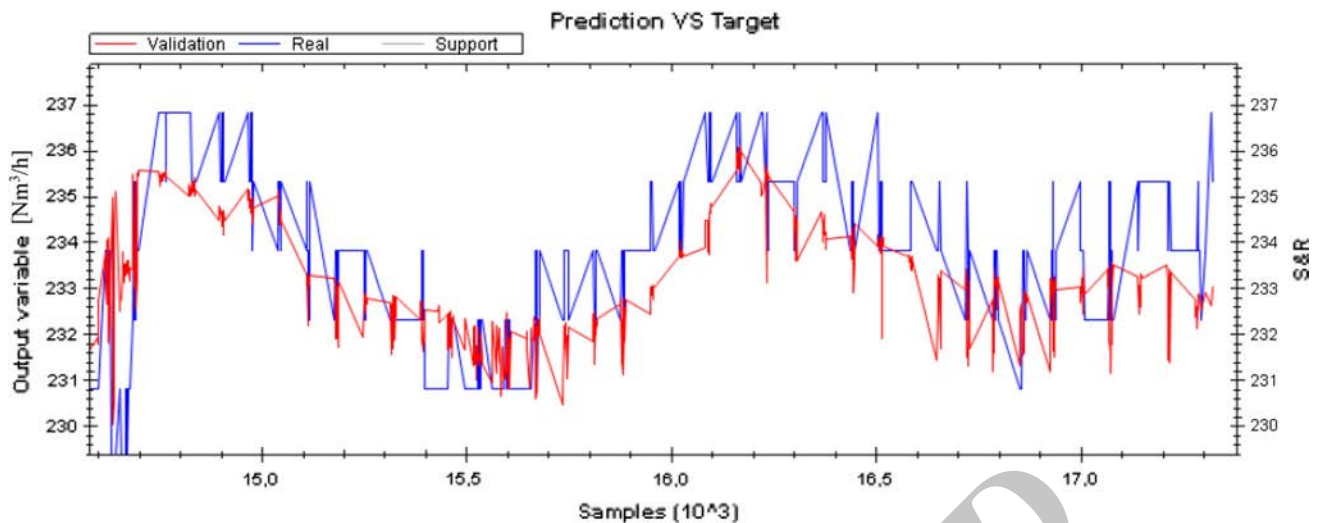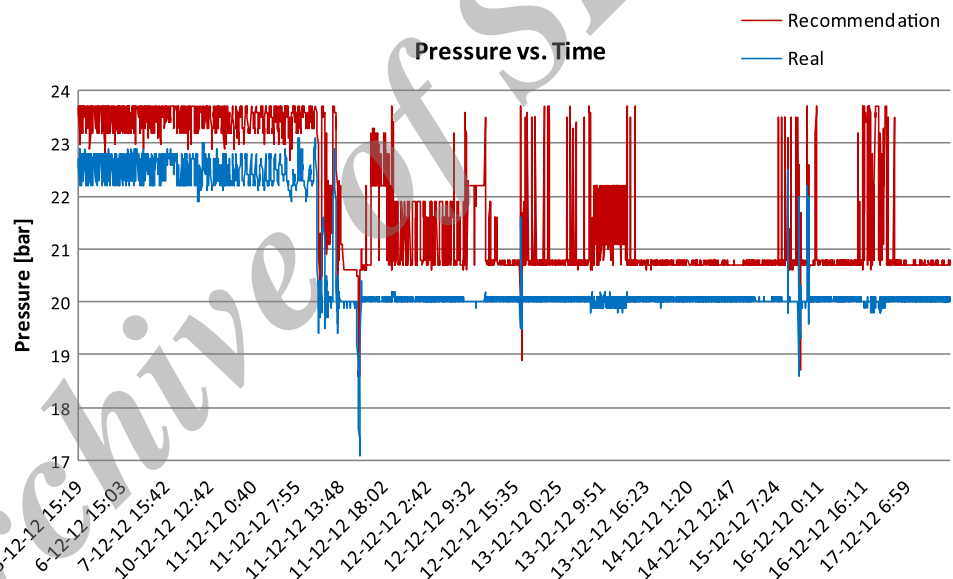
Fig. 8 Real gas flow vs. gas flow for the engine model using ANFIS

Fig. 9 Real values and recommended values for the steam turbine pressure vs. time

flow to feed the engine. As a result of this, some input variables are selected as operative variables. Different experiments are also carried out using one or more operative variables and saving of gas in the engine models and the increase of generated power in the steam turbine models are calculated for the models created with both algorithms: ANN and NF. Below, an example for each subsystem is shown.

