Applied Soft Computing 48 (2016) 347–358

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

A neuro-genetic approach for modeling and optimizing a complex cogeneration process

M.A. Braun[c,*], S. Seijo[a], J. Echanobe[a], P.K. Shukla[c], I. del Campo[a], J. Garcia-Sedano[b], H. Schmeck[c]

[a] *Department of Electricity and Electronics, UPV/EHU, Spain*
[b] *Optimitive, Vitoria, Spain*
[c] *Institute AIFB, KIT, Karlsruhe, Germany*

ARTICLE INFO

Article history:
Received 7 March 2016
Received in revised form 12 July 2016
Accepted 13 July 2016
Available online 21 July 2016

Keywords:
Neural networks
ESPEA
Genetic algorithm
Cogeneration
Multiobjective optimization

ABSTRACT

Cogeneration is the simultaneous generation of electricity and useful heat with the aim of exploiting more efficiently the energy stored in the fuel. Cogeneration is, however, a complex process that encompasses a great amount of sub-systems and variables. This fact makes it very difficult to obtain an analytical model of the whole plant, and therefore providing a mechanism or a methodology able to optimize the global behavior. This paper proposes a neuro-genetic strategy for modeling and optimizing a cogeneration process of a real industrial plant. Firstly, the modeling of the process is carried out by means of several interconnected neural networks where, each neural network deals with a particular sub-system of the plant. Next, the obtained models are used by a genetic algorithm, which solves a multiobjective optimization problem of the plant, where the goal is to minimize the fuel consumption and maximize both the generated electricity and the use of the heat. The proposed approach is evaluated with data of a real cogeneration plant collected over a one-year period. Obtained results show not only that the modeling of the plant is correct but also that the optimization increases significantly the efficiency of the cogeneration plant.

1. Introduction

The process of generating electricity and useful heat at the same time is called cogeneration and is also known as combined heat and power (CHP). The ultimate goal of cogeneration is to exploit the maximum possible energy contained in a fuel. In the industry, the high temperature flue gases generated by engines, gas turbines, or other machines can be used to produce more electricity or to perform another process demanding heat. This implies cost savings because the amount of fuel required is reduced. This fuel saving also results in a reduction of pollution. These economic and environmental factors are the reasons why nowadays the number of cogeneration plants is increasing steadily.

As many other industrial processes, CHP is a rather complex system due to a high number of variables involved, non-linear dynamics, limited analytical models and also incomplete knowledge. This fact implies that it is very troublesome to obtain a model that reproduces with fidelity the behavior of the real system. Moreover, without such a model, it becomes very difficult to carry out any formal strategy to try to optimize, in some sense, the efficiency of the process.

Soft computing (SC) algorithms provide a non-conventional way to deal with those problems characterized by their complexity, high dimensionality, hard non-linearities and vague or imprecise knowledge. Most typical soft computing algorithms are neural networks (NN), fuzzy systems (FS) and evolutionary computation (EC). Many of these techniques exhibit complementary aspects and hence, they provide very often better performance when combined in a cooperative way rather than acting exclusively (e.g., neuro-fuzzy (NF) systems, evolutionary fuzzy (EF) systems, or neuro-evolutionary (NE) systems).

Due to those interesting properties, SC methods are widely used for modeling different industrial processes, for example, in water-treatment [1], in steel-making [2] and paper-making industries [3], or modeling boilers [4,5], among others. In addition, they are also very useful for detecting and predicting faults in industrial processes [6,7] and in engines [8–10].

SC methods are also widely used in generation plants mainly for analysis/diagnosis, optimization, control or prediction purposes. Below is a summary of the more recent works, in which SC

* Corresponding author.
E-mail address: marlon.braun@kit.edu (M.A. Braun).

http://dx.doi.org/10.1016/j.asoc.2016.07.026

Nomenclature

CHP	combined heat and power
Div_A	diverter Engine A (%)
Div_B	diverter Engine B (%)
Div_C	diverter Engine C (%)
Div_D	diverter Engine D (%)
F_{Cond}	condensate effluent flow (kg/h)
F_{Ev}	flow fed evaporator (kg/h)
$F_{FlueGas}$	flue gases Flow (kg/h)
F_{Gas_A}	flow natural gas Engine A (m^3/h)
F_{Gas_B}	flow natural gas Engine B (m^3/h)
F_{Gas_C}	flow natural gas Engine C (m^3/h)
F_{Gas_D}	flow natural gas Engine D (m^3/h)
F_{Steam}	Steam flow to steam turbine (kg/h)
H_{Amb}	ambient humidity (%)
LHV	low heating value (kWh/m^3)
MLP	multilayer perceptron
NN	artificial neural networks
P_{Cond}	condenser pressure (bar)
P_{Ev}	evaporator pressure (bar)
POW_A	rated power Engine A (%)
POW_B	rated power Engine B (%)
POW_C	rated power Engine C (%)
POW_D	rated power Engine D (%)
POW_{ST}	turbine power (kW)
POW	total generated power
P_{St_Gen}	steam generator pressure (bar)
T_{B1_A}	temperature intake air Bank1 Engine A (°C)
T_{B2_A}	temperature intake air Bank2 Engine A (°C)
T_{B1_B}	temperature intake air Bank1 Engine B (°C)
T_{B2_B}	temperature intake air Bank2 Engine B (°C)
T_{B1_C}	temperature intake air Bank1 Engine C (°C)
T_{B2_C}	temperature intake air Bank2 Engine C (°C)
T_{B1_D}	temperature intake air Bank1 Engine D (°C)
T_{B2_D}	temperature intake air Bank2 Engine D (°C)
T_{Bank1_A}	temperature gases Bank1 Engine A (°C)
T_{Bank2_A}	temperature gases Bank2 Engine A (°C)
T_{Bank1_B}	temperature gases Bank1 Engine B (°C)
T_{Bank2_B}	temperature gases Bank2 Engine B (°C)
T_{Bank1_C}	temperature gases Bank1 Engine C (°C)
T_{Bank2_C}	temperature gases Bank2 Engine C (°C)
T_{Bank1_D}	temperature gases Bank1 Engine D (°C)
T_{Bank2_D}	temperature gases Bank2 Engine D (°C)
$T_{H_2O\_Ex}$	water temperature exchange (°C)
$T_{H_2O\_SH}$	water superheated temperature (°C)
$T_{H_2O\_TH}$	water temperature tubular heater (°C)
$T_{H_2O\_Tow}$	water temperature cooling tower (°C)
T_{Mixt_EngA}	water temperature to cooling the mixture Engine A (°C)
T_{Mixt_EngB}	water temperature to cooling the mixture Engine B (°C)
T_{Mixt_EngC}	water temperature to cooling the mixture Engine C (°C)
T_{Mixt_EngD}	water temperature to cooling the mixture Engine D (°C)

algorithms are applied in some way in CHP plants or in other similar thermal power plants.

- **NNs** are proposed in many papers for modeling one or several aspects of a CHP process, with different purposes (see [11] for an exhaustive bibliography revision). Some of these purposes are: (1) predicting or monitoring the power generated [12–15], the thermal efficiency and the pollutant emission [16,17] or the base-line energy [11]. (2) Reproducing the behavior of some plant components [18]. (3) Design or optimization of adaptive load-shedding models for stability [19]. (4) Optimization of operating parameters for plant efficiency maximization [20,21]. (5) Design of controllers [22,23].
- **EC** is also utilized in many papers for optimizing different aspects of thermal power plants. A standard genetic algorithm (GA) is employed in [24] to optimize the start-up operation of a combined cycled power plant by means of a single objective function. A short optimization, also using a standard GA, is carried out in [25] for a steam power plant in which the efficiency is maximized in terms of cost and exergy. The cost of electricity generated in a combined cycle power plant is carried out in [26] by using a GA. Hajabdollahi et al. [27] use a non-dominated sorting genetic algorithm (NSGA-II) in a two-objective optimization problem (i.e., efficiency and cost) in a steam cycle power plant. Ahmadi and Dincer [28] also use NSGA-II to tackle a two-objective function optimization (i.e., maximization of efficiency and the minimization of environmental impact) in a turbine power plant. Deb et al. [29] solve a four-objective optimization problem of a solar thermal electricity plant using NSGA-II. Basu [30] proposes to use NSGA-II for two objectives (economic and environmental) in a hydrothermal power system.
- **FS**. FS-based techniques have also been used in thermal power plants. A fault tolerant measurement system based on Takagi–Sugeno fuzzy models for a gas turbine in a combined cycle power plant is proposed in [31]. A Takagi–Sugeno fuzzy model of a complex parallel flow heat exchanger of a thermal plant using a fuzzy clustering technique is presented in [32]. Mazur [33] carries out a two-objective optimization by means of a fuzzy logic for thermo-economic analysis of energy-transforming systems. Rodriguez-Martinez et al. [34] propose a fuzzy controller to govern the speed response of a gas turbine power plant during startup. Also, Moon and Lee [35] propose an adaptive algorithm for dynamic matrix control (DMC) using fuzzy inference, and present its application to a drum-type boiler-turbine system in a fossil power plant. The control of a steam turbo-generators by a fully automatic fuzzy-based algorithm is presented in [36].

Regarding the SC hybrid algorithms, there exist also different papers in the CHP/thermal plants context.

- **NN + FS:** Liu et al. [37] address the modeling of an ultra super-critical boiler system in a 1000 MW power plant, using fuzzy neural network methods for controlling purposes. A neuro-fuzzy strategy is used in [38] for developing a diagnostic procedure of a cogeneration plant. In particular, the authors use NNs to model the internal combustion engines and then an FS is designed to analyze the NN outputs to detect probable system failure. An adaptive neuro-fuzzy system (i.e., ANFIS) is proposed in [39] to model the technological processes of a CHP plant.
- **NN + EC**: A procedure based on NN and artificial bees colony (ABC) is proposed to maximize the efficiency of a regenerative Rankine cycle with two feedwater heaters in [40]. Also, Suresh et al. [41] present an NN-GA based method to optimize the efficiency of a high ash coal-fired super-critical power plant.
- **EC + FS**. A particle swarm optimization (PSO) and a fuzzy decision-making system are proposed in [42] to optimize a benchmark cogeneration system. In particular, the PSO performs a three-objective optimization process and then a final optimal solution of the Pareto front is selected using the fuzzy system. A swarm intelligence fuzzy clustering technique is used in [43] to obtain a Takagi-Sugeno fuzzy model with enhanced performance of superheated steam temperature in a power plant. Saez et al.

