

# **Guidelines for WCCI(CEC)/GECCO 2023 Competition Evolutionary Computation in the Energy Domain: Operation and Planning Applications**

**Fernando Lezama, João Soares, José Almeida, Bruno Canizes, Zita Vale**

School of engineering (ISEP), Polytechnic of Porto, Porto, Portugal

[flz@isep.ipp.pt](mailto:flz@isep.ipp.pt), [jan@isep.ipp.pt](mailto:jan@isep.ipp.pt), [jorga@isep.ipp.pt](mailto:jorga@isep.ipp.pt), [bmc@isep.ipp.pt](mailto:bmc@isep.ipp.pt), [zav@isep.ipp.pt](mailto:zav@isep.ipp.pt)

**Leonardo H. Macedo, Gabriel F. Puerta, Ruben Romero**

Department of Electrical Engineering, University of São Paulo (UNESP), São Paulo, Brazil

[lhf.possagnolo@unesp.br](mailto:lhf.possagnolo@unesp.br), [gfpuerta@gmail.com](mailto:gfpuerta@gmail.com), [ruben.romero@unesp.br](mailto:ruben.romero@unesp.br)

IEEE CIS Task Force on 'Computational Intelligence in the Energy Domain (ci4energy)', part of IEEE CIS Intelligent Systems Applications TC (<http://ci4energy.uni-paderborn.de/committee/>)

IEEE PES Intelligent Systems Subcommittee (ISS), part of IEEE PES Analytic Methods for Power Systems TC (<http://sites.ieee.org/pes-iss/>)

IEEE PES Working Group on Modern Heuristic Optimization (<https://site.ieee.org/psace-mho/>)

January 2023

## Table of contents

1. Introduction.....	3
2. General description of operation and planning applications.....	4
2.1 Testbed 1.....	4
2.2 Testbed 2.....	5
3. Metaheuristic simulator framework.....	6
3.A) Encoding of the individual.....	7
Testbed 1.....	7
Testbed 2.....	7
3.B) Fitness function.....	8
Testbed 1.....	8
Testbed 2.....	10
3.C) Scenario overview.....	11
Testbed 1.....	11
Testbed 2.....	13
4. Guidelines for participants.....	14
4.A) <i>main.m</i> - Master function/script.....	14
<i>A.0 and A.1 - #main.m - Loading the testbed and case study</i> .....	15
<i>A.2 - #DEparameters.m - Set parameters of the metaheuristic</i> .....	15
<i>A.3 - #setOtherParameters.m - Set other necessary parameters and struct</i> .....	16
<i>A.4 - #setVariablesBounds.m - Set bounds of variables</i> .....	16
<i>A.5 - #HyDE.m - Algorithm proposed by the competitor</i> .....	16
<i>A.6 - #Save_results.m - Benchmark results (text-files)</i> .....	17
4.B) Fitness function evaluation.....	18
5. Evaluation guidelines.....	19
6. Material to be submitted to the organizers.....	19
Appendix: Mathematical formulation.....	20
Testbed 1.....	20
Testbed 2.....	20
<b>Bibliography</b> .....	<b>22</b>

## 1. Introduction

Following the success of the previous editions at PES GM (2017,2021), GECCO (since 2018 to 2022), WCCI and CEC (since 2017 to 2022)<sup>1</sup>, we are launching a new edition of our algorithm competition at major conferences in the field of computational intelligence. This CEC/GECCO 2023 competition proposes one track in the energy domain:

Track 1) Risk-based optimization of aggregators' day-ahead energy resource management (ERM) considering uncertainty associated with the high penetration of distributed energy resources (DER). This test bed is constructed under the same framework of past competitions (therefore, former competitors can adapt their algorithms to this new track), representing a centralized day-ahead ERM in a smart grid located in a 13-bus distribution network using a 15-scenario case study with 3 of these scenarios considered as extreme events (high impact, and low probability). A conditional value-at-risk (CVaR) mechanism is used to measure the risk associated with the extreme events for a confidence level ( $\alpha$ ) of 95%. We also add some restrictions to the initialization of the initial solution and the allowed repairs and tweak-heuristics.

Track 2) Transmission Network Expansion Planning. Long-term transmission network expansion planning (TNEP) is a classic problem of power systems. The objective is to find the optimal expansion plan that identifies the transmission lines that must be installed in the system to allow a proper operation within a predefined planning horizon with the lowest investment cost. The optimal expansion plan should define where and how many lines should be installed. A nonconvex mixed-integer nonlinear programming formulation is used to model the problem. The Northeast Brazilian transmission system is considered as case study.

The proposed track (testbed) is developed under the same framework of the past competitions with few adjustments.

**Tip: The most important part for a quick participation in this competition is section 4 of this guidelines, i.e., if you want to just implement your heuristic and treat the problem as pure black box item. Former sections (2-3) are introduction and explanation of the problem being optimized.**

---

<sup>1</sup> Check former competitions in <http://www.gecad.isep.ipp.pt/ERM-Competitions>

## 2. General description of operation and planning applications

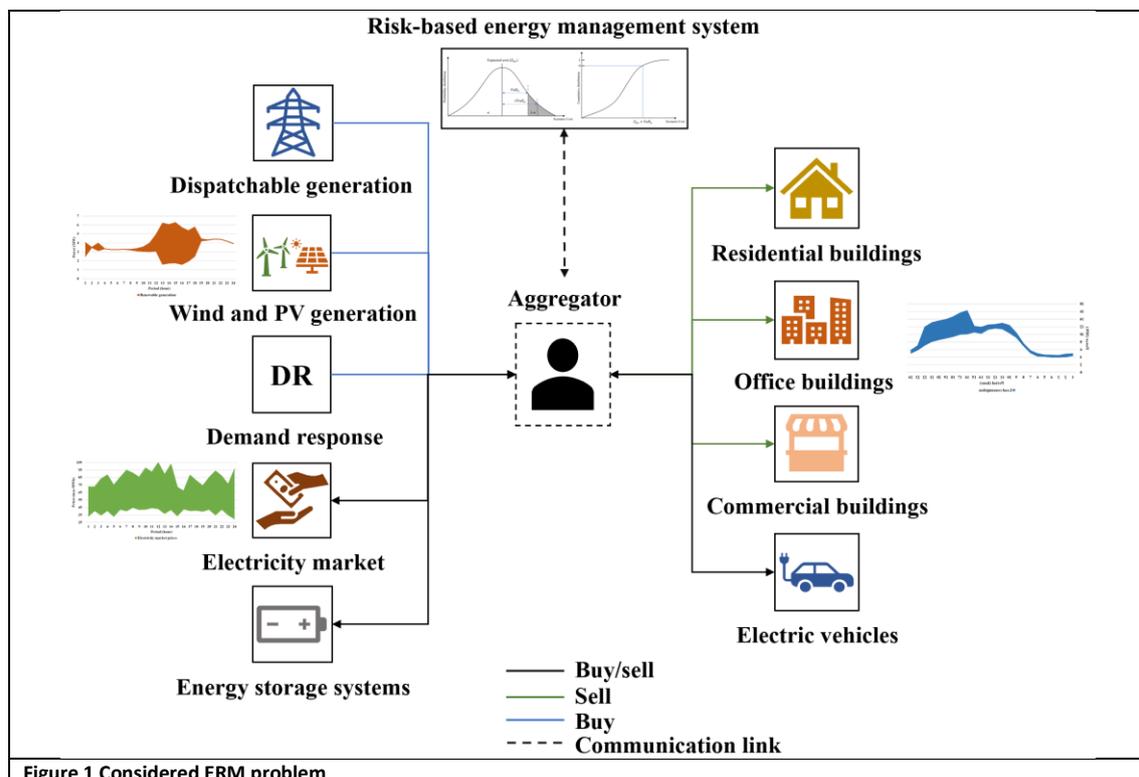
### 2.1 Testbed 1

Renewable generation, either solar or wind types, is usually weather-dependent. Therefore, their management has always associated an uncertain variability that jeopardizes the operation of the entire energy chain [1]. In this context, an energy management system should be both reliable and resilient, operating as expected in the case of events with a high probability of occurrence but a reasonably small effect (e.g., reliable under different faults in the power system), and prepared to deal with events that despite a low probability, have a considerable influence in the outcome of the operation (e.g., resilient to extreme events such as hurricanes, thunderstorm, etc.) [2]. Recently, energy management models are incorporating risk-based methods to handle the uncertainty associated with variable parameters [3]. In this track, we hypothesize that the incorporation of risk parameters into the formulation results in solutions that, despite their higher cost, protect the aggregator (and, in consequence, the end-user) against possible scenarios that might endanger the entire system.

In this testbed, we model the energy resource management (ERM) problem under uncertainty of renewable generation, load consumption, market prices, and electric vehicles trips, similar to the works presented in [4], [5]. The stochastic behavior of these parameters is considered through various scenarios that have associated a probability of occurrence. The novelty of this new testbed lies in the incorporation of risk strategies into the formulation, allowing the aggregator to plan its operation considering different degrees of risk associated with diverse scenarios (risk-neutral and risk-averse considerations). Summarizing, the new proposed track can be described as follows:

- day-ahead ERM model considering uncertainties of renewable generation, load consumption, local electricity market (LEM) prices, and electric vehicles (EVs) trips.
- the incorporation of risk analysis strategies using  $VaR$  and  $CVaR$  measures to deal with the uncertainty of parameters and get solutions that protect the aggregator against extreme scenarios.
- **participants will implement solution methods based on modern metaheuristic optimization to deal with the computational burden of the consideration of diverse possible scenarios of uncertain parameters and the large number of variables considered.**
- As a results, it is interesting to analyze the impact of  $VaR$  and  $CVaR$  measures over a set of case studies using realistic data in power and energy systems.