The *steam turbine example* uses the NF model. The pressure is selected as an operative variable. Then, the recommendations and the results of these are calculated. In Fig. 9, the graphic shows the real pressure (blue line) and the recommended pressure in the steam turbine (red line). In Fig. 10 the graphic shows the real generated power (blue

line), and that achieved by applying the recommended pressure (red line).

Note that in Fig. 10, the power generated in the steam turbine generally is higher applying the recommendations for the operative variable, i.e. the pressure of the steam turbine. Even the worst recommendations generated the same power as that without recommendations.

Optimization of both subsystems is achieved with recommendations for the objective variables. In the steam turbine, the goal is to maximize the generated power, and the best recommendations generate a 7.62 % average of generated power increase (see Fig. 11).

The *engine example* used the ANN model. The engine load restrictions imply that the variables cannot

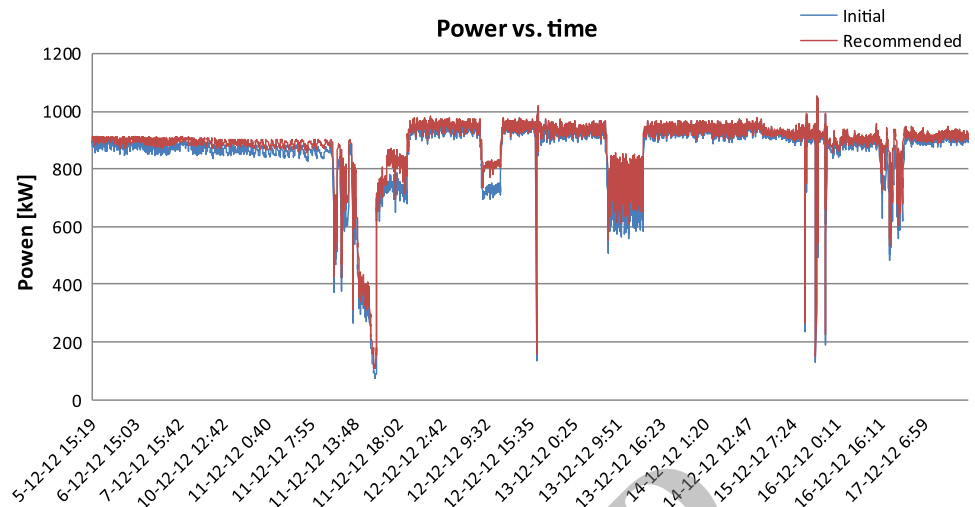**Fig. 10** Power generated with and without recommendations vs. time

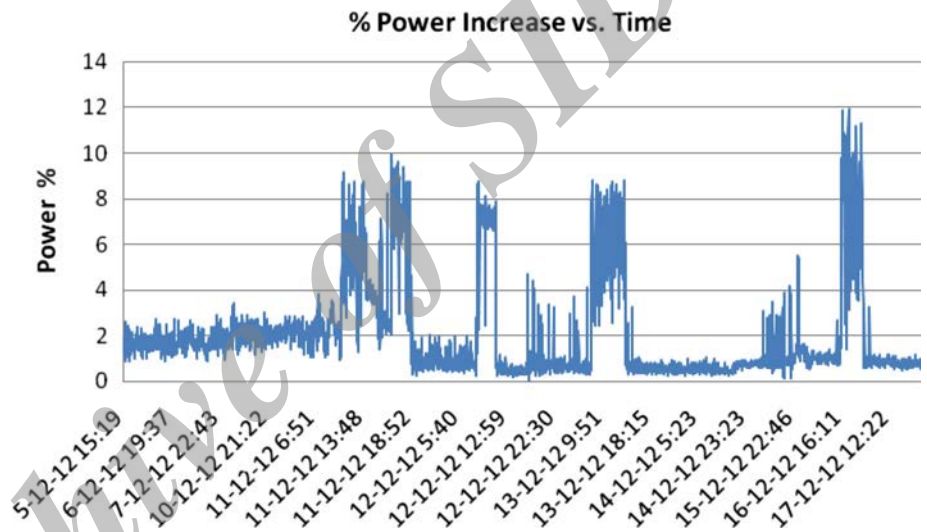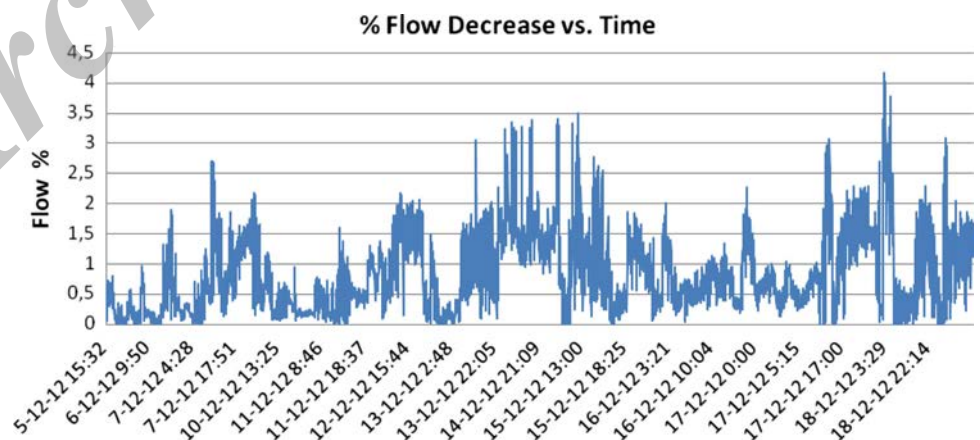**Fig. 11** Increase of power generated with recommendations vs. time