Table 1
SC applied to cogeneration.

Authors	Complete plant	Real data	SC to model	SC to optimize	Multi-objective	On-line operation
De et al. [12]	×	×	√	×	×	×
Smrekar et al. [13]	√	√	√	×	×	×
Nikpey et al. [14]	×	√	√	×	×	×
Sisworahardjo and El-Sharkh [15]	×	×	√	×	×	×
Flynn et al. [16]	√	√	√	×	×	×
Pan et al. [17]	√	√	√	×	×	×
Rossi et al. [11]	√	√	√	×	×	×
Bekat et al. [18]	×	√	√	×	×	×
Kirar and Agnihotri [19]	√	×	√	×	×	×
Zomorodian et al. [20]	√	×	√	√	×	×
Wang et al. [22]	×	√	√	×	×	×
Lee et al. [23]	√	√	√	×	×	√
Bertini et al. [24]	√	×	×	√	×	×
Ameri et al. [25]	√	×	×	√	×	×
Koch et al. [26]	√	×	×	√	×	×
Hajabdollahi et al. [27]	√	×	×	√	√	×
Ahmadi and Dincer [28]	√	√	×	√	√	×
Deb et al. [29]	√	×	×	√	√	×
Basu [30]	√	×	×	√	√	×
Berrios et al. [31]	√	√	√	×	×	×
Habbi et al. [32]	×	√	√	×	×	×
Mazur [33]	√	×	×	√	√	×
Moon and Lee [35]	√	×	√	×	×	×
Liu et al. [37]	√	√	√	×	×	×
Barelli and Bidini [38]	√	√	√	×	×	×
Mastacan et al. [39]	×	√	√	×	×	×
Rashidi et al. [40]	√	×	√	√	√	×
Suresh et al. [41]	√	×	√	√	×	√
Sayyaadi et al. [42]	√	×	×	√	√	×
Saez et al. [44]	×	×	√	√	×	×
Oh et al. [46]	×	√	√	×	×	×
Tamiru et al. [45]	×	√	√	×	×	×

[44] propose a fuzzy predictive supervisory controller, based on genetic algorithms (GA), for gas turbines of combined cycle units.

- **EC + FS + NN**. There even exist papers in which the three SC paradigms (i.e., NN, FS and EC) are used jointly in the CHP systems. A neuro-fuzzy system, whose parameters are trained by means of a PSO algorithm, is used in [45] for modeling the steam and cooling sections of a cogeneration and cooling plant (CCP). Oh et al. [46] present a neuro-fuzzy scheme, whose parameters are adjusted by means of a GA-based hybrid method, for modeling a turbine power plant.

All the SC algorithms in the papers cited above deal with the modeling of one or more components of the plant, and/or the optimization of certain parameters of the process. However, compared with the work presented here, they are incomplete regarding some of the following issues: (1) they do not apply SC algorithms to both modeling and optimization (in fact, these are the majority); (2) the modeling process is performed often for just some components and not for the complete plant; (3) in many cases they use simulation tools or thermodynamics equations and do not work with data from a real plant; (4) optimization is carried out for a single-objective function; (5) optimization is not intended for a continuous on-line operation. These aspects can be better appreciated in Table 1 where it is shown how none of the proposed methods covers all of the mentioned issues.

In this paper, we propose a global NN–EC strategy for modeling and optimizing a real CHP plant that covers all the above mentioned aspects. The modeling is made-up with several interconnected NNs, which are trained and tested with real data collected from the plant. Moreover, the modeling embraces all the process components: engines, intercoolers, steam condenser, boiler, turbine and slurry drying. The optimization process is carried out by means of a genetic algorithm and it is actually a multiobjective optimization problem, where the NNs are used to compute the values of the multiobjective function. More precisely, the goal is to simultaneously minimize the used fuel, maximize the produced electricity and maximize the useful thermal energy. In particular, we have used the ESPEA algorithm [47], which is a nondominated sorting type algorithm that is characterized by providing an excellent distribution of the individuals of the final population. In addition, the optimization process is fast enough to be applied in the continuous on-line process of the plant.

The rest of the paper is organized as follows: in Section 2 the CHP plant used in the paper is described in detail. Section 3 deals with the NN-based modeling of the entire plant. Section 4 presents the GA-based multiobjective optimization process. In both these sections, the experiments carried out and the obtained results are analyzed in detail. Finally, Section 5 presents the main conclusions of the work.

2. CHP plant

Fig. 1 depicts a schematic of the CHP plant used in this paper. The plant is located in Monzón (Huesca), in the north of Spain.[1] The main systems of the plant are: four internal combustion engines, four refrigeration engine circuits, an exhaust steam boiler, a steam turbine condenser, a steam turbine, and a slurry drying process. The plant produces electricity by means of the combustion engines and the steam turbine. The steam is generated with the heat contained in the exhaust gases of the four engines. Part of this heat is also used in a slurry drying process being the slurry provided by nearby farms.

The four internal combustion engines are all identical, with the same characteristics and the nominal power of each being 3700 kW. They are organized into two banks with eight cylinders each and the

[1] http://www.energyworks.com.

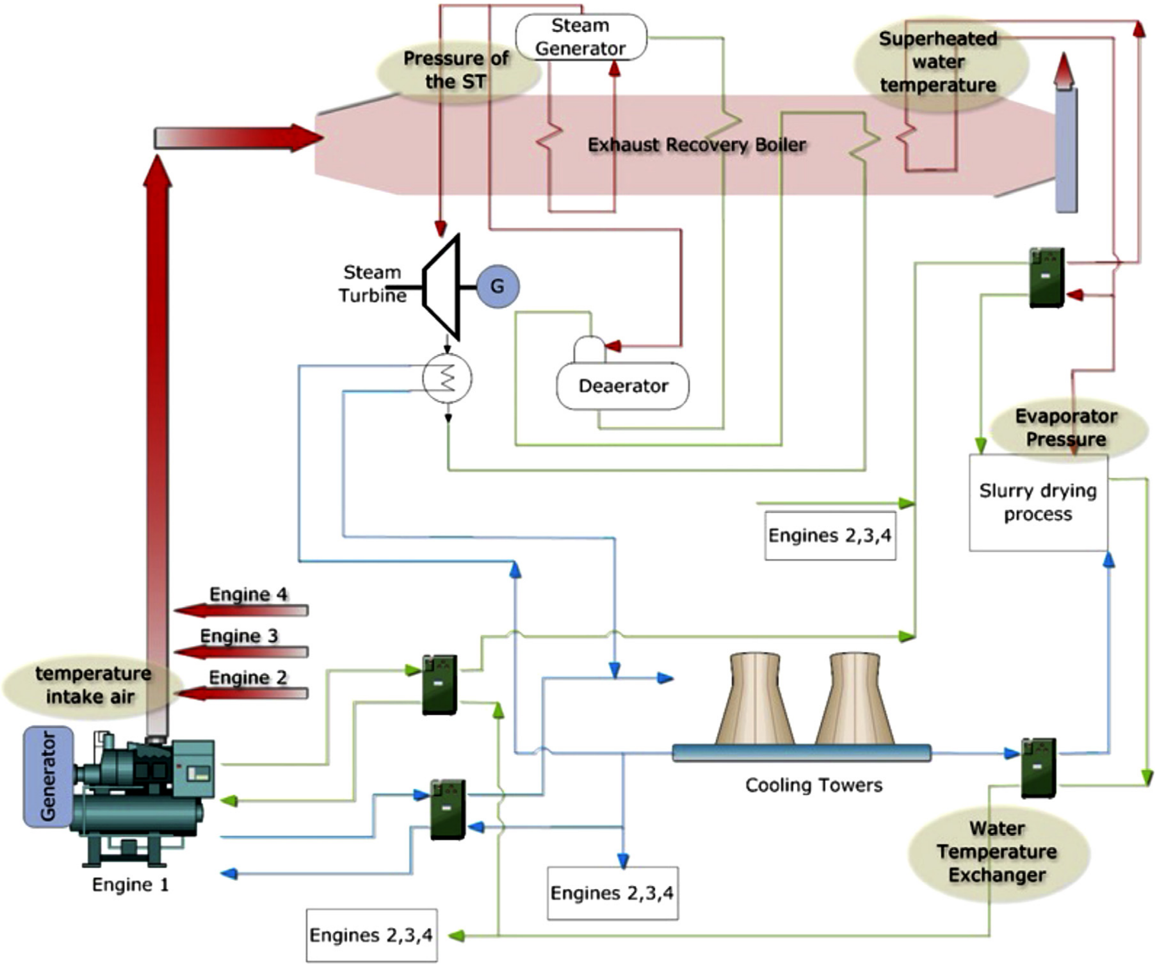

Fig. 1. Scheme of the combined heat and power process with their related equipment.

fuel used for the combustion is natural gas. The engines exchange heat with two circuits that use water from the cooling towers, as Fig. 1 shows. A cooling circuit refrigerates the mixture of air-fuel around 50 °C and the other circuit preheats the intake air to around 35 °C. The engines generate electrical energy, which is sold, and also flue gases. Each engine has a diverter which sends the flue gases to an exhaust steam boiler when the engine is working above 50% of rated power, or to the chimney if the rated power is below 50%. Engines are usually above this threshold, and therefore the flue gases go to the exhaust steam boiler most of the time.

Next, the heat from the exhaust steam boiler is used by the steam generator to create steam around 400 °C at 22.5 bar. This steam feeds the steam turbine to generate more electricity, with 1000 kW of nominal power. The condenser of the steam turbine uses water from the cooling towers to condensate the steam from the steam turbine and recirculate it to the system. In addition, as in the engines, the power generated with the steam turbine is also sold.

The slurry from the farms consists of 6% solids approximately. Firstly, a mechanical treatment is carried out to remove the solid part from the rest using rotatory equipment. Then, a chemical treatment in the liquid part is performed to remove the chemical load. After that, the heat treatment uses the result of the chemical treatment to separate the condensables from non-condensables in an evaporator using superheated water generated in the exhaust steam boiler (water with a temperature around 120 °C). A tubular heater is used to recirculate the effluent to the evaporator and preheat it. The tubular heater uses water from the refrigeration circuit, which preheats the intake air of the engines. The non-condensable part goes with the solid part resulting from the mechanical treatment and is sold as fertilizer. The condensable effluent is condensed again with the water from the cooling towers. Finally, the sterilizer uses the heat from the superheated water to purify the condensed effluent, thereby obtaining water suitable for irrigation.