Figure 1 illustrates the proposed track. The reader can be referred to the appendix section for more details of the formulation.



## 2.2 Testbed 2

Long-term transmission network expansion planning (TNEP) is a classic problem of power systems. The objective is to find the optimal expansion plan that identifies the transmission lines that must be installed in the electrical system to allow adequate operation within a predefined planning horizon with the lowest investment cost. The data for this problem are the current topology of the network, the transmission lines that can be constructed in the candidate corridors, the generation, and the demand for a planning horizon. The optimal expansion plan should define where and how many new lines should be constructed.

Several mathematical models and optimization techniques have already been used to solve the TNEP problem. The direct current (DC) model is the most used formulation to represent the network and electrical quantities. However, there are more relaxed models, such as the transportation model, and more accurate models, such as the alternating current (AC) model. The DC model, which is a nonconvex mixed-integer nonlinear programming problem, is used to formulate the problem in this testbed. More information related to the DC model for TNEP can be found in the appendix.

The objective function consists of minimizing the investment cost and the total load shedding cost. To be considered feasible, a solution must present a load shedding equal to zero. The constraints of the problem include the equations related to the calculation of the DC optimal power flow including the possibility of expanding the system.

In this testbed, the objective is to solve a very complex instance of the TNEP problem: the northeast Brazilian system, for which the optimal solution is yet not known. The participants will implement solution methods based on modern metaheuristic optimization to deal with the combinatorial search space of the problem [6].

### 3. Metaheuristic simulator framework

In this competition, the method of choice used by the participants to solve the problem must be a metaheuristic-based algorithm (e.g., Differential Evolution, Particle Swarm Optimization, Vortex Search, Hybrid approaches, etc.). The framework adopted in the competition is described in this document and follows the structure presented in **Figure 2**.

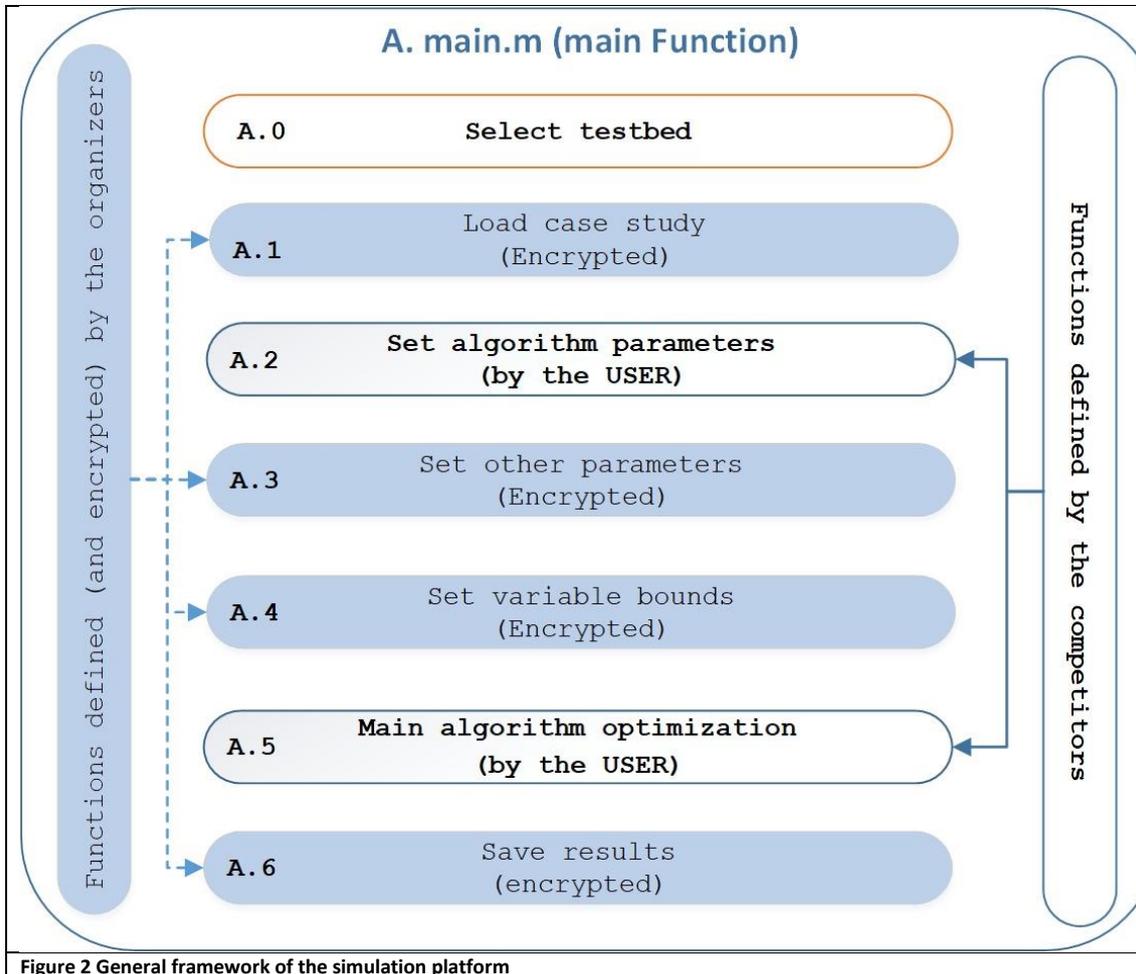


Figure 2 General framework of the simulation platform

The simulation platform has been implemented in MATLAB® 2018 64-bit and consists of different scripts with specific targets in the simulation. As shown in **Figure 2**, some scripts correspond to encrypted files provided by the organizers (blue color in the figure). The user only needs to implement two scripts (see **Sect. 4.A.2** and **Sect. 4.A.6**), namely:

- i. one script for setting the parameters required by their algorithm (A.2).
- ii. a second script for the implementation of their proposed solution method (A.6).

Examples of how to implement these two script functions, and how the organizer’s scripts work on the platform, are provided in **Sect. 4**.

A maximum number of 5,000 and 20,000 function evaluations are allowed in testbed 1 and testbed 2, respectively. Take into account that one function evaluation corresponds to each time that one solution is evaluated in the fitness (this is not the same as algorithm iterations).

### 3.A) Encoding of the individual

#### Testbed 1

One fundamental aspect of population-based algorithms is the encoding of the solutions. Depending on the problem, particles/vectors must contain all the information associated with a solution and should be evaluated in a fitness function in order to measure their performance.

The initial solutions should be initialized randomly between the upper and lower bounds of the variables. Heuristics and special tweaks for initial solutions are not accepted.

The solution structure (e.g., an individual in DE, a particle in PSO, or genotype in GA) is a fundamental part of the metaheuristics to represent a given solution. The solution representation for testbed 1 competition follows the vector representation showed in Figure 3.

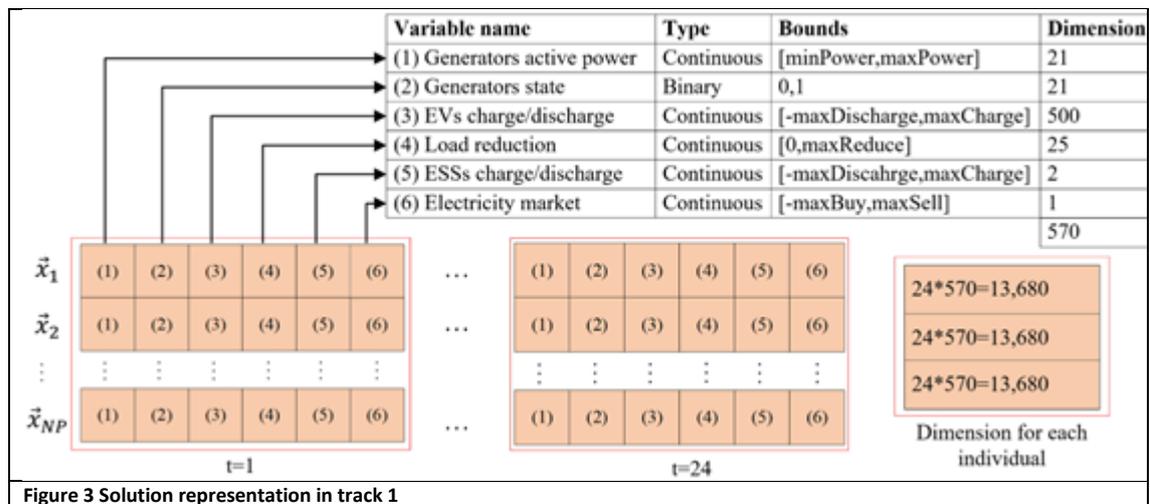


Figure 3 Solution representation in track 1

The metaheuristic's initial solutions should be created randomly between the maximum and minimum values set for each variable. Each solution is composed by a group of sequentially repeated variables per each of the 24 periods (t=1, t=2, ..., t=24). The variables in each group are defined by the technologies present in the distribution network. All variables are of continuous type varying according to the specified bounds, except for the state of the generators, which is represented by a binary state (0 indicates that the generator is not connected to the grid, and 1 that is connected).