Fig. 12 Decrease of gas flow generated with recommendations vs. time

significantly change, and as a result, the goal of minimizing the natural gas flow is hardly achieved. Nevertheless, after different experiments it is possible to generate recommendations that provide a decrease of 0.83 % gas flow (see Fig. 12). It happens when the

selected operative variable is the admission temperature of the air.

For this example, Fig. 13 shows the real temperature (blue line) and the recommended temperature (red line) in the engine. In Fig. 14, the graphic shows the real natural gas used

Fig. 13 Real values and recommended values for the Engine admission temperature vs. time
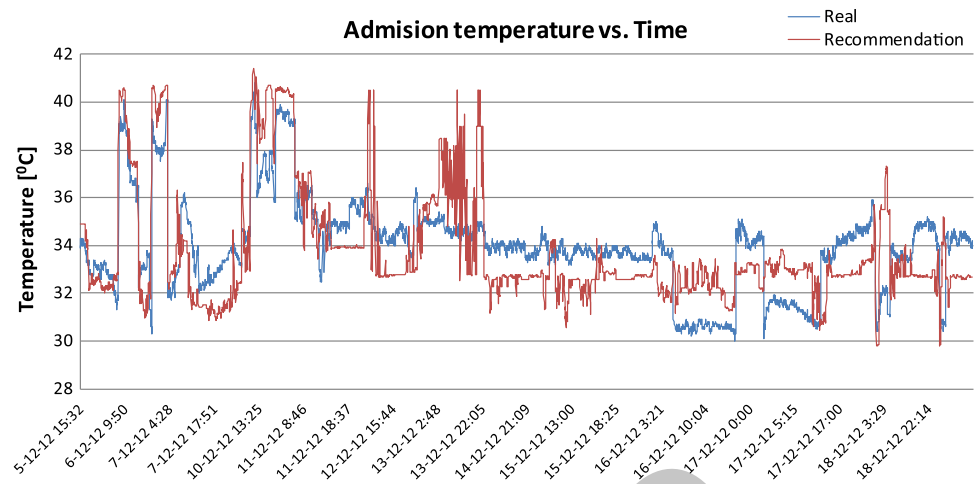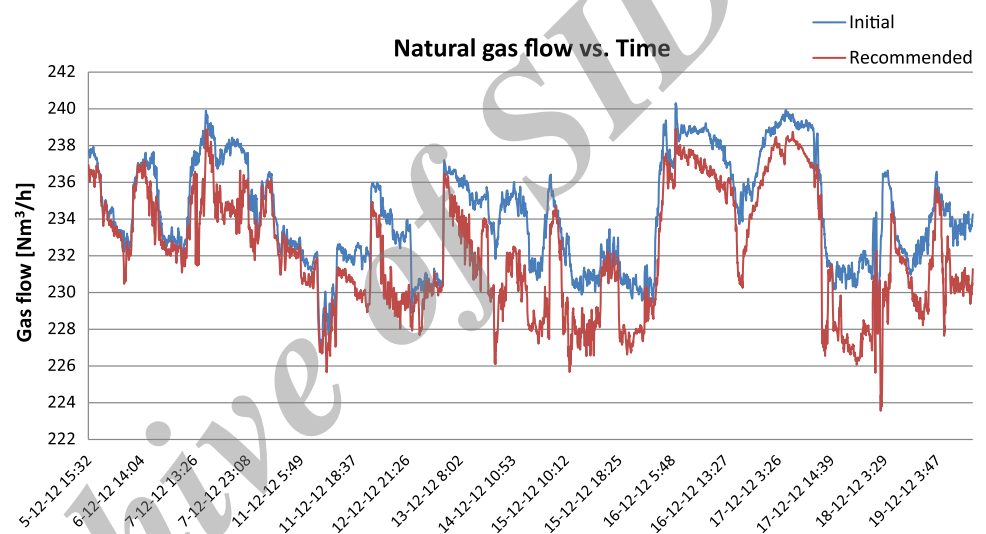

Fig. 14 Flow used with and without recommendations vs. time

(blue line), and that achieved by applying the recommended temperature (red line). The natural gas used to feed the engine generally is less applying the recommendations, but it is never higher. As in the steam turbine the worst recommendations generated the same natural gas consumption.