3. Neural network modeling

As it has been stated in the introduction, the modeling of the CHP plant described in the previous section has been performed by means of NNs. In particular, we have adopted a divide-and-conquer strategy. That is, we have modeled each main component of the plant separately with an NN, and then, we have linked (connected) all the networks by means of shared or common variables (e.g., an output variable of a network can be an input variable of one or several other networks). The global model is shown in Table 2 where a total of twelve NNs are depicted: four for the engines, four for the engine cooling circuits, one for the exhaust steam boiler, one for the steam turbine condenser, one for the steam turbine and finally another one for the slurry drying process.

As neural network model, we have selected the well-known multilayer perceptron (MLP) due to its structural flexibility, good representational capabilities and availability of a large number of training algorithms (for example the Back-Propagation algorithm which is the one used here) [27]. In fact, MLP is an universal approximator and hence it is capable of approximating any measurable function to any desired degree of accuracy [48]. It is understood,

Table 2
CHP neural networks and their corresponding variables.

Neural network	Inputs of the model	Output
Cooling Engine A	$\mathrm{T}_{\mathrm{H_2O\_Ex}}$, $T_{H_2O\_TOW}$, POW_A	T_{Mixt_EngA}
Cooling Engine B	$\mathrm{T}_{\mathrm{H_2O\_Ex}}$, $T_{H_2O\_TOW}$, POW_B	T_{Mixt_EngB}
Cooling Engine C	$\mathrm{T}_{\mathrm{H_2O\_Ex}}$, $T_{H_2O\_TOW}$, POW_C	T_{Mixt_EngC}
Cooling Engine D	$\mathrm{T}_{\mathrm{H_2O\_Ex}}$, $T_{H_2O\_TOW}$, POW_D	T_{Mixt_EngD}
Engine A	$\mathbf{T}_{B1\_A}$, $\mathbf{T}_{B2\_A}$, T_{Amb}, H_{Amb}, LHV T_{Bank1_A}, T_{Bank2_A}, T_{Mixt_EngA}, POW_A, DIV_A	F_{Gas_A}
Engine B	$\mathbf{T}_{B1\_B}$, $\mathbf{T}_{B2\_B}$, T_{Amb}, H_{Amb}, LHV T_{Bank1_B}, T_{Bank2_B}, T_{Mixt_EngB}, POW_B, DIV_B	F_{Gas_B}
Engine C	$\mathbf{T}_{B1\_C}$, $\mathbf{T}_{B2\_C}$, T_{Amb}, H_{Amb}, LHV T_{Bank1_C}, T_{Bank2_C}, T_{Mixt_EngC}, POW_C, DIV_C	F_{Gas_C}
Engine D	$\mathbf{T}_{B1\_D}$, $\mathbf{T}_{B2\_D}$, T_{Amb}, H_{Amb}, LHV T_{Bank1_D}, T_{Bank2_D}, T_{Mixt_EngD}, POW_D, DIV_D	F_{Gas_D}
Exh. steam boiler	$\mathbf{P}_{StGen}$, $F_{FlueGas}$	F_{Steam}
Steam turbine condenser	$T_{H_2O\_TOW}$, T_{ST_Cond}	P_{Cond}
Steam turbine	P_{StGen}, F_{Steam}, P_{Cond}	POW_{ST}
Slurry process	P_{EV}, $\mathrm{T}_{\mathrm{H_2O\_SH}}$, $\mathrm{T}_{\mathrm{H_2O\_Ex}}$, F_{Cond}	F_{Ev}

Table 3
MAE for training and testing samples for each neural network.

	Structure	Training $MAE \times 100$	Testing $MAE \times 100$
Cooling Engine A	3/6/1	0.21%	0.23%
Cooling Engine B	3/6/1	0.28%	0.26%
Cooling Engine C	3/6/1	0.10%	0.13%
Cooling Engine D	3/6/1	0.49%	0.33%
Engine A	10/20/1	0.39%	0.42%
Engine B	10/20/1	0.41%	0.41%
Engine C	10/20/1	0.38%	0.42%
Engine D	10/20/1	0.38%	0.37%
Recovery boiler	2/4/1	0.61%	0.63%
Steam condenser	2/4/1	1.01%	0.96%
Steam turbine	3/6/1	0.67%	0.70%
Slurry process	5/10/1	2.35%	2.52%

Table 4
Mean values of real data (target), absolute difference (between real and NN predicted data) and relative difference for a subset of testing samples.

	T_{Mixt_EngA}	F_{GasD}	P_{Cond}	F_{Steam}	POW_{ST}	F_{Ev}
Real (target)	0.36	0.48	0.725	0.75	0.74	0.49
Difference	0.0029	0.0047	0.0125	0.009	0.009	0.0036
Relative diff. %	0.8%	0.98%	1.7%	1.2%	1.2%	7.35%

therefore, that MLP is the most common used model in those above-cited works that accomplish the modeling of cogeneration processes by means of NNs. In addition, the MLP has not a very complex structure and hence, it can be used in a cooperative way with the evolutionary algorithm (EA) as proposed in this paper. As it will be explained in Section 4, the EA needs to evaluate the fitness function hundred of times by using the obtained NN models. An NN with a rather complex structure (e.g., recurrent NNs) would lead to a computation time too high for being used in the continuous on-line process of the plant.

In order to select the structure of the MLP model, some initial tests were made using different number of hidden layers. It was concluded that the accuracy did not show any significant improvement after increasing the number of hidden layers. Therefore, the simplest option was selected: only one hidden layer. Similarly, some initial tests were made with different number of hidden nodes. It was concluded that when the number of neurons increases beyond twice the number of inputs (i.e., a common practical rule), results barely improve. Therefore, the adopted criterion in all the models is, using twice as many hidden nodes as the number of inputs.

In our modeling, each NN computes a single output, which in some cases becomes the input for another model (see Table 2). These outputs are respectively: (a) the temperature of the water to cool each engine (T_{Mixt_EngA}, . . ., T_{Mixt_EngD}); (b) the flow of fuel (i.e., natural gas) required by each engine (F_{Gas_A}, . . ., F_{Gas_D}); (c) the pressure of the condenser (P_{CON}); (d) the steam flow in the boiler (F_{Steam}); (e) the electric power produced by the turbine (POW_{ST}) and (f) the flow of slurry processed by the evaporator (F_{EV}).

To train and test the NNs, a big and complete data set was collected trough a one-year observation process in the real plant. 213 parameters were identified as being potentially relevant for training and validating the NN models. Their values have been measured and retrieved with a resolution of one minute during the whole period of observation. Firstly, a careful analysis of the data was performed to choose the most relevant variables and also to filter outliers, missing data or un-informative variables. Next, based on previous knowledge of the system physics and also on a trial/error process, we determined the particular input variables involved in each NN. Table 2 shows which are the input/output variables for the different NNs. The variables written in bold refer to the decision variables that will be used later in the optimization process. To make the huge data set more tractable, we have selected 10-minute separated values. This action can be realized, because we have observed that the variables change very little in that period, due to the slow dynamics of the plant. We, thus, obtain a total of about 40,000 samples for each variable. Now, for making the NNs capable of modeling the different dynamics of the process throughout the whole year, we have made the training/testing partition in the following way: data from odd months (i.e., January, March, May, . . .) are used to train the models, and data from even months (i.e., February, April, June, . . .) are used to test the modeling performance of the trained NNs. The whole training/testing process has been carried out by using the Optibat trainer Tool [49]. In addition, all the variables have been normalized.

The results of the modeling are presented in Table 3, where the mean absolute error $MAE = (1/N)\sum_{i=1}^{N}|y_i - y_i'|$ between the desired (y_i) and actual (y_i') output, for both the training and the testing phase, is shown. We observe that, in all the cases, the obtained errors are very small. More precisely, most of the models provide a training error less than 1%. Only the slurry process exhibits a slightly higher error, which can be attributed to little information about its behavior. Even in the testing case, where the NNs deal with unseen data, the error is quite small. Note that the testing error represents the model's behavior better than the training error as it contains unseen data during the training of the models. Hence, the testing error is used to evaluate the model's predictive behavior. We can see that for all the models, the difference between the training error and the testing error was always less than 0.3%. This means that the models were capable of learning the dynamic of the systems and can make accurate predictions when dealing with unseen data. These results validate the modeling performance of the trained NNs.

For better appreciating the modeling ability of the NNs, we have plotted the absolute difference between the predicted data and the real data and also the real data (target) for a subset of the testing points (Figs. 2–7). Because the four cooling-circuit models and the four engine models have similar behavior, we have plotted only the output variables of six of the twelve models: T_{Mixt_EngA}, F_{GasD}, P_{Cond}, F_{Steam}, POW_{ST} and F_{Ev}. It can be observed in the figures that the differences are in general small, except in the slurry process, which is modeled slightly worse as we have explained above. In particular, if we compare the mean value of these differences with the mean value of the respective target values (see Table 4), we can see that, in all the cases, the difference is around 1%, except in the slurry process with a 7% error. Thus, it is confirmed that, in general, the NNs are able to learn rather well the dynamics of the plant.

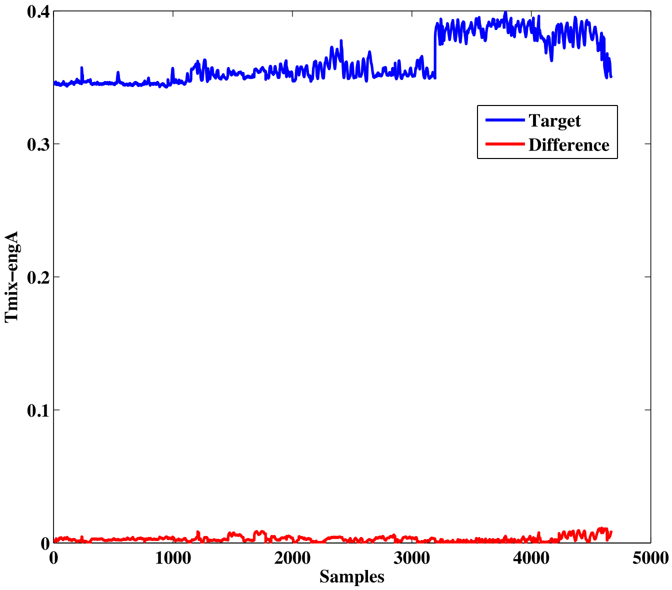

Fig. 2. Cooling circuit temperature of Engine A (T_{Mixt_EngA}): real data (target) and absolute difference between real data and NN predicted data for a subset of the testing set.

