Statistical Tuning of DEEPSO Soft Constraints in the Security Constrained Optimal Power Flow Problem

Leonel M. Carvalho[1], Fabio Loureiro,
Jean Sumaili[1], Hrvoje Keko[1], Vladimiro Miranda[1,2]
INESC TEC[1]
Faculty of Engineering of the University of Porto, FEUP[2]
Porto, Portugal
leonel.m.carvalho@inesctec.pt

Carolina G. Marcelino[3]
Elizabeth F. Wanner[4]
Intelligent System Laboratory[3], CEFET-MG
Department of Computer Science[4], CEFET-MG
Belo Horizonte, Brazil
carolina@lsi.cefetmg.br

***Abstract*—The optimal solution provided by metaheuristics can be viewed as a random variable, whose behavior depends on the value of the algorithm's strategic parameters and on the type of penalty function used to enforce the problem's soft constraints. This paper reports the use of parametric and non-parametric statistics to compare three different penalty functions implemented to solve the Security Constrained Optimal Power Flow (SCOPF) problem using the new enhanced metaheuristic Differential Evolutionary Particle Swarm Optimization (DEEPSO). To obtain the best performance for the three types of penalty functions, the strategic parameters of DEEPSO are optimized by using an iterative algorithm based on the two-way analysis of variance (ANOVA). The results show that the modeling of soft constraints significantly influences the best achievable performance of the optimization algorithm.**

Keywords — EPSO, DEEPSO; OPF; ANOVA; Non-parametric statistics

I. Introduction

Stochastic optimization methods, such as metaheuristics, are popular techniques among scientists and engineers facing difficult optimization problems. The majority of these optimization methods have a range of adjustable parameters (mutation rate, recombination probability, etc.) that need to be tuned. When proposing a new metaheuristic, it is common that its authors indicate an optimal range of variation for the parameters. As a matter of fact, a poor parameterization may even prevent the metaheuristic from finding optimal solutions. In addition, the optimal parameters, i.e. the parameters that result in the best algorithm performance, are problem specific, which indicates that systematic procedures for tuning metaheuristic parameters are of extreme value.

The output of metaheuristics is probabilistic, i.e. the solutions provided vary from run to run. In this way, a fair comparison between different metaheuristics or new features/operators of the metaheuristic, such as penalty schemes for the violations of constrains, can be only done after finding their optimal parameter configuration using a fine-tuning procedure. Successful tuning algorithms are the ones that try to find good parameters with the least possible effort, hence, outperforming naive enumeration methods which try every possible combination of parameters.

An insightful classification of tuning methods is present in [1]-[3]. Methods for parameter setting fall into one of two categories: parameter tuning and parameter control. The criteria used for its differentiation is the stage at which the tuning procedure takes place: the first category refers to optimal parameter configuration prior to the optimization process whereas the second category consists on launching the metaheuristic with a suboptimal parameters configuration and then optimizing the parameters during the search process. Parameter control is out of scope of this research work.

The Response Surface Methodology (RSM) [4] is regarded as a good statistical technique for parameter tuning. It consists on building an empirical model of the output as function of the parameters. This technique is applied after a factorial design that was planned to study the effects of two or more parameters in the output. The factorial design concept is extremely important since it avoids misleading conclusions when interactions between parameter are present.

Since the tuning procedure is itself an optimization problem, metaheuristics have been proposed to optimize the set of parameter used by other metaheuristic. Techniques for parameter tuning based on entropy concepts [5] have been used. Other statistical methods for the planning of experiments and tuning of stochastic optimization methods such as the Taguchi [6] methods, were also investigated. At last, racing techniques [7] have been studied as an alternative to statistical methods for parameter tuning. These techniques consist on setting several sub-populations with different parameters and making statistical tests to select the best performing population whose parameters are deemed as the best ones.

The research work reported refers to the tuning of an enhanced version of the metaheuristic EPSO (Evolutionary Particle Swarm Optimization) [8]. This version is denoted DEEPSO [9], for Differential EPSO, as some flavor from DE (Differential Evolution) was added to the canonical EPSO. This choice of metaheuristic is well justified: EPSO had successively been shown to outperform other meta-heuristics in a number of power system and other area problems [10]-[13], and a DEEPSO version was the winner in 2014 of the competition on the Application of Modern Heuristic Optimization Algorithms for Solving Optimal Power Flow

Problems, set up by the Working Group on Modern Heuristic Optimization of the IEEE PES [14].

