Fundamentals of the C-DEEPSO Algorithm and its Application to the Reactive Power Optimization of Wind Farms

Carolina G. Marcelino
Paulo E. M. Almeida, Elizabeth F. Wanner
Intelligent Systems Laboratory. CEFET-MG - Brasil
Email: {carolina, pema}@lsi.cefetmg.br
efwanner@decom.cefetmg.br

Leonel M. Carvalho, *IEEE Member*
Vladimiro Miranda, *IEEE Fellow*
Centre for Power and Energy Systems - CPES
INESC TEC - Portugal
Email: {lcarvalho, vmiranda}@inesctec.pt

***Abstract*—In this paper, a novel hybrid single-objective metaheuristic, the so called C-DEEPSO (Canonical Differential Evolutionary Particle Swarm Optimization), is proposed and tested. C-DEEPSO can be viewed as an evolutionary algorithm with recombination rules borrowed from PSO, or a swarm optimization method with selection and self-adaptiveness properties proper from DE. A study case of the problem of optimal control for reactive sources in energy production by Wind Power Plants (WPP), solved by means of Optimal Power Flow (OPF-like), is used to test the new hybrid algorithm and to evaluate its performance. C-DEEPSO is compared to the baseline algorithm, DEEPSO, and to a reference algorithm, Mean-Variance Mapping Optimization (MVMO). The experiments indicate that the proposed algorithm is efficient and competitive, capable to tackle this large-scale problem. The results also show that the new approach exhibits better results, when compared to MVMO.**

I. Introduction

A wide variety of algorithms and metaheuristics have been successfully applied to many optimization problems, such as: Genetic Algorithms (GA) [1], Ant Colony Optimization (ACO) [2], Particle Swarm Optimization (PSO) [3], Simulated Annealing [4], Differential Evolution (DE) [5], [6], Evolutionary Multiobjective Algorithms [7], just to name a few. However, those algorithms and metaheuristics suffer from the curse of dimensionality, implying that the performance of these methods tends to deteriorate rapidly as the dimensionality of the problem is increased [8].

Large-scale optimization problems can be found in countless practical applications such as industrial control, biomedicine, aerospace, logistics, etc. They are affected by the curse of dimensionality in many ways: the larger the dimension of the problem, the larger the search space; the larger the dimension of the problem, the greater the risk of some characteristics of the problem to be altered with the scale.

Typically, these problems are hard to solve due to the inherent difficulty of finding optima in high-dimensional spaces. Hence, new optimization methods, which are mostly metaheuristic-based, are being proposed to overcome the dreadful curse of dimensionality [9]. In recent years, hybridization emerged as an important alternative in optimization and operational research, as it has become clear that a smart combination of two or more techniques resulting on a new method, so called a hybrid metaheuristic, can overcome specific limitations of the underlying algorithms and offer more efficient behavior, less sensitivity to dimensionality and better performances.

Namely, Zhang and Xie, in 2003 [10], and Hao and others, in 2007 [11], proposed two different DEPSO hybrids, mixing together features of DE and PSO algorithms. Later, on 2013, Miranda and Alves [12] incorporated DE and EPSO algorithms together in a new hybrid, called DEEPSO, and showed that it could be efficient to solve some energy-related optimization problems. This work advances more in this direction, proposing a new hybrid algorithm, C-DEEPSO (Canonical Differential Evolutionary Particle Swarm Optimization), as an extension of DEEPSO. It was created with the aim of improving the DE inspiration, as proposed by DE creators back in 1995 [6]. C-DEEPSO can be viewed as an evolutionary algorithm with recombination rules borrowed from PSO, or a swarm optimization method with selection and self-adaptive properties, proper from DE.

In the energy production and management environment, the active power dispatch is a common problem, frequently solved by using an Optimal Power Flow (OPF) model [13]. The study of OPF plays an important role in the energy field. Usually, OPF are considered as large scale optimization problems, due to their large dimension, non-linearity, non-convexity and multimodality features [14]. OPF consists of an assessment of the best settings for control variables: active power and voltages of generators, discrete variables like transformers taps, continuous variables like shunt reactors and capacitors values and other variables, so as to attain a common objective such as, for example, minimization the operating cost [15]. Greater reliance electricity refers to a situation where the consumer does not depend only on the availability of electricity, but also on a reliable and safe supply, which guarantees high quality and uninterrupted power.