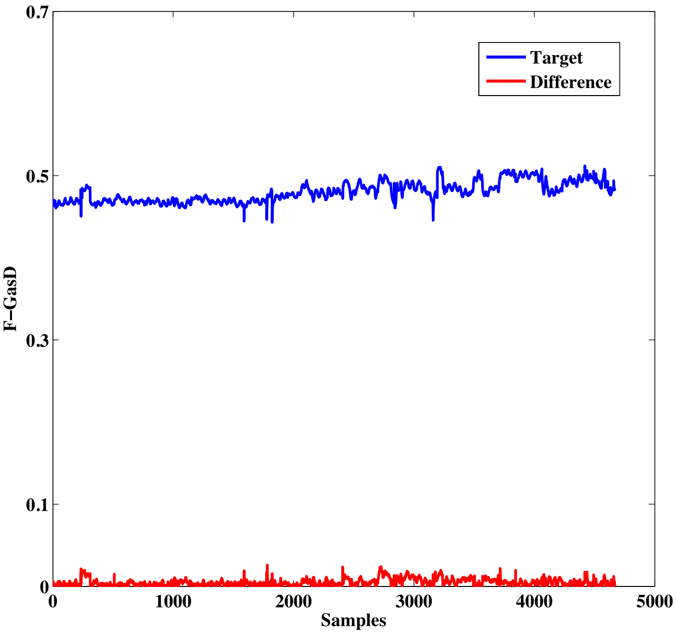

Fig. 3. Gas flow of Engine A (F_{GasA}): real data (target) and absolute difference between real data and NN predicted data for a subset of the testing set.

4. Comparative evolutionary optimization

Once the CHP plant has been modeled by means of the connected NNs, the next step is to carry out an optimization process to improve the efficiency of the whole process. In particular, we focus on three performance objectives: (1) minimizing the amount of used fuel Q_{fuel} (i.e., natural gas flow); (2) maximizing the useful thermal energy FE_v (i.e., flow of the fluent in the evaporator) and (3) maximizing the generated power *POW*. Therefore, we have actually a multi-objective optimization problem. To perform this process, a total of twelve decision variables are available in the plant; that is, a set of input variables whose values can be changed freely (within certain bounds) by the user. These twelve variables are those

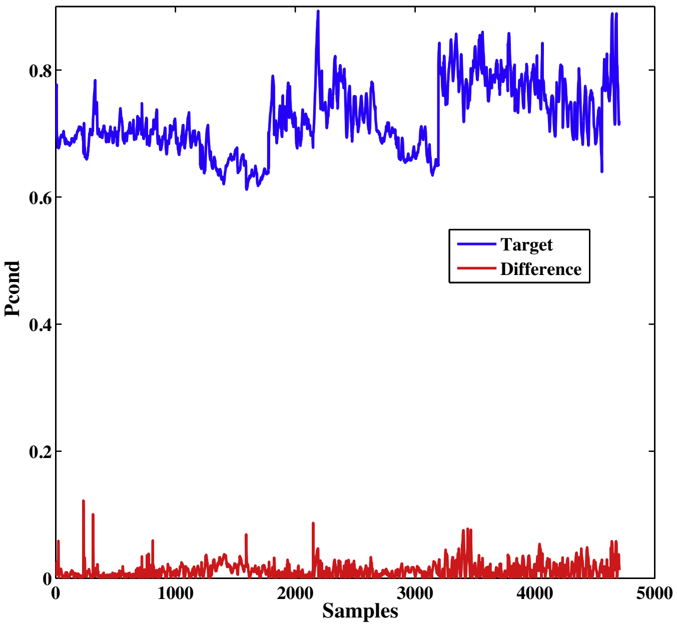

Fig. 4. Steam pressure of the condenser (P_{Cond}): real data (target) and absolute difference between real data and NN predicted data for a subset of the testing set.

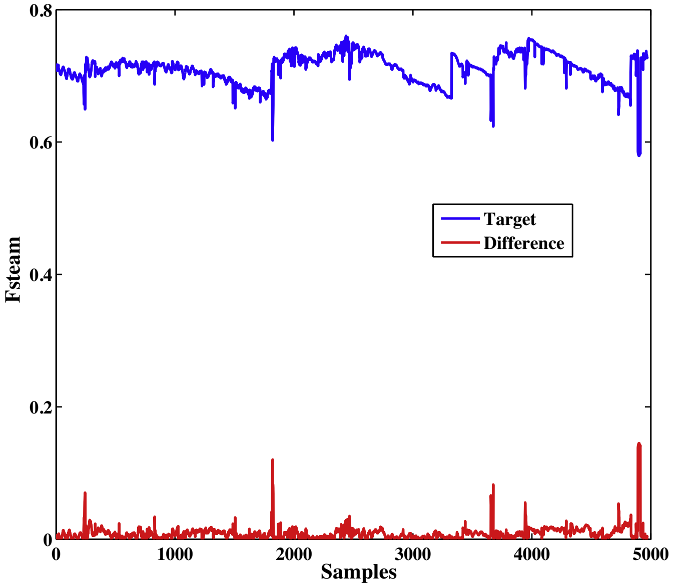

Fig. 5. Steam flow in the boiler (F_{Steam}): real data (target) and absolute difference between real data and NN predicted data for a subset of the testing set.

highlighted in bold in Table 2. The mathematical formulation of this multi-objective problem is as follows:

- Minimize used fuel: $Q_{fuel} = F_{Gas_A} + F_{Gas_B} + F_{Gas_C} + F_{Gas_D}$.
- Maximize drying process: F_{Ev}.
- Maximize power: $POW = POW_A + POW_B + POW_C + POW_D + POW_{ST}$.

The decision variables and their restrictions are listed below:

- T_{B1_A}, T_{B2_A}, T_{B1_B}, T_{B2_B}, T_{B1_C}, T_{B2_C}, T_{B1_D}, T_{B1_D} (i.e., two air intake temperatures for each engine): 30 °C $\leq T \leq$ 38 °C.
- $T_{H_2O\_Ex}$ (exchange water temperature): 61 °C $\leq T \leq$ 65 °C.
- P_{St_Gen} (pressure of the steam generator): 20 bar $\leq P \leq$ 22 bar.
- P_{Ev} (evaporator pressure): 0.13 bar $\leq P \leq$ 0.17 bar.

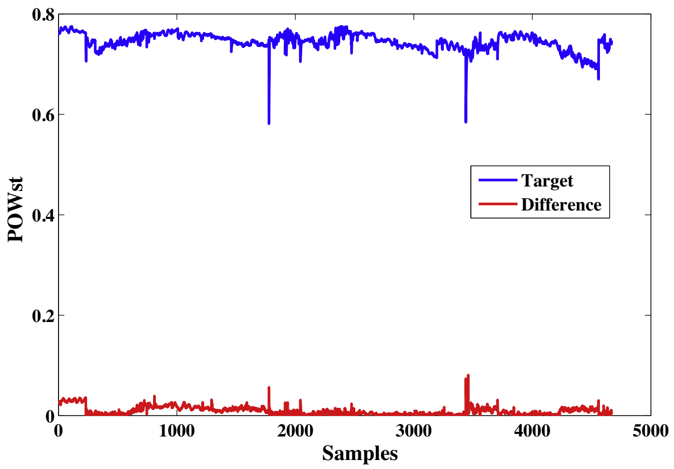

Fig. 6. Power generated in the Turbine (POW_{ST}): real data (target) and absolute difference between real data and NN predicted data for a subset of the testing set.

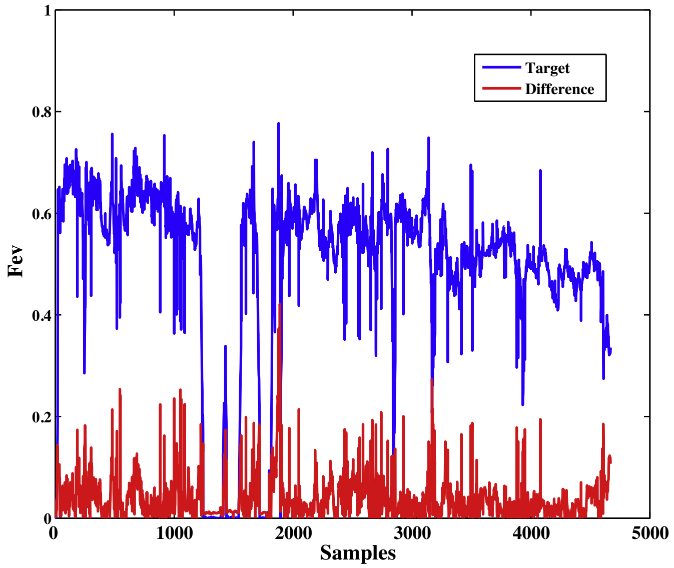

Fig. 7. Slurry drying process (F_{Ev}): real data (target) and absolute difference between real data and NN predicted data for a subset of the testing set.

Table 5
Mean and standard deviation (as subscript) of median IGD across all test problems.

AbYSS	ESPEA	IBEA	MOEAD
$3.41e-4_{2.13e-5}$	$1.69e-4_{1.92e-5}$	$1.53e-3_{1.88e-4}$	$1.51e-3_{1.72e-4}$
NSGA-II	NSGA-III	SMPSO	SMS-EMOA
$3.55e-4_{2.20e-5}$	$1.25e-3_{1.33e-4}$	$3.30e-4_{2.07e-5}$	$1.53e-3_{1.68e-4}$

- $T_{H_2O\_SH}$ (superheated water temperature): 110 °C ≤ T ≤ 125 °C.

Our combined approach of using NNs as black box functions may be applied in conjunction with any optimization algorithm that is able to handle real-valued decision variables. For this reason, several state-of-the-art, multiobjective evolutionary algorithms, which use different search strategies, are considered for solving the proposed optimization problem.