The tuning process described below is divided in two phases. First, a methodology based on factorial designs and the two-way analysis of variance (ANOVA) is used to efficiently optimize DEEPSO parameters for different types of penalty functions commonly used in the Security Constrained Optimal Power Flow (SCOPF) [15][16]. Then, the types of penalty functions tested are compared using boxplot graphs and the Wilcoxon signed-rank test. Experiments conducted for the SCOPF problem are carried out on a configuration of the IEEE 57-bus, as in one of the tasks defined in the IEEE PES competition referred to above.

II. DEEPSO

The new DEEPSO variant can be seen as enhanced hybridization of the Evolutionary Particle Swarm Optimization [8][17]. EPSO is already a metaheuristic that can either be seen as Particle Swarm Optimization (PSO) with self-adaptive properties or Evolutionary Strategies (ES) with a self-adaptive recombination operator.

The improvement of DEEPSO over EPSO consists of using a recombination mechanism that relies on past information of the optimization process to create new promising solutions. This information becomes available by storing the best positions visited in a collective memory. The generation of new individuals from such memory is inspired on the concept of macro-gradient at the basis of Differential Evolution (DE).

EPSO already utilizes past information: e.g., each individual maintains a memory of the best position ever visited and uses it as attractor. DEEPSO proposes the replacement of the individual memory by a collective memory to enable an improved sensitivity of the optimization landscape. Therefore, the (individual) memory term of EPSO is replaced by a (collective) perception term. DEEPSO has shown to be superior to plain EPSO over a wide range of linear, non-linear, and mixed-integer optimization problems.

The DEEPSO algorithm is in essence an EPSO. The important distinction between the two variants lies in a different composition of the movement rule. In EPSO, the equations used to obtain a new individual, $\mathbf{X}_t$ of dimension N from its ancestor $\mathbf{X}_{t-1}$, its best ancestor $\underline{\mathbf{X}}_b$, the best ancestor ever found by the population, $\mathbf{X}_{gb}$, and its current velocity, $\mathbf{V}_t$, are

$$\mathbf{V}_t = w_i^* \times \mathbf{V}_{t-1} + w_m^* \times (\mathbf{X}_b - \mathbf{X}_{t-1}) + w_c^* \times \mathbf{C} \times \left(\mathbf{X}_{gb}^* - \mathbf{X}_{t-1}\right) \qquad (1)$$

$$\mathbf{X}_t = \mathbf{X}_{t-1} + \mathbf{V}_t \qquad (2)$$

in which t denotes the current generation, w represents weights (the subscripts i, m, and c of the weights stand for inertia, memory and cooperation, respectively), $\mathbf{C}$ is a $N \times N$ diagonal matrix of random variables that follow a Bernoulli distribution with success probability P (this probability must be set by the user) and the superscript * indicates that the corresponding parameter undergoes evolution under a mutation process. Typically, the mutation of the weight w follows a simple additive rule

$$w^* = w + \tau \times N(0,1) \qquad (3)$$

where τ is the mutation rate, which must be set by the user, and $N(0,1)$ is a number sampled from the standard Gaussian distribution. Note that the mutated weight must not become negative and limiting controls avoid it to grow highly above 1.

The DEEPSO movement rule stems from (1) – and deviates from classical PSO – by replacing the memory term by the perception term as

$$\mathbf{V}_t = w_i^* \times \mathbf{V}_{t-1} + w_m^* \times (\mathbf{X}_r - \mathbf{X}_{t-1}) + w_c^* \times \mathbf{C} \times \left(\mathbf{X}_{gb}^* - \mathbf{X}_{t-1}\right) \qquad (4)$$

in which $\mathbf{X}_r$ is an individual that is computed each iteration according to one of the following strategies:

- **Strategy Sg:** random sampling of one individual from all the individuals of the current population;
- **Strategy Pg:** random sampling of one individual from memory **B** that contains the past best individuals;
- **Strategy Sg-rnd:** uniform recombination of the individuals in the current population;
- **Strategy Pg-rnd:** uniform recombination of the individuals in memory **B**.

The accurate use of (4) must take into account the fitness difference between $\mathbf{X}_r$ and $\mathbf{X}_{t-1}$. If $\mathbf{X}_r$ is better than $\mathbf{X}_{t-1}$, then the individual must be attracted to $\mathbf{X}_r$; in this case, (4) is applied as it is formulated. Conversely, if $\mathbf{X}_r$ is worse than $\mathbf{X}_r$, then the relative position of these two vectors must be swapped so as to keep the individual away from the inferior position $\mathbf{X}_r$. In the case where uniform recombination is used, i.e. strategies 3 and 4, the sign of $(\mathbf{X}_r - \mathbf{X}_{t-1})$ is computed for every dimension $j = 1, 2, \ldots, N$ by comparing the fitness of the individual randomly selected to donate the j^{th} dimension of $\mathbf{X}_r$ with the fitness of $\mathbf{X}_{t-1}$.