In this context, this paper addresses an OPF problem in wind power generation, a technology which takes advantage of the kinetic energy of the wind to produce electricity. Wind power generation has been used for centuries, although, in the past, its use was restricted to mechanical applications such as the windmill. In the last century, wind energy started being used to produce electricity.

In recent decades, this usage, especially in some European countries, has undergone a major development coming to what we know as major power applications, or wind farms. Recently, some countries are making great efforts to develop onshore and offshore wind farms [16]. Wind energy production can be represented in a simplified manner as in Figure 1.

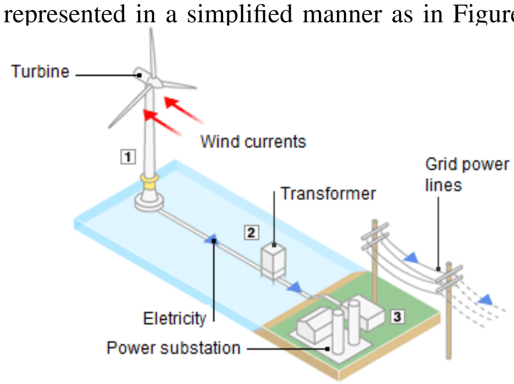

Fig. 1: Wind power generation representation.

Basically, wind energy production happens as three main steps, accordingly:

1) Blades turn shaft inside wind unit - a box on top of the turbine. An enclosed generator uses magnetic fields to convert rotational energy into electric energy;
2) Transformer (in this case, offshore) converts the energy for distribution and sends it to power substation;
3) Grid distributes power to consumers.

To explore ways for optimizing the power flow in Wind Power Plants (WPP) is highly justifiable, since environmental concerns and energy shortages have led countries to invest in several renewable energy sources [17]. Nowadays in Brazil, there is a desperate need for cheap and clean energy. The country undergoes an energy crisis, since its main source coming from hydroelectric plants has been reduced because of severe droughts over the last years [18].

According to the Energy Research Enterprise (EPE), the percentage share of all renewable energy sources will increase in Brazil over the next years. The presence of those resources, which totaled 44.8% in 2010, will reach 46.3% in 2020, according to the most recent cycle of Ten Year Plan for Expansion of Energy (PDE). Wind energy is the fastest growing source of power generation in Brazil. Over the next few years, wind energy will contribute to the generation of thousands jobs, billions in investments and million homes supplied. In 2015, 113 wind farms are under construction, with a total capacity of 2.7 GW. To join OPF research with wind power generation may lead to a more efficient energy planning for Brazil, helping to achieve its clean production goals in the near future.

Several approaches and methods for OPF problem solution can be found in literature. According to [13], [19] and [20], it is possible to describe, in a simple manner, the main advantages and disadvantages of the principal methods used for solving this problem:

1) Linear Programming: usage of linear or piecewise linear cost functions and usage of DC power flow instead of AC power flow, which provides a linear relation between injections and line flows [21], [22];
2) Quadratic Programming: usage of a quadratic objective function, all constraints are linear [23], [24];
3) Metaheuristics: GA, PSO, DE, ACO, Evolutionary Computation [25], [26], [27], [28], [29].

Social and environmental concerns related to global warning, emission of greenhouse gases, sustainable development, and the rise of fuel costs have motivated many countries to invest in renewable energy sources, such as wind generation [30]. Several research approaches addressing OPF for wind power can be found in literature.

Jabr and Pal proposed in [16] a stochastic model of wind generation with OPF, aiming to minimize costs of managing wind intermittency based on probability/relative frequency histograms of forecasting error. Results showed that the proposed model could quantify the effects of shape/skewness of the forecast error distribution. In the work of Shi et. al. [31], the minimization of wind power generation costs, using an algorithm known as Self-Adaptive Evolutionary Programming (SAEP) is proposed [32]. Simulation results demonstrated the effectiveness and validity of the proposed model and method.

Joseph et. al. [33] proposed to maximize the system loadability within stability margins with the use of a PSO method. Simulation results showed that PSO improved the load carrying capacity, when compared to the usual control of the plant. Artificial Bee Colony (ABC) was applied in [34] to minimize the total costs of production on a wind OPF model. When compared to PSO, GA, among other methods, ABC showed better average results. An approach for the problem of optimal control of reactive power in WPP was proposed by [35]. In that work, the Mean-Variance Mapping Optimization (MVMO) algorithm was proposed for online optimal controller of WPP. Results indicated that MVMO in some test scenarios was effective for solving the problem.