The difficulty in solving a multiobjective optimization problem is that there usually exists no single solution that optimizes all goals at the same time [50,51]. Instead, metaheuristics aim at finding a representative approximation to the so-called Pareto optimal front. The Pareto front comprises all solutions that can only be improved in one objective by impairing at least one other objective. The idea is that a decision maker chooses a solution to implement from this approximation. The approximation of a Pareto front is graded with respect to two criteria. The points found by an algorithm should be located as close as possible to the Pareto front. At the same time, the approximation should cover the Pareto front in its entirety so the decision maker has full knowledge about the available options. The former aspect is denoted by convergence and the latter by diversity.

AbYSS [52] uses a scatter search template as local search operator. ESPEA's [47] niching technique is inspired by the physical phenomenon of electrostatic potential energy. Indicator-based selection guides the search mechanism of IBEA [53]. MOEA/D [54] simultaneously solves multiple scalarized instances of the original problem. NSGA-II [55] uses a domination based topological sorting and the crowding distance metric as niching technique. It's successor, NSGA-III [56], applies a reference point based search method. SMS-EMOA [57] aims at finding a population that maximizes the so-called hypervolume measure. Finally, SMPSO [58] is a particle swarm optimization approach. All algorithms presented in this paragraph have been extensively studied on artificial and real-world optimization problems and shown strong results, making them ideal candidates for our study. MOEA/D and NSGA-III have also been specifically designed for many-objective problems, which deal with four or more objectives [59]. IBEA and SMS-EMOA appear to obtain good approximations of higher-dimensional Pareto fronts [60], while techniques that rely on crowding distance for niching, such as AbYSS, NSGA-II and SMPSO, generally struggle to approximate higher dimensional Pareto fronts [61]. ESPEA, on the other hand, has been successfully applied to four and five objective smart building optimization problems [62]. We have performed an extensive computational study making use of the jMetal framework version 4.5 [63], in order to assess the performance of these individual algorithms. Our code and resources are hosted online on Sourceforge and are publicly available.[2]

As stated before, the performance of the CHP plant is influenced by 213 different parameters of which 36 were found to have a significant impact as indicated in Table 2. While twelve of these parameters may be manipulated by the plant operator as decision variables, there still exist 24 parameters, whose different combinations of values potentially affect the optimization effort. We therefore randomly picked 39 parameter observations from our database, which come from a week of the month February and serve as representative sample. For each observation, we extracted the values of those 24 parameters that affect the plant operation and cannot be manipulated by the operator. These 24 parameters of each of the 39 observations serve as individual problem instances for our computational study. The objective of this study is identifying the algorithm that delivers the best performance by choosing optimal values for the twelve decision variables across all 39 problem instances. Each algorithm was run 100 times on every problem instance employing a population size of 100.[3] Algorithm configurations were taken from their original publications [52,47,53–56,58,57]. A detailed description of the algorithms' configurations can be found in Appendix A. 50,000 function evaluations were performed per run. Preliminary tests have revealed that the populations of the algorithms assessed in this study become evolutionary stable at 50,000 evaluations.

We chose the Inverted Generational Distance (IGD) as performance metric, since it captures both convergence and diversity [64]. The IGD metric computes the average of the minimum

[2] http://sourceforge.net/projects/jmetalbymarlonso/.

[3] An archive size of 100 and a population size of 20 was chosen for AbYSS as suggested in [52].

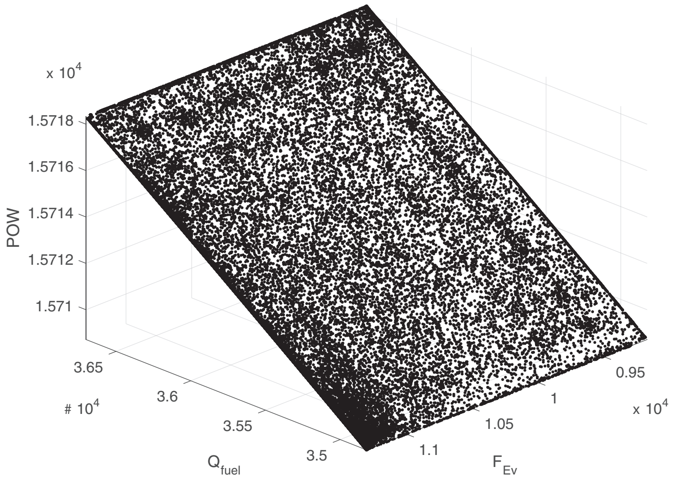

Fig. 8. Pareto front of problem instance 2 out of 39 of the cogeneration optimization problem. The front is a collection of all non-dominated solutions that were retrieved in final populations during the study. The Pareto front that these points describe can be approximated by a plane: $POW = 15530 + 5.13\text{e}-3F_{Ev} - 7.06\text{e}-7Q_{fuel}$.

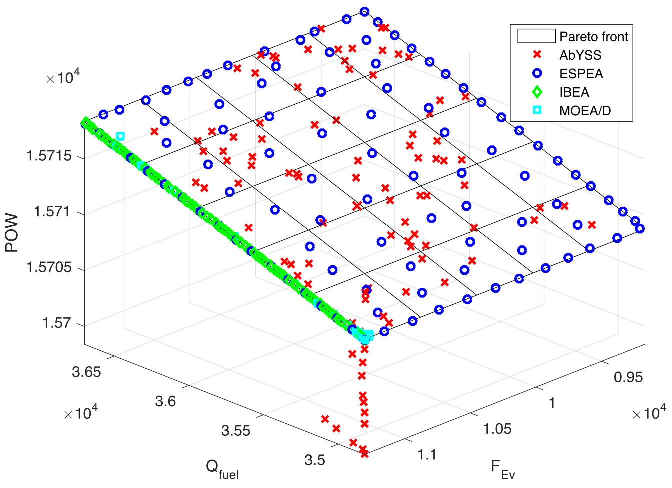

Fig. 9. Exemplary search results of the algorithms AbYSS, ESPEA, IBEA and MOEA/D on problem instance 2 for illustrating their performance.

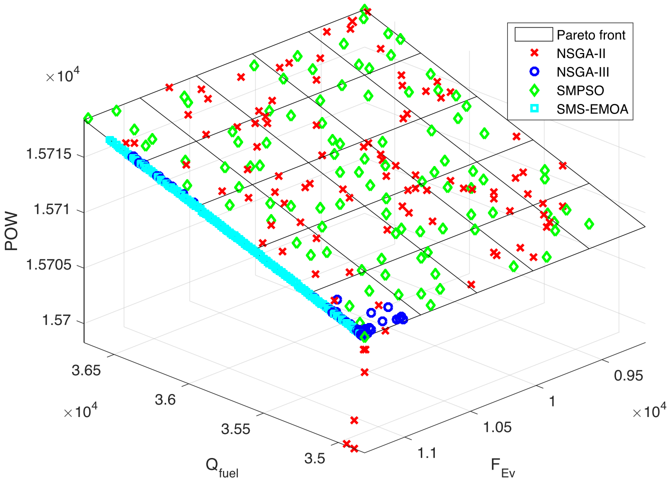

Fig. 10. Exemplary search results of the algorithms NSGA-II, NSGA-III, SMPSO and SMS-EMOA on problem instance 2 for illustrating their performance.

distances of every Pareto optimal point to a given Pareto front approximation. Since the Pareto fronts are not known in our case, we use all nondominated solutions obtained across all algorithm runs of a single problem instance as reference front (an example is given in Fig. 8). Objective values were normalized to mitigate the effect of different scales.

A preliminary analysis has revealed that the performance of an individual algorithm only differs marginally across the different problem instances. This observation indicates that our approach is very robust with respect to the parameters that cannot be influenced by the operator. For the sake of clarity, we therefore only provide a summary of the results in Table 5. Full results are provided in the Appendix in Table B.7.

The study results demonstrate that there exist clear performance differences between individual algorithms. Values of the IGD metric differ by a factor of ten from best to worst. This implies that the choice of algorithm greatly influences the optimization outcome. Best results are obtained using ESPEA, whereas AbYSS, NSGA-II and SMPSO show also good performances. IBEA, MOEA/D, NSGA-III and SMS-EMOA, on the other hand, trail behind. The smallest average IGD is achieved by ESPEA. A Kruskal–Wallis test [65] in conjunction with a post hoc analysis has revealed that the difference in IGD values between ESPEA and the other algorithms within all 39 problems is significant.[4] We also assessed, whether the performance differences across all problem instances observed in Table 5 are significant. A Friedman test [66] in conjunction with a post-hoc analysis has revealed that ESPEA's performance differs significantly with respect to the other algorithms with the exception of SMPSO. Detailed results are displayed in Table 6.

A closer analysis of the Pareto front reveals possible explanations for the performance differences observed. Fig. 8 shows the Pareto front of a cogeneration optimization problem instance. The rectangular shape of the front suggests that all Pareto optimal points lie on a plane. A regression analysis has indeed confirmed that, for 38 out of 39 problem instances, all points can be fitted in a plane with a coefficient of determination of one and a root mean square error of about 0.019.[5] Interestingly, the Pareto front is almost a linear function, although the problem itself is not. We believe that this observation can be explained by the activation function used by the neural network. We used the logistic function $1/(1 + \exp(-t))$, which can be approximated by a piecewise linear curve.

Figs. 9 and 10 offer an explanation to why algorithm performances can be divided into two tiers. AbYSS, ESPEA, NSGA-II and SMPSO capture the extent of the Pareto front in its entirety, whereas ESPEA achieves the most equidistant approximation. NSGA-II and AbYSS retrieve several dominated points as indicated in the plot. IBEA, MOEA/D, NSGA-III and SMS-EMOA focus mainly on a single edge of the front. We believe that using reference points from [67] as it is suggested in [54,56] is problematic given the presented front. We assume that the reference points do not cover the front equally, which leads to a strong focus on boundary solutions. A better choice of reference points might ameliorate this issue. Hypervolume-based methods, such as SMS-EMOA and the IBEA configuration used in this study, seem to struggle with the geometry of the front. We speculate that, despite objective normalization, boundary points yield the highest hypervolume contributions on plane-shaped fronts. Although the figures only depict single runs, these basic observations could be confirmed for other problem instances and different runs as well.

Robustness is another aspect of algorithm performance that is of interest in our setting. In practice, it is usually not feasible to

[4] *P*-values were equal to zero in our analysis with the exception of the pair (ESPEA/IBEA) on CG0 being 0.0097.