The DEEPSO pseudo-code is as follows:

Pseudo-code 1: DEEPSO

```
Initialize a population of $k$ individuals, select the perception strategy, communication probability $P$ and mutation rate $\tau$
Evaluate all individuals, update the best individual $\mathbf{X}_{gb}$, and update memory B
$t = 1$
While the convergence criteria, e.g. a maximum allowable number of generations or a maximum number of generations with no improvements is not satisfied
    For all $k$ individuals the population
        Compute the perception term $\mathbf{X}_r$ for the current individual according to the perception strategy selected
        Copy the current individual
        Mutate the strategic parameters $w_i$, $w_m$, $w_c$, and $w_{gb}$ of the copied individual according to (3)
        Move the current individual and its copy according to (2) and (4)
        Evaluate the current individual and its copy
        Select the individual with the best fitness to be part of the population of next iteration
    End For
    Update the best individual $\mathbf{X}_{gb}$ and memory B
    $t := t + 1$
End While
```

III. Security-Constrained Optimal Power Flow

The goal of the Optimal Power Flow (OPF) problem is to define the optimal value of the control variables for an expected scenario of operation while complying with the physical limits of control and non-control variables and the power flow equations. The most common control variables are the active power generation and voltage magnitude at REF and PV buses, the tap setting of the On-Load Tap Changer (OLTC) transformers and the state of capacitor/reactor banks (ON/OFF). Physical limits of the variables include the apparent power flow or current in the network circuits, the active and reactive power limits of generators, the voltage magnitude of the buses, maximum/minimum value for the tap of the OLTC transformers, etc. Note that the control of the tap setting is either continuous or discrete whereas the control of capacitor/reactor banks is traditionally discrete. Normally, the goal of the OPF problem is to minimize the total production cost, which consists typically of a quadratic function, but it is also common to minimize active power losses.

The traditional OPF problem is a mixed-integer non-linear optimization problem. This formulation does not guarantee that the operation of the network remains in a secure state after a sudden equipment outage. The SCOPF model guarantees that the power can be successfully transferred from generators to loads not only under the expected network topology but also for any predicted contingency caused by the sudden loss of sections of the network, such as transmission lines and/or transformers. Therefore, the SCOPF problem has a higher number of constraints than the OPF problem.

The formulation of the SCOPF is as follows:

$$\text{min.} \qquad c = \sum_{i=1}^{NG} a_i + b_i \times Pg_i + c_i \times Pg_i^2 \tag{5}$$

s.t.

$$P_i = Pg_i - Pl_i = \sum_{j=1}^{NB} V_i V_j [G_{ij} \cos(\theta_i - \theta_j) + B_{ij} \sin(\theta_i - \theta_j)], \quad \forall i \in NB, \forall s \in NS \tag{5.a}$$

$$Q_i = Qg_i - Ql_i = \sum_{j=1}^{NB} V_i V_j [G_{ij} \sin(\theta_i - \theta_j) - B_{ij} \cos(\theta_i - \theta_j)], \quad \forall i \in NB, \forall s \in NS \tag{5.b}$$

$$\underline{V}_i \le V_i \le \overline{V}_i, \ \forall i \in NB, \forall s \in NS \tag{5.c}$$

$$|S_{ij}| \le \overline{S}_{ij}, \ \forall i \in NC, \forall s \in NS \tag{5.d}$$

$$|S_{ji}| \le \overline{S}_{ij}, \ \forall i \in NC, \forall s \in NS \tag{5.e}$$

$$\underline{Pg}_i \le Pg_i \le \overline{Pg}_i, \ \forall i \in NG, \forall s \in NS \tag{5.f}$$

$$\underline{Qg}_i \le Qg_i \le \overline{Qg}_i, \ \forall i \in NG, \forall s \in NS \tag{5.g}$$

$$\underline{t}_i \le t_i \le \overline{t}_i, \ \forall i \in NOLTC, \ t_i \in \mathbf{Z}, \forall s \in NS \tag{5.h}$$

$$0 \le q_i \le 1, \ \forall i \in NSHUNT, \ q_i \in \mathbf{Z}, \forall s \in NS \tag{5.i}$$

in which *Pg* (MW) is the active power generation, *Qg* (MVar) is the reactive power generation, *Pl* (MW) is the active power load, *Ql* (MVar) is the reactive power load, *V* (kV) is the voltage magnitude, θ (degrees) is the voltage angle, S_{ij} (MVA) is the apparent power flow injection at the sending end of the circuit connecting bus *i* to bus *j* whereas S_{ji} (MVA) is the apparent power flow injection at the receiving end of the same circuit, *t* is the tap setting position of the OLTC, *q* is a binary variable that represents the state of the capacitor/reactor banks, *a* (\$/h), *b* (\$/MWh) and *c* (\$/MWh^2) are cost coefficients, $\mathbf{Y} = \mathbf{G} + j\mathbf{B}$ is the bus admittance matrix, *NG* is the number of generators, *NB* is the number of buses, *NC* is the number of circuits in the network, *NOLTC* is the number of OLTC transformers, *NSHUNT* is the number of capacitor/reactor banks and *NS* is the number of scenarios that represent contingency states. Note that all constraints must be satisfied for all scenarios.