The first group of variables represents the active power generation (including renewable production). The photovoltaic and wind generation cannot be controlled due to solar and wind factors, respectively. Even though the variables associated with this type of generation are considered in the solution vector, it is essential to note that these variables have an unchangeable value that depends on the uncertainty scenario, so the bounds of these variables are limited to the maximum production value. EVs and energy storage systems (ESSs) solutions are set to vary from the maximum active power discharge to the maximum active power charge, assuming the discharging as a negative variable (generation) and the charging as a positive variable (consumption). The demand response (DR) here is only assumed as a load reduction (despite in some other cases a load increase program could be considered), so the DR variable only varies from 0 to the maximum active power reduction. When it comes to LEM variables, it is assumed that a negative value is the power bought in the market, and a positive variable value is the power sold.

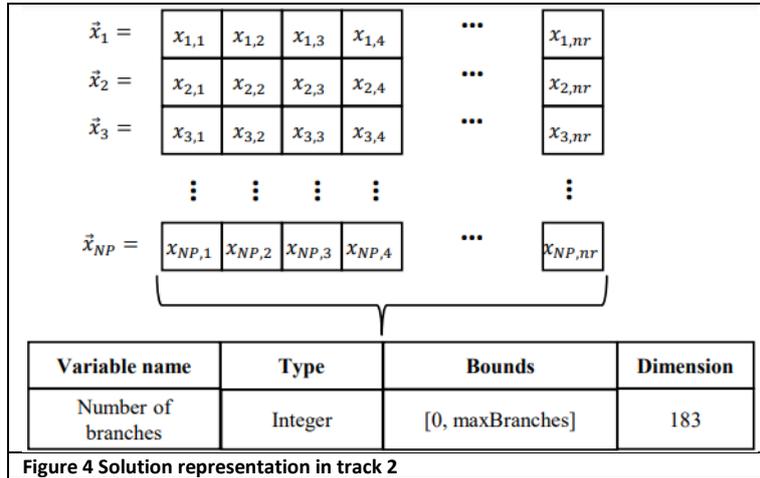
The considered ERM has 13,680 variables per individual given by 570 variables per period, with 21 variables composing the generators' active power, another 21 binary variables representing the generators' state. A total of 500 EVs were also considered with 25 load types, 2 ESSs, and 1 LEM.

#### Testbed 2

For the second testbed, the encoding is similar to testbed 1. However, only integer variables are present in a solution proposal. Figure 4 shows the representation of the solution vector for testbed 2.

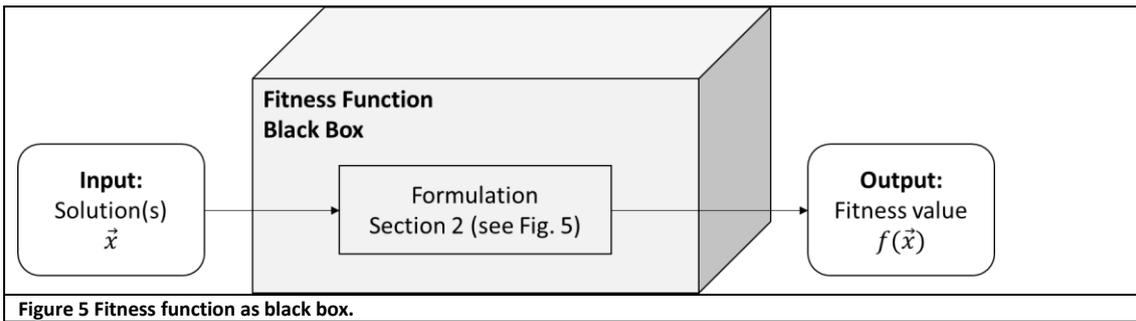
Equal to testbest 1 in testbed 2, the initial solutions should be initialized randomly between the upper and lower bounds of the variables. Heuristics and special tweaks for initial solutions are not accepted.

Each solution proposal (structure) contains the number of lines to be constructed on the branches (corridors) of the transmission system. The northeast Brazilian system has 183 corridors, therefore, a solution proposal,  $\vec{x}_i$ , will present 183 columns. The value in each column represents the number of lines that will be installed on the corresponding corridor with the rows corresponding to the number of individuals in the population. Each value must be randomly generated by the metaheuristic between the bounds  $[0, \text{maxBranches}]$ . The metaheuristics generate a random continuous value for each variable, but a **round()** function is applied, so the variables become integer



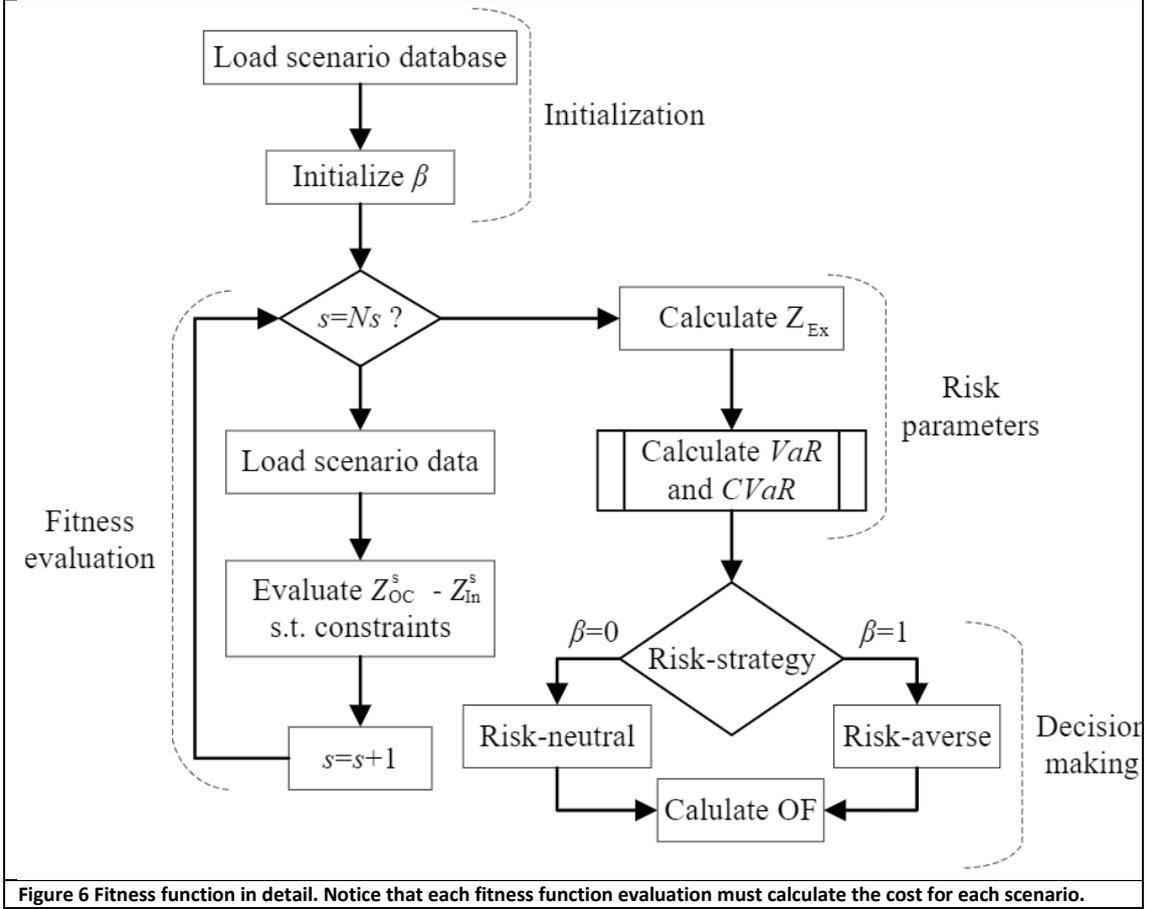
### 3.B) Fitness function

A maximum number of 5,000 and 20,000 function evaluations for testbed 1 and testbed 2, respectively, are allowed in the competition. Notice that the participant can consider the problem as a black box, in which each solution evaluated in the fitness function has a single cost associated to it, as showed in **Figure 5**.



### Testbed 1

For participants that want more details on the design of the fitness function for testbed 1, please be referred to [7], and take the following explanation into account. **Figure 6** shows the fitness function that the chosen metaheuristic evaluates for cost minimization. Initially, the database with the formulated scenarios is passed as an argument to the function, and the value of the variable that controls risk aversion is also initialized. Next, each scenario is evaluated according to the equations in the appendix section. This evaluation is done to obtain the cost of each scenario, which is saved to calculate the expected cost.



The expected cost, the cost of each scenario, and the probabilities of each scenario are used to calculate the  $VaR_\alpha$  and  $CVaR_\alpha$  values according to the formulation in [8]. After the parameters that measure risk have been calculated, the aggregator enters a decision process according to the risk aversion factor. That is, through the value of the OF, the aggregator chooses the best strategy.