[5] The Pareto front of problem instance 1 out 39 (referenced as CG0 in Table B.7) consists of two different planes.

Table 6
P-values of a post hoc analysis resulting from a Friedman test for comparing median IGD across all problem instances.

	ESPEA	IBEA	MOEAD	NSGA–II	NSGA–III	SMPSO	SMS-EMOA
AbYSS	0.0045	6.01e−8	4.90e−6	0.6182	0.0145	0.5542	5.99e−8
ESPEA		5.99e−8	5.99e−8	8.59e−7	5.99e−8	0.5863	5.99e−8
IBEA			0.7396	1.30e−5	0.0145	5.99e−8	0.4587
MOEAD				0.0145	0.6182	5.99e−8	0.0063
NSGA–II					0.7396	0.0053	5.99e−8
NSGA–III						3.00e−6	1.41e−6
SMPSO							5.99e−8

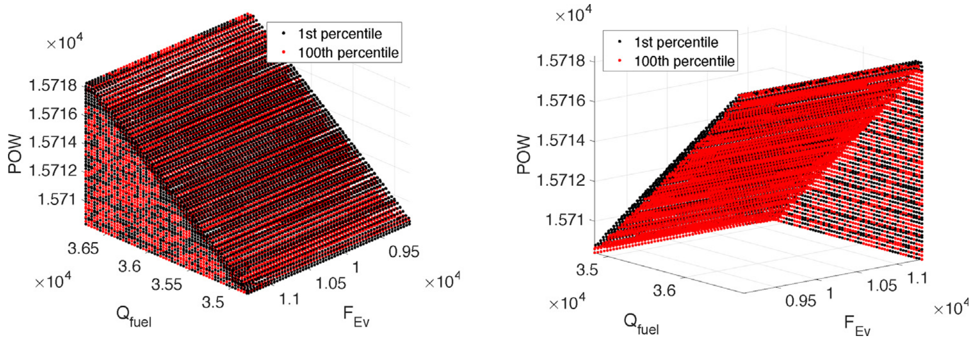

Fig. 11. Surface attainment plot of ESPEA for the first and 100th percentile of problem instance 2 from the front (left) and the back (right). The first percentile depicts the space that is dominated by all ESPEA runs combined, while the 100th percentile shows the space that is dominated in all of the 100 runs.

conduct 100 runs and choose the most preferable option from this pool of alternatives. If there is little variability between the results of individual runs, however, we can conclude that every single run yields a satisfying approximation of the Pareto front. One way of assessing robustness is considering the surface attainment [68] across multiple runs. Surface attainment measures the space that is dominated by a given approximation set. When measured across multiple runs, surface attainment yields the space that is dominated in a given percentage of runs. Fig. 11 provides an example for the visualization of the surface attainment of ESPEA of the first and 100th percentile. We can see that both surfaces nearly coincide, further indicating that our optimization approach is very robust across different runs. Similar observations were made for other problem instances.

The multiobjective approach generates a set of solutions among which a decision maker chooses an option that fits his preferences best. In the present context, there exists an efficiency measure that may be used to evaluate the quality of the cogeneration process. Efficiency can be defined as a quotient of the power generated by the unused energy contained in the fuel:

$$\varepsilon_{EE} = \frac{100 \cdot POW}{Q_{fuel} - F_{Ev}/0.9}. \tag{1}$$

A question that needs to be addressed in this context is, whether the multi-objective approach is suited to find a solution that maximizes the efficiency of the cogeneration process. Our computational study has revealed that all algorithms with the exception of NSGA-II and NSGA-III were able to find a solution possessing an efficiency of 70.6 on every problem instance in their median runs. The best solutions obtained by NSGA-II and NSGA-III across all problems exhibit an efficiency of 70.5. If a choice rule such as (1) is given, it makes sense to focus the search from the beginning on those regions of the Pareto front that yield the highest efficiency. In a multi-objective context, however, a decision maker is usually not only interest in obtaining a preconceived optimum, but also in comparing his choice to other options available [69,70].

ESPEA provides a mechanism that bridges the gap between those conflicting goals. The basic notion of this algorithm consists of interpreting the Pareto front as closed physical system in which charged particles, represented by solutions, may move freely. The aim of ESPEA is to obtain an approximation of a fixed, given size of the Pareto front that minimizes the energy of the system. For this purpose, the algorithm maintains an archive of nondominated solutions. The energy of the archive is computed as the sum of all pair-wise energies between individual solutions of the archive. In physics, the electrostatic energy that works between two particles is the product of their charges divided by their Euclidean distance. ESPEA uses the distance in objective space for computing energies. Charges are interpreted as utility values, where a lower utility corresponds to a higher desirability. In case all solutions are equally desirable, charges are set to one, as it was done for the computational study. In the second part of our analysis, we use (1) as charges for biasing the search towards the efficiency optimum. Efficiency values were raised by three to increase the bias of the search as hinted at in [47].

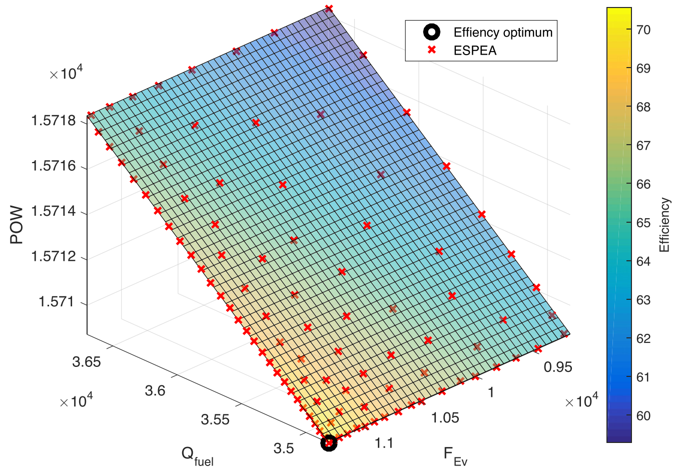

Fig. 12. A single run of ESPEA using efficiency values as charges on problem instance 31.

Fig. 12 illustrates the effect of using the charge mechanism in ESPEA. We observe that the density of solutions is higher in those regions that show high efficiencies. At the same time, an

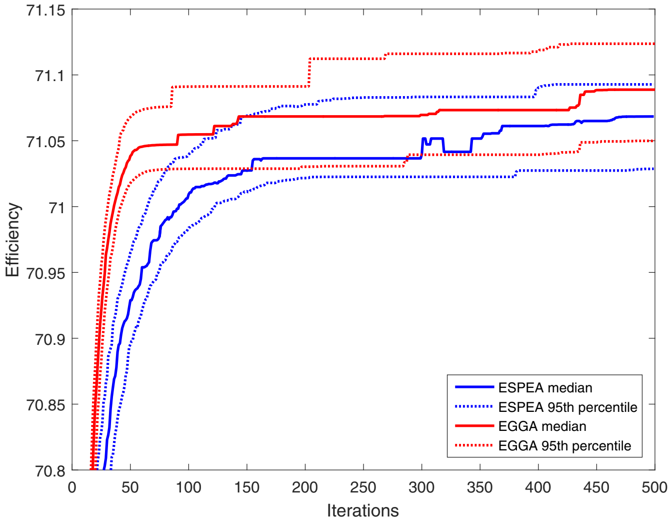

Fig. 13. Comparison of ESPEA using efficiency values as charges to an EGGA. We computed the median and the 95th percentile of the best efficiency observed in each iteration and for every problem across all 100 runs for each algorithm. The median of medians and the 95th percentile of 95th percentiles across all 39 problems are depicted in the figure.

approximation to the Pareto front in its entirety is retained enabling the decision maker to compare the highest efficiency option to other alternatives available.

Using ESPEA's charge mechanism focuses the search on generating more solutions that are close to the efficiency optimum. In this context, it appears reasonable to compare ESPEA to a single-objective optimization algorithm with respect to the efficiency optima that both algorithms obtain. We have chosen an elitist generational genetic algorithm (EGGA) for this task, which uses binary tournament selection [71], simulated binary crossover [72], polynomial mutation [73] and a population size of 100. The genetic operators were configured in the same way as employed by ESPEA. Fig. 13 depicts a comparison of both algorithms' performances across all test problems. Although the EGGA converges faster, especially within the first 150 iterations, the performances of both algorithms align the more function evaluations are performed. Considering absolute values, the performance differences are almost negligible.

Since the cogeneration optimization problem as presented in this work is embedded in a real-time application, algorithm run times are a critical issue in this context. As mentioned before, the thermodynamical processes inside the CHP change rather slowly. Therefore, the plant is expected to be reconfigured in regular intervals of 15 min. After 15 min have elapsed, optimization is performed using the values of the 24 significant parameters that are currently observed. The plant rescheduling is performed automatically by the algorithm choosing the solution of the final population possessing the highest efficiency. A plant operator may optionally review the algorithm's choice by analyzing the tradeoffs observed on the Pareto front. A single algorithm run should therefore take considerably less than 15 min on commercial off-the-shelf medium-class computer hardware. With the exception of ESPEA and SMS-EMOA, all algorithms finish 50,000 function evaluations within less than 10 s on an Intel Core i5-4300U processor running Windows 8.1. A single ESPEA run takes about half a minute, whereas an SMS-EMOA execution can take up to 10 min. Although run-times also highly depend on the implementation and the chosen programming language, the most important factor is the computational complexity of executing a single iteration of a given optimization algorithm. ESPEA and SMS-EMOA are steady-state algorithms implying that more costly operations are performed per function evaluation. SMS-EMOA, for example, computes hypervolume contributions for every member of the last non-dominated front in each iteration. Computing hypervolume contributions is an NP-hard problem [74], which consequently results in a very high run-time. Considering all this, we may draw the conclusion that every algorithm assessed in this study is suitable for an on-line implementation with the exception of SMS-EMOA.

5. Conclusion and future work

We have presented a neuro-genetic approach for both modeling and optimizing a complete real CHP plant. The modeling is accomplished by means of interconnected neural networks, where each network deals with one sub-system of the plant. Obtained error rates in the experiments show a satisfactory modeling for both training and test data. Next, multiobjective evolutionary algorithms use the NNs as black box functions for finding optimal operation schedules. Different evolutionary algorithms have been analyzed and obtained results have revealed that ESPEA is the most suitable algorithm to solve the problem. Unlike other similar approaches proposed in the literature, our work addresses the CHP plant problem in a wider and more general way: (1) all the sub-systems of the CHP process are considered; (2) both the modeling and the optimization are carried out using soft computing approaches; (3) only real values from the physical plant are used in the work; (4) the optimization has been carried out as a multiobjective problem; and (5) the optimization algorithm is fast enough to allow an on-line optimization of the plant. Our findings can spur further research into robust optimization formulations across uncertainty modeling in parameter configurations for cogeneration plants.