Recently, new approaches for OPF problem have been made through the usage of hybrid optimization methods. According to Frank et. al. [19], the most promising developments in the OPF field have been in hybrid methods, recently. In many cases, hybrid methods have been shown to be more robust and to converge more quickly to optimal solutions than their individual component methods operating alone. The main objectives of this paper are to propose C-DEEPSO as a new hybrid method, and then, to evaluate its performance for solving an OPF problem. In this case, the OPF is an adapted version of the reactive power optimization of wind farms. And, which, in this work, is denoted by OPF-like problem.

The paper is organized as follows. Section 2 presents and describes mechanisms of C-DEEPSO algorithm. Section 3 presents the OPF-like problem for the reactive power optimization in wind farms. Section 4 presents the experimental setup. Finally, Section 5 includes an overall discussion on the results obtained and presents a brief conclusion to the research work reported in this paper.

II. The C-DEEPSO Algorithm

C-DEEPSO, which stands for Canonical Differential Evolutionary Particle Swarm Optimization, is a hybrid single-objective metaheuristic that incorporates distinct features of Evolutionary Computation, Particle Swarm Optimization (PSO), and Differential Evolution (DE). This algorithm, which is an enhancement over EPSO (Evolutionary Particle Swarm Optimization) [36] and Differential Evolutionary Particle Swarm Optimization (DEEPSO) [12], can be viewed as an evolutionary algorithm with recombination rules borrowed from PSO, or a swarm optimization method with selection and self-adaptiveness properties proper from DE.

Like every population-based metaheuristic, C-DEEPSO relies on the repeated application of mutation, recombination, and selection operators over a population of solutions (individuals), to create new solutions such that the overall fitness of the population is gradually improved until a desired convergence criterion is met. Generation of new solutions in C-DEEPSO is based on successive recombination operations applied on current and past solutions, in the same way as in DEEPSO. Recombination is governed by the so called *Movement Rule*, which in DEEPSO [12] is given by Eq. (1) and Eq. (2):

$$V_t = w_I^* \times V_{t-1} + w_M^* \times (X_r - X_{t-1}) + w_C^* \times C \times (X_{gb}^* - X_{t-1}), \tag{1}$$

$$X_t = X_{t-1} + V_t, \tag{2}$$

in which t represents the current DEEPSO generation, X_r is an individual different from X_{t-1} and can be obtained according to one of the following four options [12]:

1) sampled from all individuals in current generation: S_g;
2) sampled from a Memory B of the best individual found so far: P_b;
3) sampled as an uniform recombination from the individuals of the current generation: S_g-rnd;
4) sampled as an uniform recombination within Memory B: P_b-rnd.

Analyzing Eq. (1), it is possible to see that this algorithm is best described as an optimization method for particle swarm with selection and self-adaptation. This characteristic is supported by the fact that in original DEEPSO there is no inspiration strongly linked to the classical DE algorithm, regarding its search space conducted by the mutation operator, which uses three vectors (see [5], [6]). For the sake of clarity to the reader, the DE/rand/1 mutation operator is shown:

$$V_{t,i} = x_{t,r1} + F(x_{t,r2} - x_{t,r3}); r_1, r_2, r_3 \in \{1, \dots, N\}, \tag{3}$$

in which parameters $x_{t,r1}$, $x_{t,r2}$ and $x_{t,r3}$ are different vectors obtained in the population and F is a number, which generally belongs to the interval $[0, 2]$ aiming to control the amplification of differential variation. Comparing the DE mutation operator given by Eq. (3) and the DEEPSO movement rule given by Eq. (1), it is easy to see that three vectors are used in the mutation process, while in DEEPSO movement rule, only two vectors, represented by X_r and X_{t-1}, are used in the process. On the other hand, C-DEEPSO uses the original DE mutation operator, as described by Eq. (3).