IV. Statistical Methodology for DEEPSO Parameter Optimization

Similarly to other metaheuristics, DEEPSO has a probabilistic convergence behavior, i.e. its response can be modeled as a random variable with an underlying probability distribution. For this reason, a rigorous analysis of its performance can only be made using statistical methods.

DEEPSO can be seen as a stochastic optimization process of the black box type. The study of this particular type of processes is normally performed using controlled experiments whose results are used to create empirical models that link inputs with the output. These experiments consist on making controlled changes to the inputs to observe and identify the cause for the variation observed in the output. The field of statistics that deal with this type of experiments is designated as factorial design. A factorial design consists of identifying two or more factors, each with discrete levels, to carry out experiments that take on all possible combinations of the discrete levels of the factors. Such experiments allow to study not only the effect of each factor individually on the response of the process but also possible interactions among the factors.

DEEPSO response is optimized based on a simple 2×2 factorial design. The two factors investigated are the mutation rate τ and the communication probability *P*. By using the ANOVA results, the levels of the factors are changed one at a time until there is no evidence that an updated set of parameters produce a better response. A graphical analysis based on boxplot graphs is also used to determine the best response, namely, by analyzing its dispersion. Note that there are other factors that might affect DEEPSO output, such as population size and perception strategy. If all these factors are taken into account, then the optimization of DEEPSO response will be a computationally heavy task with no guarantee of statistical significance.

The iterative optimization procedure starts by defining two values for each parameter: a low value (e.g. 0▹2) and a high value (e.g. 0▹8). Then, this gap is successively decreased, by using the bisection method, until the two-way ANOVA identifies that there is no evidence the parameters τ and *P* have an effect on DEEPSO response. The process is also stopped when the next level selected for the factor differs from the previous one by less than a threshold (e.g. 0.1). The two-way ANOVA *F*-test statistic is used to select the factor/parameter whose range will be further tightened: only one of the limits of factor with the highest *F*-test statistic is updated. After that, the main effect of the factor is analyzed: if it shows that DEEPSO output decreases with the increase of the factor, then the lowermost limit is updated according to the bisection method; otherwise, the uppermost limit is updated. A new 2×2 factorial design is then run with the updated limit until convergence is reached.

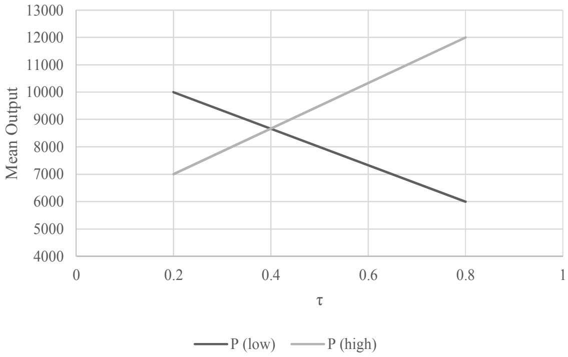

Fig. 1. Example of interaction between mutation rate, τ, and communication probability, P.

TABLE I. MEAN DEEPSO OUTPUT FOR A 2×2 FACTORIAL DESIGN

2×2 Factorial Design		
$\tau \backslash P$	0.2	0.8
0.2	10000	12000
0.8	6000	8000

TABLE II. MEAN DEEPSO OUTPUT FOR THE 2×2 FACTORIAL DESIGNS WITH UPDATED LIMITS

2×2 Factorial Design A		
$\tau \backslash P$	0.2	0.8
0.5	14000	5000
0.8	6000	8000

2×2 Factorial Design B		
$\tau \backslash P$	0.2	0.5
0.2	11000	12000
0.8	6000	2000

If the two-way ANOVA reports that the interaction between τ and P (see Fig. 1) has the highest F-test statistic, then the limits of the variables are updated for one variable at a time and two new 2×2 factorial designs are carried out. The extra factorial design is necessary to guarantee that no loss of information occurs. Boxplot graphs can be used to select the best design in addition to the typical two-way ANOVA results: the design that leads to the best mean response proceeds to the next iteration.