Our future research focuses on combining different preferences function with our approach, such as tradeoff based techniques [75–78]. A special emphasis is on multiobjective algorithms that employ a cone-based domination technique [79,80]. Cone-based domination have shown to be a powerful tool in preference modeling [81–83]. Further soft computing techniques besides evolutionary algorithms, such as artificial immune systems [84] or infeasibility driven optimization [85] or techniques from [86,87] may also be applied to solve the optimization problem proposed in this paper. Another interesting approach would be to approximate the activation function used in our study, and to have a mathematical formulation of our optimization problem. Fast exact algorithms, like Newton-based techniques [88], could then be applied to solve this optimization problem.

Acknowledgements

This work has been partially funded by the Basque Government under Grant IT733-13, and the Spanish Ministry of Science and Innovation under Grant TEC2013-42286-R.

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at http://dx.doi.org/10.1016/j.asoc.2016.07.026.

References

[1] I. Noshadi, A. Salahi, M. Hemmati, F. Rekabdar, T. Mohammadi, Experimental and ANFIS modeling for fouling analysis of oily wastewater treatment using ultrafiltration, Asia-Pac. J. Chem. Eng. 8 (4) (2013) 527–538.

[2] G. Isazadeh, R. Hooshmand, A. Khodabakhshian, Design of an adaptive dynamic load shedding algorithm using neural network in the steelmaking cogeneration facility, Iran. J. Sci. Technol. Trans. Electr. Eng. 36 (1) (2012) 67–82.

[3] Y. Zhang, J. Ai, Zeta potential modeling of papermaking wastewater on neural network, in: 2012 Second International Conference on Instrumentation,

Measurement, Computer, Communication and Control (IMCCC), 2012, pp. 63–66.

[4] M. Budnik, W. Stanek, H. Rusinowski, Application of neural modelling in hybrid control model of fluidized bed boiler fired with coal and biomass, in: 2012 13th International Carpathian Control Conference (ICCC), IEEE, 2012, pp. 69–74.

[5] H. Xian, W. Hui, Modelling of combustion process characteristics of a 600 MW boiler by T-S fuzzy neural networks, in: International Conference on Energy and Environment Technology, 2009, ICEET'09, vol. 1, 2009, pp. 737–740.

[6] E. Rakhshani, I. Sariri, K. Rouzbehi, Application of data mining on fault detection and prediction in boiler of power plant using artificial neural network, in: International Conference on Power Engineering, Energy and Electrical Drives, 2009, POWERENG'09, 2009, pp. 473–478.

[7] T.A. Lemma, F.M. Hashim, IFDD: Intelligent fault detection and diagnosis-application to a cogeneration and cooling plant, Asian J. Sci. Res. 6 (3) (2013) 478.

[8] Y. Shatnawi, M. Al-khassaweneh, Fault diagnosis in internal combustion engines using extension neural network, IEEE Trans. Ind. Electron. 61 (3) (2014) 1434–1443.

[9] V. Ghate, S. Dudul, Cascade neural-network-based fault classifier for three-phase induction motor, IEEE Trans. Ind. Electron. 58 (5) (2011) 1555–1563.

[10] S. Refaat, H. Abu-Rub, M. Saad, E. Aboul-Zahab, A. Iqbal, ANN-based for detection, diagnosis the bearing fault for three phase induction motors using current signal, in: 2013 IEEE International Conference on Industrial Technology (ICIT), 2013, pp. 253–258.

[11] F. Rossi, D. Velázquez, I. Monedero, F. Biscarri, Artificial neural networks and physical modeling for determination of baseline consumption of CHP plants, Expert Syst. Appl. 41 (10) (2014) 4658–4669.

[12] S. De, M. Kaiadi, M. Fast, M. Assadi, Development of an artificial neural network model for the steam process of a coal biomass cofired combined heat and power (CHP) plant in Sweden, Energy 32 (11) (2007) 2099–2109.

[13] J. Smrekar, D. Pandit, M. Fast, M. Assadi, S. De, Prediction of power output of a coal-fired power plant by artificial neural network, Neural Comput. Appl. 19 (5) (2010) 725–740.

[14] H. Nikpey, M. Assadi, P. Breuhaus, Development of an optimized artificial neural network model for combined heat and power micro gas turbines, Appl. Energy 108 (2013) 137–148.

[15] N. Sisworahardjo, M. El-Sharkh, Validation of artificial neural network based model of microturbine power plant, in: 2013 IEEE Industry Applications Society Annual Meeting, 2013, pp. 1–5.

[16] D. Flynn, J. Ritchie, M. Cregan, Data mining techniques applied to power plant performance monitoring, in: World Congress, vol. 16, 2005, p. 1636.

[17] L. Pan, D. Flynn, M. Cregan, Statistical model for power plant performance monitoring and analysis, in: 42nd International Universities Power Engineering Conference, 2007, UPEC 2007, 2007, pp. 121–126.

[18] T. Bekat, M. Erdogan, F. Inal, A. Genc, Prediction of the bottom ash formed in a coal-fired power plant using artificial neural networks, Energy 45 (1) (2012) 882–887, the 24th International Conference on Efficiency, Cost, Optimization, Simulation and Environmental Impact of Energy, ECOS 2011.

[19] M.K. Kirar, G. Agnihotri, Design of high speed adaptive load shedding for industrial cogeneration system, Int. Rev. Electr. Eng. 8 (2) (2013).

[20] R. Zomorodian, M. Rezasoltani, M.B. Ghofrani, Static and dynamic neural networks for simulation and optimization of cogeneration systems, Int. J. Energy Environ. Eng. (2011).

[21] O. Arslan, Power generation from medium temperature geothermal resources: ANN-based optimization of kalina cycle system-34, Energy 36 (5) (2011) 2528–2534.

[22] J.-J. Wang, C.-F. Zhang, Y.-Y. Jing, Self-adaptive RBF neural network PID control in exhaust temperature of micro gas turbine, in: 2008 International Conference on Machine Learning and Cybernetics, vol. 4, 2008, pp. 2131–2136.

[23] K. Lee, J. Van Sickel, J. Hoffman, W.-H. Jung, S.-H. Kim, Controller design for a large-scale ultrasupercritical once-through boiler power plant, IEEE Trans. Energy Convers. 25 (4) (2010) 1063–1070.

[24] I. Bertini, M.D. Felice, A. Pannicelli, S. Pizzuti, Soft computing based optimization of combined cycled power plant start-up operation with fitness approximation methods, Appl. Soft Comput. 11 (6) (2011) 4110–4116.

[25] M. Ameri, P. Ahmadi, A. Hamidi, Energy, exergy and exergoeconomic analysis of a steam power plant: a case study, Int. J. Energy Res. 33 (5) (2009) 499–512.

[26] C. Koch, F. Cziesla, G. Tsatsaronis, Optimization of combined cycle power plants using evolutionary algorithms, Chem. Eng. Process.: Process Intensif. 46 (11) (2007) 1151–1159, Special Issue on Process Optimization and Control in Chemical Engineering and Processing.

[27] F. Hajabdollahi, Z. Hajabdollahi, H. Hajabdollahi, Soft computing based multi-objective optimization of steam cycle power plant using NSGA-II and {ANN}, Appl. Soft Comput. 12 (11) (2012) 3648–3655.

[28] P. Ahmadi, I. Dincer, Thermodynamic and exergoenvironmental analyses, and multi-objective optimization of a gas turbine power plant, Appl. Thermal Eng. 31 (14–15) (2011) 2529–2540.

[29] K. Deb, F. Ruiz, M. Luque, R. Tewari, J.M. Cabello, J.M. Cejudo, On the sizing of a solar thermal electricity plant for multiple objectives using evolutionary optimization, Appl. Soft Comput. 12 (10) (2012) 3300–3311.

[30] M. Basu, Economic environmental dispatch of fixed head hydrothermal power systems using nondominated sorting genetic algorithm-II, Appl. Soft Comput. 11 (3) (2011) 3046–3055.

[31] R. Berrios, F. Nú nez, A. Cipriano, Fault tolerant measurement system based on Takagi–Sugeno fuzzy models for a gas turbine in a combined cycle power plant, Fuzzy Sets Syst. 174 (1) (2011) 114–130, Theme: Systems Engineering.

[32] H. Habbi, M. Kidouche, M. Zelmat, Data-driven fuzzy models for nonlinear identification of a complex heat exchanger, Appl. Math. Modell. 35 (3) (2011) 1470–1482.

[33] V. Mazur, Fuzzy thermoeconomic optimization of energy-transforming systems, Appl. Energy 84 (7–8) (2007) 749–762, Industrial Energy Analysis and Management: A European Perspective.

[34] A. Rodriguez-Martinez, R. Garduno-Ramirez, L. Vela-Valdes, PI fuzzy gain-scheduling speed control at startup of a gas-turbine power plant, IEEE Trans. Energy Convers. 26 (1) (2011) 310–317.

[35] U.-C. Moon, K. Lee, An adaptive dynamic matrix control with fuzzy-interpolated step-response model for a drum-type boiler-turbine system, IEEE Trans. Energy Convers. 26 (2) (2011) 393–401.

[36] M. Gunes, N. Dogru, Fuzzy control of brushless excitation system for steam turbogenerators, IEEE Trans. Energy Convers. 25 (3) (2010) 844–852.

[37] X. Liu, X. Kong, G. Hou, J. Wang, Modeling of a 1000 MW power plant ultra super-critical boiler system using fuzzy-neural network methods, Energy Convers. Manage. 65 (2013) 518–527, Global Conference on Renewable Energy and Energy Efficiency for Desert Regions 2011.

[38] L. Barelli, G. Bidini, Design of the measurements validation procedure and the expert system architecture for a cogeneration internal combustion engine, Appl. Thermal Eng. 25 (17-18) (2005) 2698–2714.

[39] L. Mastacan, I. Olah, C. Dosoftei, D. Ivana, Neuro-fuzzy models of thermoelectric power station installations, in: International Conference on Computational Intelligence for Modelling, Control and Automation, 2005 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, vol. 1, 2005, pp. 899–904.