Regarding DEEPSO, the distinguishing feature of C-DEEPSO consists on using an improved assimilation of the optimization landscape. Similarly to some general evolutionary algorithms, this assimilation can be roughly obtained by comparing different solutions, i.e., by computing macro-gradients. To take advantage of the information collected by the population throughout the search, C-DEEPSO relies on a collective memory instead of multiple and independent memories that encompass the search experience of each individual. For the best landscape of assimilation, C-DEEPSO proposes a new movement equation inspired on Eq. (1). In C-DEEPSO, the movement equation is described by Eq. (4) and Eq. (5) as:

$$V_t = w_I^* \times V_{t-1} + w_A^* \times (X_{best} + F \times (X_r - X_{t-1})) + w_C^* \times C \times (X_{gb}^* - X_{t-1}), \tag{4}$$

in which the *DE/best/1* strategy by DE algorithm is applied when X_r is better than X_{t-1}.

$$V_t = w_I^* \times V_{t-1} + w_A^* \times (X_{best} + F \times (X_{t-1} - X_r)) + w_C^* \times C \times (X_{gb}^* - X_{t-1}), \tag{5}$$

in which the *DE/best/1* strategy of DE algorithm is applied when X_{t-1} is better than X_r.

In Eq. (4) and (5), t denotes the current generation, X the current position or solution, X_{best} the best solution ever found by the individual, X_{gb} the best solution ever found by the population, V_t is the velocity of the individual, and C represents a $n \times n$ diagonal matrix of random variables that is sampled at every iteration and follows a Bernoulli distribution. The variables w_I, w_A and w_C are the weights relating to inertia, assimilation and communication, respectively. The superscript $*$ indicates that the corresponding parameter/quantity undergoes evolution under a mutation process.

C-DEEPSO also has a memory mechanism, called Memory B, which must enclose not only the position of the individual but also its fitness. Aiming to ensure a great assimilation of the search space, a new way to generate X_r is proposed. The new strategy, named S_gP_b-rnd, is a combination of S_g-rnd and P_b-rnd strategies. In this case, when using S_gP_b-rnd, an uniform recombination from different solutions is used to obtain X_r, and the reversion of the position of X_r and X_{t-1} in Equations (4) and (5) is done for every dimension of the search space.

Hence, after randomly selecting a position from the memory, that provides the dimension i of X_r, the fitness of the selected position is compared to the fitness of X_{t-1} to decide whether the individual will be attracted or repelled to that particular dimension of the search space. This procedure is repeated for all dimensions of X_r.

Typically, the mutation of a generic weight w of an individual follows a simple additive rule as described by (6),

$$w^* = w + \tau \times N(0,1), \tag{6}$$

in which τ is the mutation rate that must be set by the user. $N(0,1)$ is a number sampled from the standard Gaussian Distribution.

Observe that the mutated weight must not become negative or greater than 1. Moreover, not only the weights presented in Eq. (1) are mutated but also is X_{gb}. This attracting position is slightly moved in the search space using a Gaussian Distribution to prevent the population to be trapped in a given area, which is especially evident in those cases when the cooperation term becomes more dominating than the other terms. Mutation of X_{gb}, which is done for every particle, is performed according to the following equation:

$$X^*_{gb} = X_{gb}[1 + \tau \times N(0,1)]. \tag{7}$$

C-DEEPSO can, therefore, be viewed as a hybrid evolutionary algorithm, based on the DE mutation operator, that borrows the recombination rules from PSO algorithm. Algorithm 1 shows the pseudo-code for C-DEEPSO, in which $MaxIT$ is maximum number of iterations, NP is Population size, MB is Memory B size, P is communication probability rate and τ is mutation rate.

Algorithm 1: Pseudo-code of C-DEEPSO

```
begin
    INITIALIZE MaxIT, NP, MB, P and τ;
    EVALUATE NP
    UPDATE X_gb and MB;
    while stopping criterion is not satisfied do
        for all individuals in the population do
            COMPUTE X_r using S_g P_b-rnd;
            COPY X_t;
            MUTATE weights using Eq. (6) ;
            MOVE X_t and its copy using Eq. (4) or Eq. (5);
            EVALUATE X_t and its copy;
            SELECT X_best to be part of the new NP
                (by stochastic tournament e.g.);
        UPDATE X_gb and MB;
```

III. Real-World Application

This section presents an application of C-DEEPSO algorithm in a well-studied real world large-scale problem of OPF. Generally, an OPF problem can be expressed as the minimization of the cost of production of a power plant or system. However, many other OPF objectives are also possible, such as the minimization of changes in controls (in case of N-1 contingencies for example), minimization of system losses, minimization of pollutant emission or minimization of power not supplied. Besides, a multiobjective function can also be used by integration and combination of two or more simple objective functions [37].

Regardless of the objective function, an OPF must also verify the entire set of constraints that stem from the power flow equations. The equality constraints are associated with power balance at each node and power flow equations. In the inequality constraints, the operational limits are included, as well as the limits of the control variables, line flows and voltages (magnitude and angle) and security constraints.