As a didactic example, Table I contains the mean response of DEEPSO for the SCOPF problem according to a 2×2 factorial design for two values of τ and P. The data in this table shows that there is a clear interaction between the two factors: the main effect of τ is -4000 whereas the main effect of P is 2000. Hence, we can conclude that DEEPSO output decreases with the increase of τ and with the decrease of P. Two new experiments are carried out with the updated ranges for τ and P: one 2×2 factorial design with $\tau = \{0.5, 0.8\}$ and $P = \{0.2, 0.8\}$ and another 2×2 factorial design with $\tau = \{0.2, 0.8\}$ and $P = \{0.5, 0.8\}$. Possible results for these experiments are presented in Table II. Note that 2×2 factorial design B has led to the lowest mean response ($\tau = 0.8$ and $P = 0.5$) which indicates that the set of values used in design B should be selected for further partitioning. Further analysis based on boxplot charts can also be made to analyze the dispersion of the DEEPSO output for each experiment.

As previously mentioned, DEEPSO is applied in this paper to solve the SCOPF problem. The parameter optimization procedure is carried out for all different types of penalty functions: in this case, only three penalty functions will be tested. Then, the experiments that led to the lowest mean response for each type of penalty function are selected for comparison.

The final comparative analysis is based on the Wilcoxon signed-rank test which is a non-parametric statistical hypothesis test that is used to compare two sets of samples and assess whether their population mean are different. The use of non-parametric tests is required since the distribution of DEEPSO output for the SCOPF problem does not follow a Gaussian distribution (see Fig. 2).

After determining if the population means are different, then the one that has the lowest mean output is selected as the best one.

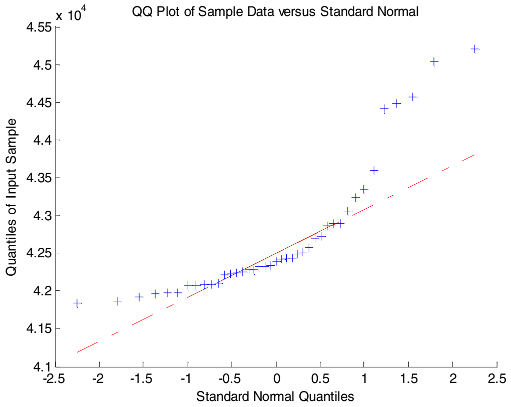

Fig. 2. Illustration of the Q-Q plot of the cost of the solutions provided by DEEPSO for the SCOPF problem after 25,000 fitness evaluations.

V. CASE STUDY

The proposed optimization methodology was used to compare the performance of DEEPSO for three different penalty functions that can be used to solve the SCOPF problem for the IEEE 57-bus system. In this research work, the active power production at PV buses, the voltage magnitude at the REF and PV buses, the OLTC tap setting position and the state of the capacitor banks were modeled as hard constraints, i.e. the limits of these variables were not allowed to be violated.

Since DEEPSO is a metaheuristic devised for continuous search spaces, a simple rounding procedure was used to convert real variables into integer ones. Soft constraints, i.e. the constraints that were modeled as penalties to (5), consist of the active power production of the REF bus, the voltage magnitude at PQ buses, the apparent power flow through branches, and the reactive power generation at REF and PV buses. The deviations from the limits were computed in per-unit system.

The three penalty functions tested were labeled as f_A, f_B and f_C and are defined in (6), (7) and (8) respectively.

$$f_A = [1 \quad 1 \quad 1 \quad 1] \begin{bmatrix} \max\left(0, Pg_{REF} - \overline{P}g_{REF}\right) + \max\left(0, \underline{Pg}_{REF} - Pg_{REF}\right) \\ \sum_{i=1}^{NB_{PQ}} \left[\max\left(0, V_i - \overline{V}_i\right) + \max\left(0, \underline{V}_i - V_i\right)\right] \\ \sum_{i=1}^{NC} \left[\max\left(0, |Sij| - \overline{S}_{ij}\right) + \max\left(0, |Sji| - \overline{S}_{ij}\right)\right] \\ \sum_{i=1}^{NB_{REF+PV}} \left[\max\left(0, Q_i - \overline{Q}_i\right) + \max\left(0, \underline{Q}_i - Q_i\right)\right] \end{bmatrix} \tag{6}$$