The metaheuristic does this evaluation in order to minimize the value of the OF in a given number of iterations, so when  $\beta = 0$ , the metaheuristic will only minimize the expected cost. Still, when the  $\beta = 1$ , the metaheuristic tries to minimize the expected cost as well as the  $CVaR_\alpha$  [3].

A risk-neutral strategy considered for the day-ahead ERM considers the uncertain behavior of an aggregator's technologies such as renewable generation, load consumption, market prices, EV consumption behavior. In this case, the stochastic behavior of these parameters is considered in the approach used through various scenarios with an associated probability of occurrence.

When the risk is not considered, this aggregator's scheduling is formulated based on the expected scenario. The cost and the value of the objective function when a risk aversion strategy is not considered is given by the expected cost, and its formulation is given by:

$Z_s^{\text{tot}} = Z_s^{\text{OC}} - Z_s^{\text{In}} + P_s$	(1)
$Z^{\text{Ex}} = \sum_{s=1}^{N_s} (\rho_s * Z_s^{\text{tot}})$	(2)

where  $Z_s^{\text{tot}}$  is the total objective function (OF) value of each scenario  $s$  given by the difference between operational costs in each scenario ( $Z_s^{\text{OC}}$ ) and the income in each scenario ( $Z_s^{\text{In}}$ ) and the penalty for bound violations of any variable ( $P_s$ ). The expected OF cost is represented by  $Z^{\text{Ex}}$ , and  $\rho_s$  is the probability of the respective scenario.

A risk-aversion strategy considers the risk associated with the uncertainty of the previously mentioned technologies.  $CVaR_\alpha$  is an additional cost that is added to  $Z^{Ex}$  in  $(1 - \rho)\%$  of the scenarios with the highest costs. After calculating the value of  $VaR_\alpha$ , the  $CVaR_\alpha$  is calculated using the following formulation [8]:

$CVaR_\alpha(Z_s^{tot}) = VaR_\alpha(Z_s^{tot}) + \frac{1}{1 - \alpha} \sum_{s=1}^{N_s} (\rho_s * \varphi)$	(3)
---	-----

where:

$\varphi = \begin{cases} Z_s^{tot} - Z^{Ex} - VaR_\alpha(Z_s^{tot}) & \text{if } Z_s^{tot} \geq Z^{Ex} + VaR_\alpha(Z_s^{tot}) \\ 0 & \text{otherwise} \end{cases}$	(4)
$VaR_\alpha(Z_s^{tot}) = z - score(\alpha) * std(z_s^{tot})$	

$\varphi$  is a parameter associated with the cost in the worst scenarios; that is, when the cost of each scenario  $s$  exceeds the expected cost with the addition of the  $VaR_\alpha$  value. If the opposite occurs,  $\varphi = 0$ . z-score is calculated using **norminv()** function in MATLAB with  $\alpha = 95\%$ .

Taking this parameter into account, the fitness value (and OF) of the scheduling problem varies according to the level of risk aversion considered. The model of the fitness value (or OF) in this situation is then given by:

$OF = Z^{Ex} + \beta * CVaR_\alpha(Z_s^{tot})$	(5)
--	-----

In this situation, the  $\beta$  parameter represents the percentage of aversion to the risk. This parameter can vary between 0 and 1. When  $\beta = 0$ , the  $OF$  value is only equal to the expected cost, which is a risk-neutral strategy. On the contrary,  $\beta = 1$  means that the strategy has 100% aversion to risk presenting the safest solution when it comes to the worst scenarios.

"In this competition,  $\beta = 1$  is set as default". Finally, the equations for operational costs ( $Z_s^{OC}$ ) and incomes ( $Z_s^{In}$ ) are provided in the appendix section.

## Testbed 2

The formulation of the TNEP problem considering the DC model is presented in the appendix. In the formulation, artificial generators are considered at each demand bus, with a generation capacity equal to the corresponding active power demand of the bus in which it is installed. The generation price of an artificial generator is  $\Upsilon$  MUS\$/MW. The total generation level of all the artificial generators represents the load shedding of an expansion proposal. Therefore, it is a measure of infeasibility of a solution proposal. To be feasible, a solution proposal must present a total load shedding equal to zero.

If the number of lines to be constructed in each corridor is known (determined by the metaheuristic) the TNEP model becomes a linear programming (LP) problem, in which the objective function is to minimize the generation cost from the artificial generators (see the appendix). This resulting model, when the number of lines in each corridor is fixed, is a DC OPF problem.

MATPOWER [9] is used to solve this resulting DC OPF problem for each solution proposal to determine the total load shedding. Note that the investment cost can be easily calculated by  $\vec{C}\vec{x}_i$ , where  $\vec{C}$  is the vector of costs and  $\vec{x}_i$  is the  $i^{\text{th}}$  solution proposal. Let  $r_j$  be the load shedding at bus  $j$ , then the value of the objective function,  $v''(\vec{x}_i)$  is presented in Eq. (6).

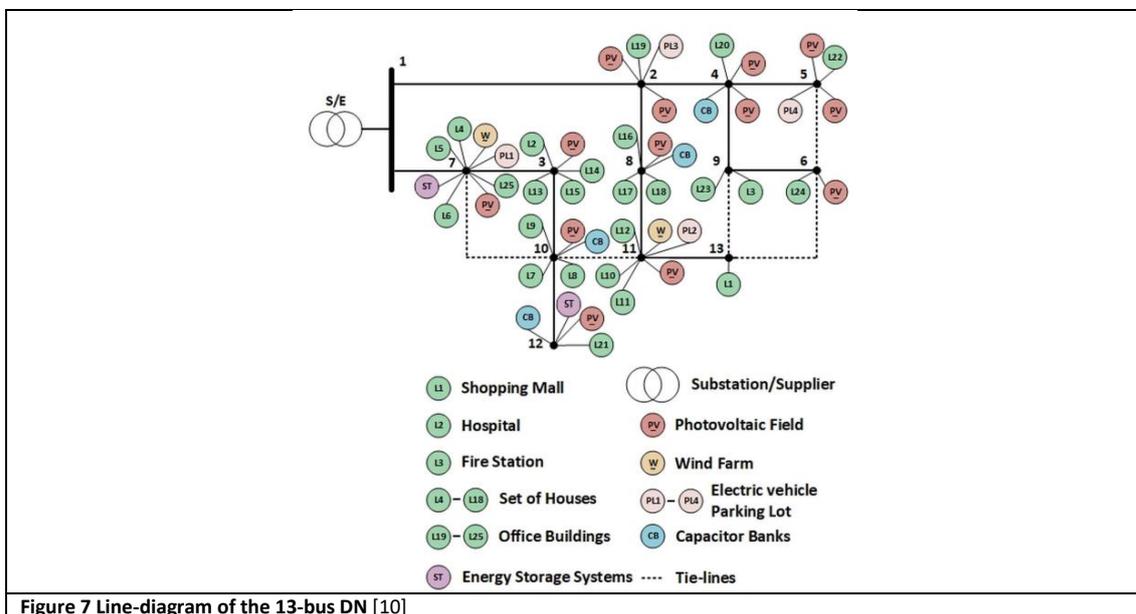
$v''(\vec{x}_i) = \vec{C}\vec{x}_i + \Upsilon \sum_{j \in \Omega_N} r_j$	(6)
--	-----

In the proposed formulation, every feasible solution  $\vec{x}_i$  will present a total load shedding equal to zero, while infeasible solutions will present a total load shedding (or artificial generation value) greater than zero. Moreover, note that the DC OPF problem will always be feasible, since it is possible to attend the demand through the artificial generation.

### 3.C) Scenario overview

#### Testbed 1

A medium voltage (MV), distribution network (DN) of an smart city located in the BISITE laboratory in Salamanca, Spain, was chosen for this case study [10]. This DN features one 30MVA substation in bus 1, 15 DG units (2 wind farms and 13 PV parks), and four 1Mvar capacitor banks. When it comes to consumption, this DN has 25 different loads composed of residential and office buildings and some buildings that provide a service (hospital, fire station, and shopping mall). The smart city in question has seven charging stations allowing EVs to charge their batteries, including four 7.2kW slow charging stations per connection point and 50kW fast charging stations per connection point. The single-line diagram presented in **Figure 7** of the 13-bus 30-kV shows all the technologies listed. This case study considers the high penetration of EVs and renewables.



The initial dataset with 5000 scenarios is then reduced to 150 scenarios using GAMS/SCENRED. It is also later reduced to 15 scenarios to minimize computational time due to the high number of variables. **Figure 8 a)** shows the scenarios representing the uncertainty associated with the load consumption, where scenario seven (extreme case two) presents higher values, mainly from hours 16 to 22, compared to the remaining scenarios. **Figure 8 b)** gives the scenario representation of the wind generation, where a large decrease is noticed in scenario one, which also corresponds to a zero total PV production (**Figure 8 c)**). This scenario corresponds to the first extreme event generated. The first extreme scenario also shows an increase in LEM market prices throughout when compared to the other scenarios, as shown in **Figure 8 d)** The other extreme scenario is scenario eleven, which just affects the electricity market capacity, and renewable generation (only a small percentage).