C-DEEPSO algorithm is applied to to an optimal reactive power dispatch (ORPD) inside a WPP. Subsections (III-A) and (III-B) include the characterization of the Bus System 41 and the mathematical model for this problem.

A. IEEE 41 Bus System - Offshore Wind Power Plant (WPP)

A version of IEEE 41 bus system [38] is considered to investigate the effectiveness of the proposed methodology in ORPD problems. Figure 2 shows the layout of a WPP system. The presented scheme corresponds to a typical topology of a WPP, that is connected to the main grid through an AC cables.

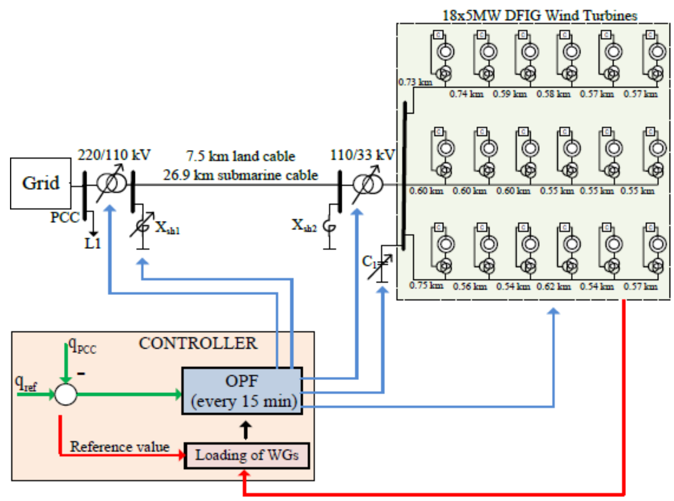

Fig. 2: Layout of an offshore WPP control scheme [38].

As can be seen in the layout details on Figure 2, two factors, X_{sh1} and X_{sh2}, that can be continuously adjusted, and the capacitor C_1, provide support to auxiliary reactive power. L1 represents a load which indicates the active power generated by the WPP. The system control mode considers that the WPP will provide the necessary reactive power (q_{ref}) until it reaches the power of common coupling (q_{PCC}). The schematic shown in Table I characterizes the IEEE 41 bus system.

TABLE I: IEEE 41 bus test system features

	IEEE 41-bus System		
Composition of test system	Genetarions		18
	Submarine cable		1
	Transfomers	T1 (fixed tap)	33
		T2 (fixed tap)	13
Composition ofthe optimization problem	Optimization variables	Continuous variables	18
		Discrete variables	2
		Continuous and Stepwise variables	2
Constrains			123

B. Problem formulation

In this problem, the objective function is given by the minimization of total losses of the wind energy system:

$$min\ P_{loss} = \sum_{K=1}^{N_L} G_K[U_i^2 + U_j^2 - 2|U_i||U_j|\cos(\delta_i - \delta_j)], \quad (8)$$

in which, N_L is the total number of lines in the system; G_K is the conductance of the line K, U_i and U_j are the magnitudes of the sending end and receiving end voltages of the line; δ_i and δ_j are angles of the bus voltages.

This ORPD problem must also satisfy some constraints. Constraints can be either hard constraints, which set conditions for the variables that are required to be satisfied, or soft constraints, which have some variable values that are penalized in the objective function if the conditions on the variables are not satisfied. The following defines the constraints of ORPD:

- Active and Reactive power balance constraints:

$$P_i = P_i^{gen} - P_i^{load} = \quad (9)$$
$$\sum_{j=1}^{n} U_i U_j [G_{ij}\cos(\theta_i - \theta_j) + B_{ij}\sin(\theta_i - \theta_j),$$
$$Q_i = Q_i^{gen} - Q_i^{load} =$$
$$\sum_{j=1}^{n} U_i U_j [G_{ij}\sin(\theta_i - \theta_j) - B_{ij}\cos(\theta_i - \theta_j),$$

  in which P_i refers to the active power injected, Q_i to the reactive power, U_i to the voltage magnitude and θ_i to the voltage angle;
- Bus voltage constraints:

$$U_i^{min} \leq U_i \leq U_i^{max}; \quad (10)$$

- Active and reactive power generation constraints:

$$P_g^{min} \leq P_g \leq P_g^{max}, \quad (11)$$
$$Q_g^{min} \leq Q_g \leq Q_g^{max};$$

- Branch apparent power flow constraint:

$$S_{ij}^{min} \leq S_{ij} \leq S_{ij}^{max}; \quad (12)$$

- Transformer tap constraints:

$$T_k^{min} \leq T_k \leq T_k^{max}, \quad (13)$$

- Shunt var constraints:

$$Q_k^{min} \leq Q_k \leq Q_k^{max}, \quad (14)$$

  in which T_k and Q_k are a discrete variables.