[40] M. Rashidi, N. Galanis, F. Nazari, A.B. Parsa, L. Shamekhi, Parametric analysis and optimization of regenerative Clausius and organic Rankine cycles with two feedwater heaters using artificial bees colony and artificial neural network, Energy 36 (9) (2011) 5728–5740.

[41] M. Suresh, K. Reddy, A.K. Kolar, ANN-GA based optimization of a high ash coal-fired supercritical power plant, Appl. Energy 88 (12) (2011) 4867–4873.

[42] H. Sayyaadi, M. Babaie, M.R. Farmani, Implementing of the multi-objective particle swarm optimizer and fuzzy decision-maker in exergetic, exergoeconomic and environmental optimization of a benchmark cogeneration system, Energy 36 (8) (2011) 4777–4789, {PRES} 2010.

[43] Z. Su, P. Wang, J. Shen, Y. Zhang, L. Chen, Convenient T-S fuzzy model with enhanced performance using a novel swarm intelligent fuzzy clustering technique, J. Process Control 22 (1) (2012) 108–124.

[44] D. Saez, F. Milla, L. Vargas, Fuzzy predictive supervisory control based on genetic algorithms for gas turbines of combined cycle power plants, IEEE Trans. Energy Convers. 22 (3) (2007) 689–696.

[45] A. Tamiru, C. Rangkuti, F. Hashim, Neuro-fuzzy and PSO based model for the steam and cooling sections of a cogeneration and cooling plant (CCP), in: 3rd International Conference on Energy and Environment, 2009, ICEE 2009, 2009, pp. 27–33.

[46] S.-K. Oh, W. Pedrycz, H.-S. Park, Fuzzy relation-based neural networks and their hybrid identification, IEEE Trans. Instrum. Meas. 56 (6) (2007) 2522–2537.

[47] M.A. Braun, P.K. Shukla, H. Schmeck, Obtaining optimal Pareto front approximations using scalarized preference information, in: Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation, GECCO'15, ACM, New York, NY, USA, 2015, pp. 631–638.

[48] W.H. Hornik, K.M. Stinchcombe, Multilayer feedforward networks are universal approximators, Neural Netw. 5 (2) (1989) 359–366.

[49] Optimitive, Optibat, 2014 http://www.optimitive.com/ (accessed 01.01.16).

[50] K. Deb, Multi-Objective Optimization using Evolutionary Algorithms, vol. 16, John Wiley & Sons, 2001.

[51] C. Coello Coello, G. Lamont, D. Van Veldhuizen, Evolutionary Algorithms for Solving Multi-Objective Problems, Springer, 2007.

[52] A. Nebro, F. Luna, E. Alba, B. Dorronsoro, J. Durillo, A. Beham, ABYSS: adapting scatter search to multiobjective optimization, IEEE Trans. Evol. Comput. 12 (4) (2008) 439–457.

[53] E. Zitzler, S. Künzli, Indicator-based selection in multiobjective search, in: Parallel Problem Solving from Nature-PPSN VIII, Springer, 2004, pp. 832–842.

[54] H. Li, Q. Zhang, Multiobjective optimization problems with complicated pareto sets, MOEA/D and NSGA-II, IEEE Trans. Evol. Comput. 13 (2) (2009) 284–302.

[55] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2) (2002) 182–197.

[56] K. Deb, H. Jain, An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: Solving problems with box constraints, IEEE Trans. Evol. Comput. 18 (4) (2014) 577–601.

[57] N. Beume, B. Naujoks, M. Emmerich, SMS-EMOA: multiobjective selection based on dominated hypervolume, Eur. J. Oper. Res. 181 (3) (2007) 1653–1669.

[58] A. Nebro, J. Durillo, J. Garcia-Nieto, C. Coello Coello, F. Luna, E. Alba, SMPSO: a new PSO-based metaheuristic for multi-objective optimization, in: IEEE Symposium on Computational Intelligence in Multi-Criteria Decision-Making, 2009, MCDM'09, 2009, pp. 66–73.

[59] H. Ishibuchi, N. Tsukamoto, Y. Nojima, Evolutionary many-objective optimization: a short review, in: IEEE Congress on Evolutionary Computation, Citeseer, 2008, pp. 2419–2426.
[60] T. Wagner, N. Beume, B. Naujoks, Pareto-, aggregation-, and indicator-based methods in many-objective optimization, in: Evolutionary Multi-Criterion Optimization, Springer, 2007, pp. 742–756.
[61] S. Kukkonen, K. Deb, Improved pruning of non-dominated solutions based on crowding distance for bi-objective optimization problems, in: IEEE Congress on Evolutionary Computation, 2006, CEC 2006, IEEE, 2006, pp. 1179–1186.
[62] M. Braun, T. Dengiz, I. Mauser, H. Schmeck, Comparison of multi-objective evolutionary optimization in smart building scenarios, in: Applications of Evolutionary Computation, Springer, 2016, pp. 443–458.
[63] J.J. Durillo, A.J. Nebro, jMetal: a java framework for multi-objective optimization, Adv. Eng. Softw. 42 (2011) 760–771.
[64] D.A. Van Veldhuizen, G.B. Lamont, Evolutionary computation and convergence to a Pareto front, in: Late Breaking Papers at the Genetic Programming 1998 Conference, Citeseer, 1998, pp. 221–228.
[65] W.H. Kruskal, W.A. Wallis, Use of ranks in one-criterion variance analysis, J. Am. Stat. Assoc. 47 (260) (1952) 583–621.
[66] M. Friedman, The use of ranks to avoid the assumption of normality implicit in the analysis of variance, J. Am. Stat. Assoc. 32 (200) (1937) 675–701.
[67] I. Das, J.E. Dennis, Normal-boundary intersection: a new method for generating the pareto surface in nonlinear multicriteria optimization problems, SIAM J. Optim. 8 (3) (1998) 631–657.
[68] C.M. Fonseca, P.J. Fleming, On the performance assessment and comparison of stochastic multiobjective optimizers, in: Parallel Problem Solving from Nature-PPSN IV, Springer, 1996, pp. 584–593.
[69] B. Roy, Multicriteria Methodology for Decision Aiding, Springer Science & Business Media, 1996.
[70] D. Kahneman, A. Tversky, Prospect theory: an analysis of decision under risk, Econometrica (1979) 263–291.
[71] D.E. Goldberg, K. Deb, A comparative analysis of selection schemes used in genetic algorithms, Found. Genet. Alg. 1 (1991) 69–93.
[72] R.B. Agrawal, K. Deb, Simulated binary crossover for continuous search space, Complex Syst. 9 (3) (1994) 1–15.
[73] K. Deb, M. Goyal, A combined genetic adaptive search (GeneAS) for engineering design, Comput. Sci. Inform. 26 (1996) 30–45.
[74] K. Bringmann, T. Friedrich, Approximating the least hypervolume contributor: NP-hard in general but fast in practice, in: Evolutionary Multi-Criterion Optimization, Springer, 2009, pp. 6–20.
[75] M.A. Braun, P.K. Shukla, H. Schmeck, Preference ranking schemes in multi-objective evolutionary algorithms, in: R.H. Takahashi, K. Deb, E.F. Wanner, S. Greco (Eds.), Evolutionary Multi-Criterion Optimization, vol. 6576 of Lecture Notes in Computer Science, Springer, 2011, pp. 226–240.
[76] P.K. Shukla, M.A. Braun, H. Schmeck, Theory and algorithms for finding knees, in: R.C. Purshouse, P.J. Fleming, C.M. Fonseca, S. Greco, J. Shaw (Eds.), Evolutionary Multi-Criterion Optimization, vol. 7811 of Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2013, pp. 156–170.
[77] P.K. Shukla, C. Hirsch, H. Schmeck, A framework for incorporating trade-off information using multi-objective evolutionary algorithms, in: Parallel Problem Solving from Nature, PPSN XI, Springer, Berlin, Heidelberg, 2010, pp. 131–140.
[78] P.K. Shukla, C. Hirsch, H. Schmeck, Towards a deeper understanding of trade-offs using multi-objective evolutionary algorithms, in: Applications of Evolutionary Computation, Springer, Berlin Heidelberg, 2012, pp. 396–405.
[79] C. Hirsch, P.K. Shukla, H. Schmeck, Variable preference modeling using multi-objective evolutionary algorithms, in: Evolutionary Multi-Criterion Optimization, Springer, 2011, pp. 91–105.
[80] P.K. Shukla, M.A. Braun, Indicator based search in variable orderings: theory and algorithms., in: EMO, 2013, pp. 66–80.
[81] M.M. Wiecek, Advances in cone-based preference modeling for decision making with multiple criteria, Decis. Making Manuf. Serv. 1 (2007) 153–173.
[82] M. Emmerich, A. Deutz, J. Kruisselbrink, P.K. Shukla, Cone-based hypervolume indicators: construction, properties, and efficient computation, in: Evolutionary Multi-Criterion Optimization, Springer, Berlin, Heidelberg, 2013, pp. 111–127.
[83] P.K. Shukla, M. Emmerich, A. Deutz, A theoretical analysis of curvature based preference models, in: Evolutionary Multi-Criterion Optimization, Springer, Berlin, Heidelberg, 2013, pp. 367–382.
[84] R. Shang, L. Jiao, F. Liu, W. Ma, A novel immune clonal algorithm for mo problems, IEEE Trans. Evol. Comput. 16 (1) (2012).
[85] L. Jiao, J. Luo, R. Shang, F. Liu, A modified objective function method with feasible-guiding strategy to solve constrained multi-objective optimization problems, Appl. Soft Comput. 14 (Part C) (2014) 363–380.
[86] R. Shang, Y. Wang, J. Wang, L. Jiao, S. Wang, L. Qi, A multi-population cooperative coevolutionary algorithm for multi-objective capacitated arc routing problem, Inform. Sci. 277 (2014) 609–642.
[87] R. Shang, J. Wang, L. Jiao, Y. Wang, An improved decomposition-based memetic algorithm for multi-objective capacitated arc routing problem, Appl. Soft Comput. 19 (2014) 343–361.
[88] A. Fischer, P.K. Shukla, A Levenberg–Marquardt algorithm for unconstrained multicriteria optimization, Oper. Res. Lett. 36 (5) (2008) 643–646.