$$f_B = [1 \quad 1 \quad 1 \quad 1] \begin{bmatrix} \max\left(0, Pg_{REF} - \overline{P}g_{REF}\right)^2 + \max\left(0, \underline{Pg}_{REF} - Pg_{REF}\right)^2 \\ \sum_{i=1}^{NB_{PQ}} \left[\max\left(0, V_i - \overline{V}_i\right) + \max\left(0, \underline{V}_i - V_i\right)\right]^2 \\ \sum_{i=1}^{NC} \left[\max\left(0, |Sij| - \overline{S}_{ij}\right) + \max\left(0, |Sji| - \overline{S}_{ij}\right)\right]^2 \\ \sum_{i=1}^{NB_{REF+PV}} \left[\max\left(0, Q_i - \overline{Q}_i\right) + \max\left(0, \underline{Q}_i - Q_i\right)\right]^2 \end{bmatrix} \tag{7}$$

$$f_C = [1 \quad 1 \quad 1 \quad 1] \begin{bmatrix} \max\left(0, Pg_{REF} - \overline{P}g_{REF}\right) + \max\left(0, \underline{Pg}_{REF} - Pg_{REF}\right) \\ \max\limits_{\forall i \in NB_{PQ}} \left[\max\left(0, V_i - \overline{V}_i\right) + \max\left(0, \underline{V}_i - V_i\right)\right] \\ \max\limits_{\forall i \in NC} \left[\max\left(0, |Sij| - \overline{S}_{ij}\right) + \max\left(0, |Sji| - \overline{S}_{ij}\right)\right] \\ \max\limits_{\forall i \in NB_{REF+PV}} \left[\max\left(0, Q_i - \overline{Q}_i\right) + \max\left(0, \underline{Q}_i - Q_i\right)\right] \end{bmatrix} \tag{8}$$

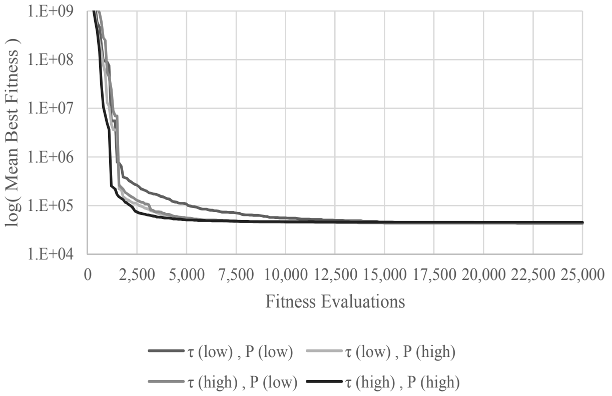

Fig. 3. Log plot of the DEEPSO mean best fitness (penalty function f_A) for different values of τ and P.

The metric used to compare the performance of the individuals is the final cost represented by (5). Only samples that are feasible solutions, i.e. solutions with no violations, were considered. The fitness of the individuals computed by DEEPSO depends on the penalty function selected and is defined as

$$fit = c + 1 \times 10^7 \times f \tag{9}$$

in which f represents one of the three penalty functions.

Each 2×2 factorial design consists of four experiments, each comprising 41 randomized runs of DEEPSO for each pair $\{\tau, P\}$. Each run was stopped after 25,000 fitness evaluations (the exploitation phase is reached after 15,000 fitness evaluations - see Fig. 3). The number of individuals in the initial population is 50. The perception strategy Pb-rand was adopted.

A. Parameter Optimization

This section contains a simple illustration of the iterative process for the optimization of DEEPSO parameters τ and P. The penalty function considers is PF_A. Initially, the following four pairs of $\{\tau, P\}$ were defined: A = {0.2, 0.2}, B = {0.2, 0.8}, C = {0.8, 0.2} and D = {0.8, 0.8}. A maximum variation of 0.1 was defined for the absolute difference between the levels of τ and P in two consecutive iterations: if the difference between the two consecutive values is less than 0.1, then the optimization process is stopped for that factor/parameter. Table III shows DEEPSO parameters during the tuning process for f_A. This optimization process was stopped at the 4th iteration, when the p-value of all the factors computed by ANOVA was greater than 0.05 meaning that there is no evidence that the factors have an impact on the DEEPSO response. Note that only 10 runs of DEEPSO were required.