Although 0.83 % is a small percentage, it becomes a good value dealing with very big engines with high fuel consumption.

Therefore, the results show how the recommendations applied to the models improve the behaviour of the plant. It is possible to provide higher energy production, with a higher power generated in the steam turbine, with a lower cost, and using less natural gas in the engine.

Conclusions

In the present study models with both algorithms, ANN and ANFIS, are created from two subsystems of a cogeneration plant, one of the four engines and the steam turbine. To obtain patterns from the dataset, data mining techniques are used. Models are used to generate recommendations in some input variables to optimize the subsystems. The main findings of the work are

- The models for the steam turbine and the engine show very good modelling performance with both an ANN and an ANFIS system. An ANFIS advantage against ANN is that the model provides us with linguistic information about the system, but ANN provides better results for the engine.
- In the engine model, recommendations decrease the gas flow to feed it and for the steam turbine, recommendations increase the power generated. This implies an optimization which means fuel savings and a higher power production.

Further work considers modelling of the whole plant for maximum energy efficiency of the cogeneration process.

References

1. The European Association for the Promotion of Cogeneration. http://www.cogeneurope.eu/. Accessed 10 Jul 2013
2. Villares Martín, M.: Cogeneration. FC Editorial, 13 (2003)
3. Wakui, T., Yokoyama, R.: Optimal structural design of residential cogeneration systems in consideration of their operating restrictions. Energy J. **64**, 719–733 (2014)
4. Skolnik, P., Hubka, L., Modrlak, O., Nahlovsky, T.: Cogeneration units simulation models library. In: IEEE 2013 International Conference on Process Control (PC), pp. 252–256 (2013)
5. Wu, L., Hu, Y., Gao, C.: Optimum design of cogeneration for power and desalination to satisfy the demand of water and power. Desalin. J. **324**, 111–117 (2013)
6. Moghavvemi, M., Yang, S.S., Kashem, M.A.: A practical neural network approach for power generation automation. In: IEEE Proceedings of International Conference on Energy Management and Power Delivery, pp. 305–310 (1998)
7. Tamiru, A.L., Rangkut, C., Hashim, F.M.: Neuro-fuzzy and PSO based model for the steam and cooling sections of a cogeneration and cooling plant (CCP). In: Proceedings of ICEE 2009 3rd International Conference on Energy and Environment, pp. 27–33 (2009)
8. Kim, D.H.: Neuro-fuzzy tuning of PID controller for control of actual gas turbine power. In: IEEE International Conference on Computational Intelligence for Measurements Systems and Applications, pp. 192–197 (2004)
9. Barelli, L., Bidini, G.: Design of the measurements validation procedure and the expert system architecture for a cogeneration internal combustion engine. Appl. Therm. Eng. **25**, 2698–2714 (2005)
10. Repin, A.I., Sabanin, V.R., Smirnov, N.I., Andreev, S.N.: Diagnostic of the information subsystem of an automatic process control system using artificial intelligence. Therm. Eng. **53**, 476–482 (2006)
11. Chitashvili, G.P.: A proportional method for calculating the efficiency and specific consumption of fuel at gas-turbine cogeneration stations. Therm. Eng. **53**, 983–988 (2006)
12. Xiao-li, H., Guo-qiang, Z.: Maximum useful energy rate and efficiency of a recuperative Brayton cogeneration plant. J. Cent. South Univ. Technol. **20**, 156–163 (2013)
13. Zomorodian, R.A., Rezasoltani, M.B., Ghofrani, M.B.B.: Static and dynamic neural networks for simulation and optimization of cogeneration systems. Int. J. Energy Environ. Eng. **4**, 51–61 (2011)
14. Rapid Miner®. http://rapid-i.com/. Accessed 15 Jul 2013
15. Neil, T.: Applied Multivariate Analysis, pp. 80–83. Springer, NY (2002). 445–453
16. Eberhart, R., Simpson, P., Dobbins, R.: Computational Intelligence PC Tools. Academic Press, Boston (1996)
17. Jang, J.S.R., Sun, C.T., Mizutani, E.: Neuro-Fuzzy and Soft Computing. Prentice Hall, NY (1997)
18. Mendel, J.M.: Fuzzy logic systems for engineering: a tutorial. Proc. IEEE **83**(3), 345–377 (1995)
19. Roffel, B., Betlem, B.: Process Dynamics and Control, Modeling for control and Prediction, pp. 361–399. Wiley, London (2006)
20. Optibat Trainer®. http://www.optimitive.com/en/. Accessed 21 Jul 2013