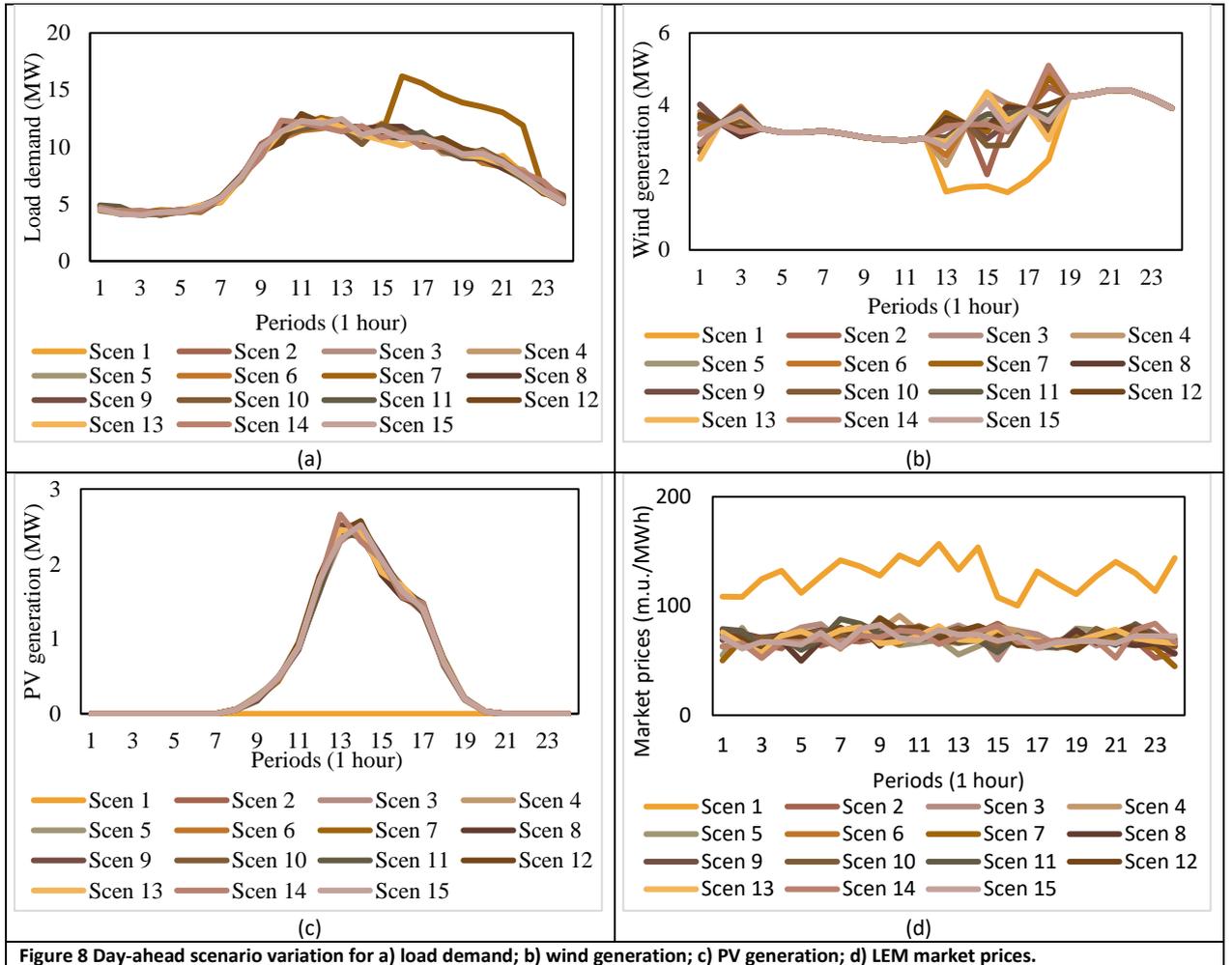


Figure 8 Day-ahead scenario variation for a) load demand; b) wind generation; c) PV generation; d) LEM market prices.

The aggregator must manage its respective resources, power bought from the external supplier, and energy bought/sold in the marketplaces to satisfy the consumption. Also are integrated into the DN four capacitors but are not considered in this problem, so they are set to zero because we did not consider reactive power in the problem formulation. Two ESSs units were also considered in the situation. **Table 1** presents the energy resources data associated with the aggregator for the day-ahead formulation considering the extreme scenarios. A distinction is made for the minimum and maximum values considered for the prices of the resources, capacity, forecasted values from the renewables and loads, and the number of units corresponding to each resource.

Energy Resources		Prices (m.u./MWh)	Capacity (MW)	Forecasted (MW)	Units
		min-max	min-max	min-max	
Wind		29-29		0.00-0.81	2
Photovoltaic		31-31		0.30-3.07	13
External supplier		50-90	0.00-30.00		1
ESSs	Charge	110-110	0.00-1.25		2
	Discharge	60-60	0.00-1.25		
EVs	Charge	0-0	0.01-0.05		500
	Discharge	60-60	0.01-0.05		
Demand response	Load reduction	100-100	0.00-1.21		25
Load		0-0		0.01-2.38	25
LEM		44.78-156.91			1

## Testbed 2

The northeast Brazilian system was chosen for this case study. It consists of 87 buses and 183 corridors. The total generation of the system is 29,748 MW, which is equal to the total demand, since generation rescheduling is not considered and losses are disregarded. The values of generations and demands are obtained from the original system by multiplying the values by 1.2. Of the 183 corridors, 71 have existing lines, while 112 have no existing lines. Up to 15 lines are allowed to be constructed in each corridor.

Figure 9 illustrates the northeast Brazilian system. A dotted line represents a corridor without existing lines while each continuous line represents a transmission line. New lines are allowed to be constructed in every corridor, with and without existing lines. The buses are numbered from 1 to 87. An arrow entering a bus represents a generation plant, while an arrow leaving a bus represents a power demand.

As mentioned previously, an artificial generation is considered at each load bus, representing a load shedding, with a capacity equal to the power demand at the bus. To each artificial generator, a generation cost (penalization for load shedding) of  $\gamma = 1e^9$  MUS\$/MW is assigned. Existing generation plants have a generation cost equal to zero.

The expansion plan provided by the metaheuristic should present, therefore, a total artificial generation (load shedding) value equal to zero to be considered feasible.

Finally, note that each corridor  $ij$  has different values for  $N_{ij}^0$ , the number of existing lines on corridor  $ij$ ,  $X_{ij}$ , the reactance of a line on corridor  $ij$ ,  $\bar{F}_{ij}$ , the capacity of a line on corridor  $ij$ , and  $C_{ij}$ , the cost of constructing a line on corridor  $ij$

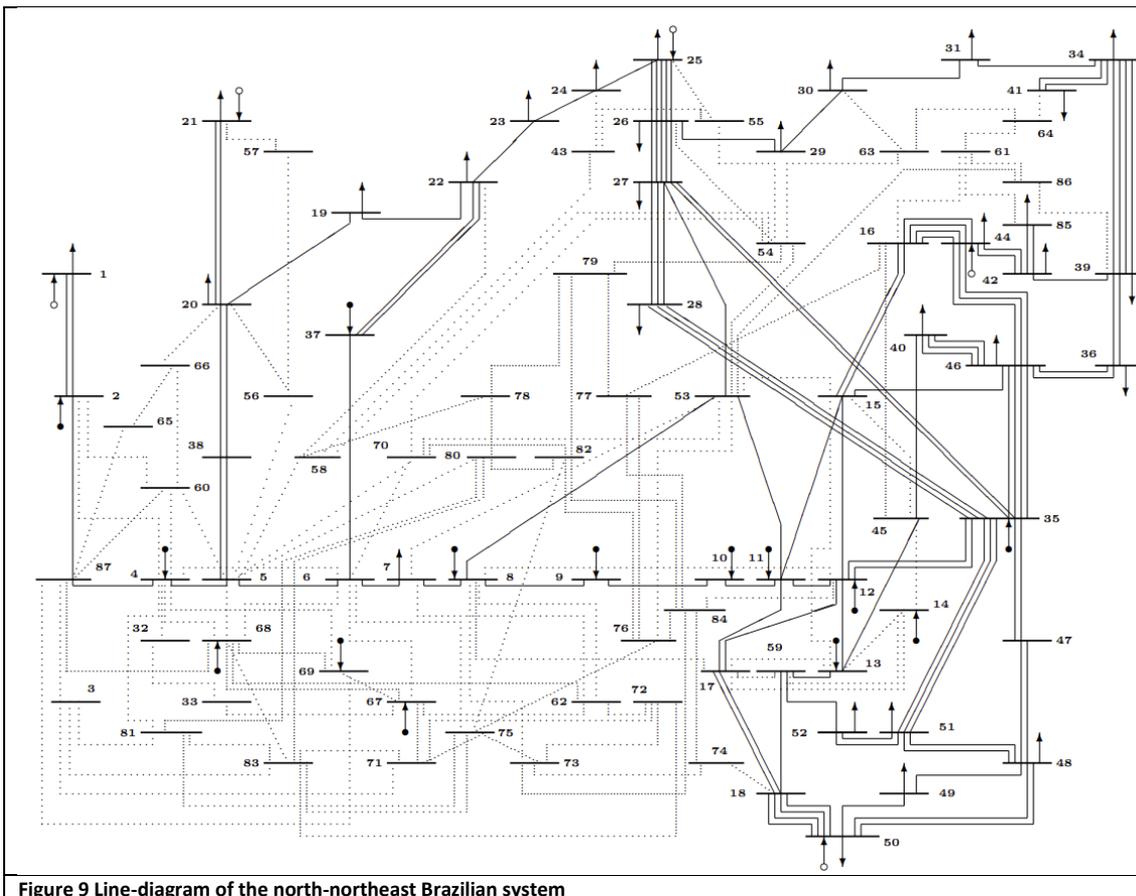


Figure 9 Line-diagram of the north-northeast Brazilian system

**NOTE: For this case study the installation of MATPOWER is needed (install the file highlighted in Figure 10). The installation files are provided in the competition platform (see README.txt for the installation process).**