TABLE III. ANOVA RESULTS - PF_A

Factor	F-test Statistic	P-value	Main Effect
	1st Iteration		
τ	3.474	0.064	-350.73
P	6.675	**0.011**	486.14
$\tau \times P$	0.225	0.636	89.17
	2nd Iteration		
τ	5.469	0.021	-548.35
P	6.022	**0.015**	575.42
$\tau \times P$	1.496	0.223	286.79
	3nd Iteration		
τ	5.508	**0.020**	-448.7
P	1.448	0.231	230.01
$\tau \times P$	0.958	0.329	187.14
	4th Iteration		
τ	2.849	0.093	-306.78
P	2.497	0.116	287.21
$\tau \times P$	1.807	0.181	244.34

TABLE IV. SETTING VALUES FOR τ AND P - f_A

Factor	Iteration			
	1st	**2nd**	**3rd**	**4th**
τ (low)	0.2	0.2	0.2	**0.5**
τ (high)	0.8	0.8	0.8	0.8
P (low)	0.2	0.2	0.2	0.2
P (high)	0.8	**0.5**	**0.35**	0.35

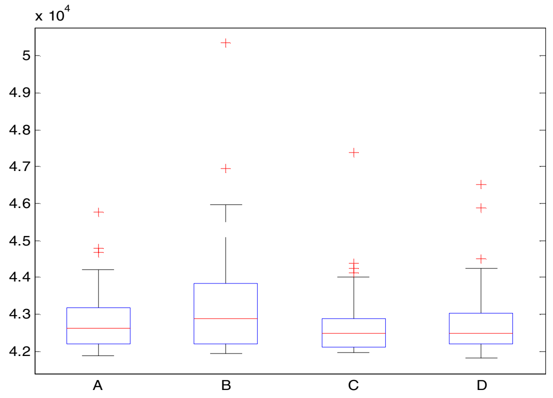

Fig. 4. Boxplot of the DEEPSO response for the final pairs A = {0.5, 0.2}, B = {0.5, 0.35}, C = {0.8, 0.2} and D = {0.8, 0.35} (penalty function f_A).

Figure 4 highlights the boxplot of the DEEPSO output for the four final parameters pairs. Note there is no visible difference between the means and the dispersion of the

DEEPSO output showing that all parameter pairs enable DEEPSO to produce statistically similar outputs.

TABLE V - SETTING VALUES FOR τ AND P - f_B (LEFT) AND f_C (RIGHT)

Factor	Iteration
	1st
τ (low)	0.2
τ (high)	0.8
P (low)	0.2
P (high)	0.8

Factor	Iteration	
	1st	2nd
τ (low)	0.2	0.5
τ (high)	0.8	0.8
P (low)	0.2	0.2
P (high)	0.8	0.5

Tables V show the final parameters pairs obtained for the remaining penalty functions after the tuning process. Note that the final pairs are always different for the three functions.

B. Statistical Comparison of Penalty Functions

This section presents the results for the final comparison between the proposed penalty functions. The comparisons is made for the combination of parameters that resulted in the lowest mean output. Other criteria based on the minimum, maximum, median, standard deviation of DEEPSO output, could also be used.

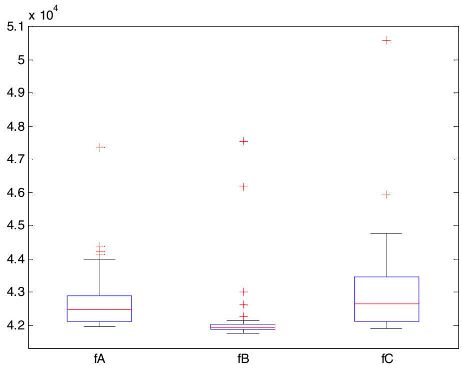

Fig. 5. Boxplot of the DEEPSO response for the final pairs f_A = {0.8, 0.2}, f_B = {0.8, 0.8} and f_C = {0.5, 0.2}.

Figure 5 shows the boxplot of DEEPSO results obtained with the best set of parameters for each penalty function. As one can see, f_B leads to the lowest mean output as well as the lowest dispersion of results.

TABLE V. RESULTS OF THE WILCOXON SIGNED-RANK TEST

P-value	f_A	f_B	f_C
f_A	1.000	0.000	0.359
f_B	0.000	1.000	0.000
f_C	0.359	0.000	1.000

To detect whether the performance of DEEPSO is different for the proposed fitness functions, the Wilcoxon signed-rank test was used. This non-parametric test investigates the null hypothesis that two different samples are from continuous distributions with equal means against the alternative that they are not, normally for a given significance level, which is this case, was assumed to be 0.05. Hence, a p-value greater than or equal to 0.05 indicates that the null hypothesis should be rejected at the default 5% significance level. Bearing this in mind, we can see that DEEPSO output when using f_B is significantly different from the output when using f_A and f_C. Moreover, there is no evidence that the mean output provided by DEEPSO when using f_A and f_C are different, which means that f_B with optimized parameters {0.8, 0.8} would yield the best DEEPSO results for the SCOPF problem.