📁 .github	09/10/2020 00:47	Pasta de ficheiros	
📁 .travis	09/10/2020 00:47	Pasta de ficheiros	
📁 data	09/10/2020 00:47	Pasta de ficheiros	
📁 docker	09/10/2020 00:47	Pasta de ficheiros	
📁 docs	09/10/2020 00:47	Pasta de ficheiros	
📁 extras	09/10/2020 00:47	Pasta de ficheiros	
📁 lib	09/10/2020 00:47	Pasta de ficheiros	
📁 mips	09/10/2020 00:47	Pasta de ficheiros	
📁 most	09/10/2020 00:47	Pasta de ficheiros	
📁 mp-opt-model	09/10/2020 00:47	Pasta de ficheiros	
📁 mptest	09/10/2020 00:47	Pasta de ficheiros	
📄 .dockerignore	09/10/2020 00:47	Ficheiro DOCKERI...	1 KB
📄 .gitignore	09/10/2020 00:47	Ficheiro GITIGNORE	1 KB
📄 .travis.yml	09/10/2020 00:47	Ficheiro YML	6 KB
📄 AUTHORS	09/10/2020 00:47	Ficheiro	1 KB
📄 CHANGES.md	09/10/2020 00:47	Ficheiro MD	122 KB
📄 CITATION	09/10/2020 00:47	Ficheiro	4 KB
📄 CONTRIBUTING.md	09/10/2020 00:47	Ficheiro MD	13 KB
📄 install_matpower.m	09/10/2020 00:47	Ficheiro M	14 KB
📄 LICENSE	09/10/2020 00:47	Ficheiro	3 KB
📄 README.md	09/10/2020 00:47	Ficheiro MD	18 KB

Figure 10 File for MATPOWER installation

## 4. Guidelines for participants

These instructions include as example the metaheuristic hybrid-adaptive differential evolution (HyDE) [11] implemented and adapted to the present framework (It has been modified by GECAD).

It is important that the participants use the following recommendations and structure to avoid issues in using the supplied datasets and codes.

### 4.A) *main.m* - Master function/script

**# main.m** is the main file for the competition. The competitors can modify this main script as needed. Nevertheless, it is worth noting that this main script is ready to use. Participants should only include their functions to perform the optimization of the problem.

<i>main.m</i>	
<pre> %% %% Select testbed Select_testbed=1; %Testbed 1: Risk-based energy scheduling %Testbed 2: Transmission expansion planning DB=Select_testbed; % 1: Case study testbed 1 % 2: Case study testbed 2  [caseStudyData, DB_name]=callDatabase(DB);  Select_Algorithm=1; %1: HyDE algorithm (test algorithm) %2: Your algorithm algorithm='HyDE'; %'The participants should include their algorithm here' <b>DEparameters %Function defined by the participant</b> No_solutions=deParameters.I_NP; % %Notice that some algorithms are limited to one individual </pre>	A.0
<pre> [caseStudyData, DB_name]=callDatabase(DB); </pre>	A.1
<pre> Select_Algorithm=1; %1: HyDE algorithm (test algorithm) %2: Your algorithm algorithm='HyDE'; %'The participants should include their algorithm here' <b>DEparameters %Function defined by the participant</b> No_solutions=deParameters.I_NP; % %Notice that some algorithms are limited to one individual </pre>	A.2

<pre> %% %% Set other parameters otherParameters =setOtherParameters(caseStudyData,No_solutions, Select_testbed); </pre>	A.3
<pre> %% %% Set lower/upper bounds of variables [lowerBounds,upperBounds] = setVariablesBounds(caseStudyData,otherParameters); </pre>	A.4
<pre> %% %% Call the MH for optimization ResDB=struc([]); for iRuns=1:noRuns %Number of trails     tOpt=tic;     rand('state',sum(noRuns*100*clock))% ensure stochastic indpt trials      otherParameters.iRuns=iRuns;     switch Select_Algorithm         case 1             [ResDB(iRuns).Fit_and_p, ...              ResDB(iRuns).sol, ...              ResDB(iRuns).fitVector]= deopt_simple(deParameters,caseStudyData,otherParameters,lowerB,upperB);             ...         ...     end </pre>	A.5
<pre> %% %% Save the results and stats Save_results </pre>	A.6
<pre> ... </pre>	

As it can be seen, the main script follows the structure from Figure 2 (Sect. 3). Details in the implementation of each part of the code are given next.

### A.0 and A.1 - #main.m - Loading the testbed and case study

**#main**– This is the main framework file where you can select the testbed, which will load the caseStudyData struct (callDatabase.p – encrypted) with all the relevant dataset information depending on the selected testbed. Participants do not need to worry about the content of the case study and loading the files.

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Select_testbed=1;
%Testbed 1: Risk-based energy scheduling
%Testbed 2: Transmission expansion planning
%% Load Data base
noRuns=20; %this can be changed but final results should be based on 20 trials
DB=Select_testbed;
% 1: Case study testbed 1
% 2: Case study testbed 2
[caseStudyData, DB_name]=callDatabase(DB);

```

### A.2 - #DEparameters.m - Set parameters of the metaheuristic

**#DEparameters.m** file – This function file must be specific to the metaheuristic implemented by the participant. This is just an example using DE to show how participants should implement this function with all the parameters related to their algorithm.

```

deParameters.I_NP= 10; % Size of the population in DE
deParameters.F_weight= 0.3; %Mutation factor
deParameters.F_CR= 0.5; %Recombination constant
deParameters.I_itermax= 500; % number of max iterations/gen
deParameters.I_strategy = 1; %DE strategy

deParameters.I_bnd_constr = 1; %Using bound constraints
% 1 repair to the lower or upper violated bound
% 2 rand value in the allowed range
% 3 bounce back

```

### A.3 - #setOtherParameters.m - Set other necessary parameters and struct

**# setOtherParameters.m (encrypted)** – This file is encrypted and should not be changed or modified by the user. It just sets parameters and data needed for the fitness function to work. **Please take into account a third parameter is added to the previous framework to consider another track.** It is a mandatory function that creates a struct “otherParameters” and should be run as illustrated in main function section:

```
%% Set other parameters
otherParameters =setOtherParameters(caseStudyData,No_solutions, Select_testbed);
```

Participants must pass the “otherParameters” struct as argument to the functions:

```
[lowerBounds,upperBounds]=
setVariablesBounds(caseStudyData,otherParameters, Select_testbed);
...
[ResDB(iRuns).Fit_and_p, ...
    ResDB(iRuns).sol, ...
    ResDB(iRuns).fitVector]=
HyDE(deParameters,caseStudyData,otherParameters,lowerB,upperB);
```

### A.4 - #setVariablesBounds.m - Set bounds of variables

**# setVariablesBounds.m (encrypted)** – This file is encrypted and should not be changed or modified by the user. It just sets the bounds of the problem variables. **Please take into account a third parameter is added to the previous framework to consider another track.**

```
%% Set lower/upper bounds of variables
[lowerBounds,upperBounds]=
setVariablesBounds(caseStudyData,otherParameters,Select_testbed);
```

The outputs of this function “[lowerBounds,upperBounds]” – should be used by your algorithm to generate the initial solutions and to validate if the bounds are being respected in each iteration.

The order of the variables in the implemented codes cannot be modify for the proper functioning of the fitness function. The structure of the solution is indicated in **Sect. 3.A** of this document.

### A.5 - #HyDE.m - Algorithm proposed by the competitor

The participants should generate a scrip called **#MAlgorithm.m** or similar. This algorithm should replace **#HyDE.m** which is provided as example:

```
[ResDB(iRuns).Fit_and_p, ...
    ResDB(iRuns).sol, ...
    ResDB(iRuns).fitVector]=
HyDE(deParameters,caseStudyData,otherParameters,lowerB,upperB);
```

Your metaheuristic should receive as input parameters:

1. **deParameters**: struct with the parameters configuration for your algorithm to work (it is generated by the user)
2. **caseStudyData**: struct with the information of the case study
3. **otherParameters**: struct with additional information required by the fitness function
4. **lowerB/upperB**: lower and upper bounds of variables

Your metaheuristic code should return to the main script the following variables:

1. **ResDB(iRuns).fit\_and\_p**: array of size 1x2 with the best fitness and penalties values
2. **ResDB(iRuns).sol**: vector of size: 1 x noVariables with the best candidate solution found by your algorithm
3. **ResDB(iRuns).fitVector**: array of size: 2xnolterations with the value of the fitness and penalties over the iterations.

The participants are encouraged to save the results of each trial/run in a struct “ResDB”, as shown in the example. That will ease the evaluation process by the organizers.

### A.6 - #Save\_results.m - Benchmark results (text-files)

#Save\_results.m (encrypted) – The output is written to text-files using this script. The following tables should be produced:

**Table 2. Table\_Time:** Computing time spent for all optimization trials (benchmark\_Time.txt)

	timeSpent (s)
Run1	
Run2	
Run3	
...	
Run20	

**Table 3. Table\_Fitness:** Individual benchmark of the fitness in each iteration (benchmark\_Fitness.txt)

	Fitness
Run1	
Run2	
Run3	
...	
Run20	

**Table 4. Table\_Results:** Individual benchmark of the trials for testbed 1 (benchmark\_Results.txt)

	OF	$Z^{Ex}$	VaR	CVaR	Worst scenario	Bound violations	ConvergenceRate
Run1							
Run2							
Run3							
...							
Run20							

**Table 5. Table\_Results:** Individual benchmark of the trials for testbed 2 (benchmark\_Results.txt)

	OF	LScost	investmentCost	ConvergenceRate
Run1				
Run2				
Run3				
...				
Run20				

**Table 6. Table\_TrialStats:** Summary statistics or the trials (benchmark\_Summary.txt)

Ranking Index	Standard deviation	Minimum	Maximum	Variance	Average time	Code
RankingIndex	PstdOF	PminOF	PmaxOF	PvarOF	AvgTime	validationCode

In addition, this function should automatically generate the file “*solutionRiskERM.mat*”, which should include the best solutions found in each of the trials. That file will be used to double-check the reported results by validating all the solutions contained of the case study. For that reason, it is important that the participants put special care in returning the best solutions from their algorithms and stored in “*ResDB.sol*” (see Sect. 4.A.6).

To clarify, the “*send2Organizers\_TX.mat*” file will include a matrix called “*solutions*” with the solutions stored in “*ResDB.sol*”. The solutions there will be evaluated according to Sect. 5 in order to double check the ranking index of each participant. The lower the ranking index, the better the performance of a participant.

**\*A number 20 trials should be made.**

**\*5,000 evaluations per trial for testbed 1.**

**\*20,000 evaluations per trial for testbed 2.**

## 4.B) Fitness function evaluation

**#fitnessFun\_riskERM.m and #fitnessFun\_transEP (encrypted)** – this is the fitness function to be used by participants and should be called as below. The “fnc” parameter will be assigned automatically to load the corresponding fitness function according to the selected testbed **Sect. 4.A.0**.