VI. CONCLUSIONS

This paper reports the use of parametric and non-parametric statistics to compare three different penalty functions that can be used to solve the SCOPF problem by new hybrid metaheuristic Differential Evolutionary Particle Swarm Optimization (DEEPSO). The strategic parameters of DEEPSO were optimized to obtain the best performance for the three types of penalty functions using an iterative algorithm based on the factorial designs and two-way analysis of variance (ANOVA).

Results prove that the best DEEPSO performance in the SCOPF problem is obtained for only one type of penalty function. The statistical analysis presented in this paper can also be extended and used to make robust comparisons between individual aspects of the metaheuristic as well as the performance of different metaheuristics applied to other optimization problems.

REFERENCES

[1] A. E. Eiben and S. K. Smit, “Evolutionary Algorithm Parameters and Methods to Tune Them,” in *Autonomous Search,* Springer Berlin Heidelberg, 2012, pp. 15–36.

[2] A. E. Eiben and S. K. Smit, “Parameter tuning for configuring and analyzing evolutionary algorithms,” *Swarm Evol. Comput.*, vol. 1, no. 1, pp. 19–31, Mar. 2011.

[3] A. E. Eiben, R. Hinterding, and Z. Michalewicz, “Parameter control in evolutionary algorithms,” *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 2. pp. 124–141, 1999.

[4] D. C. Montgomery, *Design and Analysis of Experiments*, John Wiley & Sons, 2000.

[5] V. Nannen, S. Smit, and A. Eiben, “Costs and Benefits of Tuning Parameters of Evolutionary Algorithms,” in *Parallel Problem Solving from Nature – PPSN X*, vol. 5199, Springer Berlin Heidelberg, 2008, pp. 528–538.

[6] T. Fearn, “Taguchi methods,” *NIR News*, vol. 12. 2001.

[7] P. Balaprakash, M. Birattari, and T. Stützle, “Improvement Strategies for the F-Race Algorithm: Sampling Design and Iterative Refinement,” in *Hybrid Metaheuristics SE*, vol. 4771, Springer Berlin Heidelberg, 2007, pp. 108–122.

[8] V. Miranda and N. Fonseca, “EPSO – best-of-two-worlds meta-heuristic applied to power system problems”, in Proc. of the *2002 Congress on Evolutionary Computation,* Honolulu (HI), USA, May 2002.

[9] V. Miranda and R. Alves, “Differential Evolutionary Particle Swarm Optimization (DEEPSO): A Successful Hybrid,” in Proc. of the *2013 BRICS Congress on Computational Intelligence and 11th CBIC Brazilian Congress on Computational Intelligence,* Ipojuca (Pernanbuco), Brazil, Sep. 2013.

[10] M. Eghbal, E.E.El-Araby, N. Yorino and Y. Zoka, "Application of Metaheuristic Methods to Reactive Power Planning: A Comparative Study for GA, PSO and EPSO", in Proc. of the *IEEE International Conference on Systems, Man and Cybernetics*, Montreal (Quebec), Canada, Oct. 2007.

[11] H. Mori and Y. Komatsu, “A Hybrid Method of Optimal Data Mining and Artificial Neural Network for Voltage Stability Assessment”, in Proc. of the *IEEE Power Tech*, St. Petersburg, Russia, Jun. 2005

[12] D. Midence and A. Vargas, “Comparative Study of Evolutionary Computation Algorithms in the Optimization of the Reliability of Power Distribution Networks” (in Spanish), in Proc. of the *XIII CIGRE Regional Iberoamerican Meeting*, Puerto Ibiza, Argentina, May 2009.

[13] D. Souza, O. Teixeira, D. Monteiro, R. L. de Oliveira, “A CUDA-Based Cooperative Evolutionary Multi-Swarm Optimization Appliedto Engineering Problems”, in Proc. of the *XXXII Congress of the Brazilian Computing Society*, Curitiba (Paraná), Brazil, Jul. 2012.

[14] IEEE PES Working Group on Modern Heuristic Optimizatiom [Online]. Available: https://www.uni-due.de/ieee-wgmho/competition2014

[15] P. E. Onate Yumbla, J. M. Ramirez, and C. A. Coello Coello, “Optimal Power Flow Subject to Security Constraints Solved With a Particle Swarm Optimizer,” *IEEE Trans. on Power Syst.,* vol. 23, no. 1. pp. 33–40, 2008.

[16] L. G. Dias and M. E. El-Hawary, “Effects of load modeling in security constrained OPF studies,” *IEEE Trans. on Power Syst.,* , vol. 6, no. 1. pp. 87–93, 1991.

[17] V. Miranda, H. Keko, and A. Duque, “Stochastic star communication topology in evolutionary particle swarms (EPSO),” *Int. J. Comput. Intell.* vol. 4, no. 2, pp. 105–116, 2008