```
[S_val, ~]=feval(fnc, FM_pop, caseStudyData, otherParameters);
```

The function receives as input:

1. **fnc**: string with the fitness function m file name: and “fitnessFun\_riskERM.m” for testbed 1 and “fitnessFun\_transEP” for testbed 2.
2. **FM\_pop**: matrix of size  $N_{sol} \times D$ , in which  $N_{sol}$  (rows) represents the number of individuals/solutions in an array, and  $D$  (columns) represents the dimension (i.e., number of variables) of the optimization problem. This variable should be encoded in the metaheuristic algorithm proposed by participants (e.g., **#MAlgorithm.m, Sect. 4.A.5**). Only 1 individual is also possible (one row).
3. **caseStudyData**: struct with data of the case study with all the scenarios as loaded by callDatabase function (i.e., **#callDatabase.m, Sect. 4.A.1**).
4. **otherParameters**: Struct with additional information as loaded by **#setOtherParameters.m Sect. 4.A.3**.

The function returns as output:

1. **S\_val**: Matrix of size  $N_{sol}$  represents the number of individuals. This matrix includes the fitness values including penalties of the solutions.

The **#fitnessFun** evaluates all the population (individuals) at once. A maximum number of 5,000 function evaluations is set for this competition. The table below helps the participant to have an idea of the maximum number of iterations and population It can set without surpassing the allowed number of evals. So, take it account when designing your algorithm:

**Table 7. Algorithm population/iterations limits**

Size of the population	Max. iterations Track 1	Max. iterations Track 2
1	5,000	20,000
5	1,000	4,000
20	250	1,000
50	100	400
100	50	200
1000	5	20

## 5. Evaluation guidelines

A ranking index will be calculated using the 20 final solutions (one for each trial) provided by each participant. With these solutions, the organizers will calculate the ranking index ( $RI_{user}$ ) for each participant  $a$  based on the average fitness of solutions in this competition. The values  $RI_{user(a),T1}$  will be normalized and a final ranking will be produced.

$$RI_{user(a),T1} = \frac{1}{N_{trials}} \cdot \left[ \sum_{i=1}^{N_{trials}} (Fit_a(\vec{X}_{i,T1})) \right] \quad (6)$$

where  $Fit_a(\vec{X}_{i,T1})$  is a function that returns the fitness value of the solution found in trial  $i$  (i.e.,  $\vec{X}_i$ ) by participant  $a$  (See [Sect. 3.B](#)).

Therefore, the winner of the competition will be the one that gets the minimum value of  $RI_{user}$  (minimization problems). The participants must consider this criterion while selecting the best search strategy in their algorithms.

## 6. Material to be submitted to the organizers

For the validation of the results, the 4 benchmark text files and the “`send2organizers_TX.mat`” file produced by `# Save_results.m` (see [Sect. 4.A.6](#)) should be submitted to the organizers. The implementation codes of each algorithm entering the competition must also be submitted along with final results for full consideration in the evaluation. The submitted codes will be used for further tests, which are intended to crosscheck the submitted results (Note: this evaluation could consider the modification of the case study and number and the encoding of variables, so that algorithms should be designed generally enough to handle different case studies of the same problem). The submitted codes will be in the public domain and no intellectual property claims should be made.

Each participant is kindly requested to put the text files corresponding to final results, as well as the implementation files (codes), obtained by using a specific optimizer, into a zipped folder named:

`CEC2023_testbedX_AlgorithmName_ParticipantName.zip`  
(e.g., `CEC2023_testbed2_DE_Lezama.zip`).

The zipped folder must be submitted to [jan@isep.ipp.pt](mailto:jan@isep.ipp.pt); [flz@isep.ipp.pt](mailto:flz@isep.ipp.pt), [jorga@isep.ipp.pt](mailto:jorga@isep.ipp.pt) by 1th June 2023 (anywhere on Earth)

## Appendix: Mathematical formulation

### Testbed 1

The day-ahead ERM's mathematical formulation takes into account the whole operation cost and profits in each scenario  $s$ . As a result, the total operational costs and profits for each scenario are calculated as follows [7]:

$Z_s^{OC} = \sum_{t=1}^T \left( \begin{aligned} & \sum_{i \in \Omega_{DG}^d} P_{(i,t)}^{DG} \cdot C_{(i,t)}^{DG} + \sum_{k=1}^{N_k} P_{(k,t)}^{ext} \cdot C_{(k,t)}^{ext} + \\ & \sum_{i \in \Omega_{DG}^{nd}} P_{(i,t,s)}^{DG} \cdot C_{(i,t)}^{DG} + \sum_{l=1}^{N_l} P_{(l,t,s)}^{Curt} \cdot C_{(l,t,s)}^{Curt} + \\ & \sum_{e=1}^{N_e} ESS_{(e,t,s)}^{Cost} + \sum_{v=1}^{N_v} EV_{(v,t,s)}^{Cost} + \\ & \sum_{r=1}^{N_r} P_{(r,t,s)}^{ENS} \cdot C_{(r,t)}^{ENS} + \sum_{i=1}^{N_i} P_{(i,t,s)}^{GCP} \cdot C_{(i,t)}^{GCP} \end{aligned} \right) \cdot \Delta t \quad \forall s \quad (7)$
$Z_s^{In} = \sum_{t=1}^T \left( \sum_{m=1}^{N_m} P_{(m,t)}^{Market} \cdot MP_{(m,t,s)} \right) \cdot \Delta t \quad \forall s \quad (8)$

where:

$ESS_{(e,t,s)}^{Cost} = \begin{cases} ESS_{(e,t,s)}^{Power} \cdot ESSC_{(e,t)}^{Disch} & \text{if } ESS_{(e,t,s)}^{Power} \leq 0 \\ 0 & \text{otherwise} \end{cases} \quad (9)$
$EV_{(v,t,s)}^{Cost} = \begin{cases} EV_{(v,t,s)}^{Power} \cdot EV_{(v,t)}^{Disch} & \text{if } EV_{(v,t,s)}^{Power} \leq 0 \\ 0 & \text{otherwise} \end{cases} \quad (10)$

It's worth noting that  $ESS_{(e,t,s)}^{Power} / EV_{(v,t,s)}^{Power}$  are continuous variables with a negative value when discharging and a positive value when charging. As a result, in Eq. (7),  $ESS_{(e,t,s)}^{Cost}$  and  $EV_{(v,t,s)}^{Cost}$  are only considered while the battery system is discharging (Eqs. (9)-(10)). To put it another way, the aggregator must pay a fee for the electricity utilized by these technologies. The cost of charging the batteries, on the other hand, is not included in Eq. (7) because it is a payment that the aggregator receives for supplying the requested energy. Similarly, because we only evaluate market transactions in Eq. (8), these payments are not taken into account in this equation.

The suggested approach seeks to minimize Eq. (7) while maximizing Eq. (8) for each case  $s$  by integrating both components into a minimization problem. The aggregator trades energy as much as feasible to make a profit, depending on market prices and operational costs. The remaining formulations, and constraints can be seen in [7].

### Testbed 2

The DC model for the TNEP problem is presented in (11)–(17).

$\text{minimize } v = \sum_{ij \in \Omega_B} C_{ij} n_{ij} + \Upsilon \sum_{i \in \Omega_N} r_i \quad (11)$
---

Subject to:

$\sum_{ji \in \Omega_B} f_{ji} - \sum_{ij \in \Omega_B} f_{ij} + G_i + r_i = D_i \quad \forall i \in \Omega_N \quad (12)$
---

$f_{ij} = (N_{ij}^o + n_{ij}) \frac{(\theta_i - \theta_j)}{X_{ij}} \quad \forall ij \in \Omega_B \quad (13)$
--

$ f_{ij}  \leq \bar{F}_{ij} (N_{ij}^o + n_{ij}) \quad \forall ij \in \Omega_B \quad (14)$
---

$0 \leq r_i \leq D_i \quad \forall i \in \Omega_N \quad (15)$
---

$0 \leq n_{ij} \leq \bar{N}_{ij}$	$\forall ij \in \Omega_B$	(16)
$n_{ij} \in \mathbb{Z}$	$\forall ij \in \Omega_B$	(17)

In this model,  $\Omega_N$  is the set of buses and  $\Omega_B$  is the set of branches (or corridors). The parameters are:  $C_{ij}$  is the cost of constructing a line on corridor  $ij$ ,  $\Upsilon$  is the load shedding cost,  $G_i$  is the active power generation at bus  $i$ ,  $D_i$  is the active demand at bus  $i$ ,  $N_{ij}^o$  is the number of existing lines on corridor  $ij$ ,  $X_{ij}$  is the reactance of a line on corridor  $ij$ ,  $\bar{F}_{ij}$  is the capacity of a line on corridor  $ij$ , and  $\bar{N}_{ij}$  is the maximum number of lines that can be constructed on corridor  $ij$ . The variables are:  $n_{ij}$  is the number of lines constructed on corridor  $ij$ ,  $r_i$  is the load shedding at bus  $i$ ,  $f_{ij}$  is the active power flow on corridor  $ij$ , and  $\theta_i$  is the voltage phase angle at bus  $i$ .

The objective function (11) minimizes the investment cost for building new transmission lines and the total load shedding cost. Equation (12) represents Kirchhoff's current law for each bus in the system. Constraint (13) represents the application of Kirchhoff's voltage law for each fundamental loop of the system. Constraint (14) determines the transmission capacity limits of the corridors considering the numbers of existing lines and new lines. In this constraint, the use of the absolute value is necessary because the flows are bidirectional. Constraint (15) is the load shedding limit on the buses. Constraint (16) represents the limit for the number of new lines that can be installed in each corridor. Finally, (17) indicates that the variable  $n_{ij}$  must be integer, representing the greatest source of complexity in the problem. Furthermore,  $\theta_i$  must be fixed at zero at the reference bus. This is a nonconvex MINLP problem, which is also multimodal and very hard to solve for large instances.

If  $n_{ij}$  is fixed at  $n_{ij}^* \forall ij \in \Omega_B$ , then the resulting model becomes the linear programming (LP) problem presented in (18)–(22).

minimize $v' = \Upsilon \sum_{i \in \Omega_N} r_i$		(18)
Subject to:		
$\sum_{j \in \Omega_B} f_{ji} - \sum_{ij \in \Omega_B} f_{ij} + G_i + r_i = D_i$	$\forall i \in \Omega_N$	(19)
$f_{ij} = (N_{ij}^o + n_{ij}^*) \frac{(\theta_i - \theta_j)}{X_{ij}}$	$\forall ij \in \Omega_B$	(20)
$ f_{ij}  \leq \bar{F}_{ij}(N_{ij}^o + n_{ij}^*)$	$\forall ij \in \Omega_B$	(21)
$0 \leq r_i \leq D_i$	$\forall i \in \Omega_N$	(22)

In this model, the load shedding is minimized in the objective function (18). Constraints (19) and (20) are the application of Kirchhoff's laws to the system, (21) is the transmission capacity limit and (22) is the limit of load shedding at each bus. Furthermore,  $\theta_i$  must be fixed at zero at the reference bus.

This model provides the load shedding for a solution proposal  $n_{ij}^* \forall ij \in \Omega_B$  provided by the metaheuristic. Note that a solution proposal  $n_{ij}^*$  is feasible only if the corresponding value of the load shedding,  $v'$ , in (18) is zero. This is a DC optimal power flow problem (DC OPF), which can be constructed and solved using MATPOWER and an LP solver.

The objective function for a given solution proposal,  $v''$ , is shown in (23).

$v'' = \sum_{ij \in \Omega_B} C_{ij} n_{ij}^* + \Upsilon \sum_{i \in \Omega_N} r_i$	(23)
---	------

The first part of  $v''$  is calculated by multiplying each element of a solution proposal,  $n_{ij}^*$ , by the corresponding cost,  $C_{ij}$ , and adding the values. The second term, which represents the load shedding, is provided by solving an DC OPF problem using MATPOWER.

## Bibliography

- [1] G. Mavromatidis, K. Orehounig, and J. Carmeliet, "A review of uncertainty characterisation approaches for the optimal design of distributed energy systems," *Renew. Sustain. Energy Rev.*, vol. 88, pp. 258–277, May 2018, doi: 10.1016/j.rser.2018.02.021.
- [2] M. Tavakoli, F. Shokridehaki, M. Funsho Akorede, M. Marzband, I. Vechiu, and E. Poursmaeil, "CVaR-based energy management scheme for optimal resilience and operational cost in commercial building microgrids," *Int. J. Electr. Power Energy Syst.*, vol. 100, pp. 1–9, Sep. 2018, doi: 10.1016/j.ijepes.2018.02.022.
- [3] F. Samadi Gazijahani and J. Salehi, "Optimal Bilevel Model for Stochastic Risk-Based Planning of Microgrids Under Uncertainty," *IEEE Trans. Ind. Inform.*, vol. 14, no. 7, pp. 3054–3064, Jul. 2018, doi: 10.1109/TII.2017.2769656.
- [4] F. Lezama, J. Soares, P. Hernandez-Leal, M. Kaisers, T. Pinto, and Z. Vale, "Local Energy Markets: Paving the Path Toward Fully Transactive Energy Systems," *IEEE Trans. Power Syst.*, vol. 34, no. 5, pp. 4081–4088, Sep. 2018, doi: 10.1109/TPWRS.2018.2833959.
- [5] J. Soares, B. Canizes, M. A. Fotouhi Gazvhini, Z. Vale, and G. K. Venayagamoorthy, "Two-stage Stochastic Model using Benders' Decomposition for Large-scale Energy Resources Management in Smart grids," *IEEE Trans. Ind. Appl.*, pp. 1–1, 2017, doi: 10.1109/TIA.2017.2723339.
- [6] M. Rahmani, R. Romero, and M. J. Rider, "Strategies to Reduce the Number of Variables and the Combinatorial Search Space of the Multistage Transmission Expansion Planning Problem," *IEEE Trans. Power Syst.*, vol. 28, no. 3, pp. 2164–2173, Aug. 2013, doi: 10.1109/TPWRS.2012.2223241.
- [7] J. Almeida, J. Soares, F. Lezama, and Z. Vale, "Robust Energy Resource Management incorporating Risk Analysis using Conditional Value-at-Risk," *IEEE Access*, pp. 1–1, 2022, doi: 10.1109/ACCESS.2022.3147501.
- [8] M. Esmaeeli, A. Kazemi, H. Shayanfar, G. Chicco, and P. Siano, "Risk-based planning of the distribution network structure considering uncertainties in demand and cost of energy," *Energy*, vol. 119, pp. 578–587, Jan. 2017, doi: 10.1016/j.energy.2016.11.021.
- [9] R. D. Zimmerman, C. E. Murillo-Sanchez, and R. J. Thomas, "MATPOWER: Steady-State Operations, Planning, and Analysis Tools for Power Systems Research and Education," *IEEE Trans. Power Syst.*, vol. 26, no. 1, pp. 12–19, Feb. 2011, doi: 10.1109/TPWRS.2010.2051168.
- [10] B. Canizes, J. Soares, Z. Vale, and J. Corchado, "Optimal Distribution Grid Operation Using DLMP-Based Pricing for Electric Vehicle Charging Infrastructure in a Smart City," *Energies*, vol. 12, no. 4, p. 686, Feb. 2019, doi: 10.3390/en12040686.
- [11] F. Lezama, J. Soares, R. Faia, T. Pinto, and Z. Vale, "A New Hybrid-Adaptive Differential Evolution for a Smart Grid Application Under Uncertainty," in *2018 IEEE Congress on Evolutionary Computation (CEC)*, Rio de Janeiro, Jul. 2018, pp. 1–8. doi: 10.1109/CEC.2018.8477808.