In this particular case there is also a constraint that is the difference between q_{ref} and q_{PCC} for normal conditions, the q_{PCC} is the actual reactive power injection at the PCC.

IV. Experiments and Results

C-DEEPSO algorithm is going to be tested for solving the described WPP/ORDP problem. Experiments are performed in order to highlight the importance of reactive power control problem, which is defined by progressive changes q_{ref} in one day period (24 hours). The variability of this period is defined by 15-minute intervals, totalizing 96 intervals, to which the ORDP problem must be solved. Figure 3 is an example of the characteristic behavior of output power in a WPP.

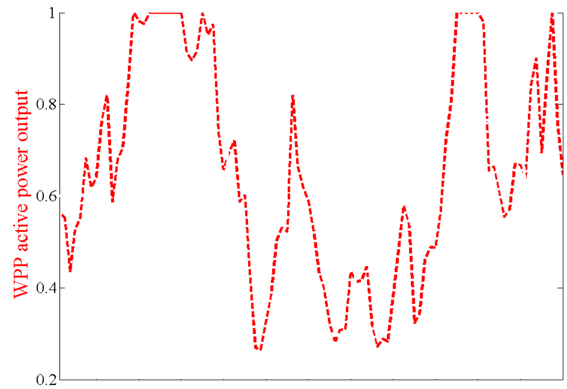

Fig. 3: WPP output power profile (behavior e. g.) [38].

While operating under ideal conditions, to guarantee the system availability, normal data acquisition and to continuously meet q_{ref}, a control strategy is needed. The experimental design is carried out to check if C-DEEPSO algorithm matches the control premises for the treatment of OPF-like on the WPP. C-DEEPSO is executed 31 times for each benchmark problem, using an Intel Xeon 2.4 Ghz and 12 GB RAM in Matlab.

There are many studies addressing the fine-tuning parameters in evolutionary algorithms. It is known that such procedure can ensure a better algorithm performance. However, the fine-tuning of C-DEEPSO parameters is out of scope of this work. That said, the parameters were empirically defined during the experiments, resulting on the values shown at Table II.

TABLE II: C-DEEPSO parameters setting.

Max Fit. Eval.	Pop. Size	Memory Size	Com. P	Mut. τ
10000	30	6	0.5	0.9

In order to validate the efficiency of C-DEEPSO solution to the problem, the obtained results are compared with the results of DEEPSO and MVMO algorithms. The results of these two algorithms were extracted from the database of Competition on Application of Modern Heuristic optimization algorithms for solving Optimal Power Flow problems [38].

The other algorithms participating of that competition were not considered in this comparison, because they violated the restrictions of the problem or failed to generate a characteristic power curve. According to the assumptions imposed by the competition, each algorithm must run for 31 times. The comparative graph of the mean result obtained by each algorithm is shown in Figure 4.

Figure 4 does not allow an effective comparative analysis. Figure 5 shows an expansion of this graphic in the range of scenarios 14-31. It can be seen that, in this interval, C-DEEPSO presents a better performance when compared to the other algorithms.

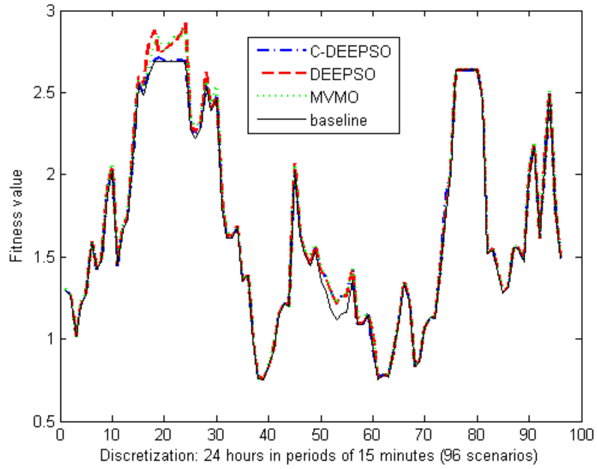

Fig. 4: Comparative WPP output power (mean results).

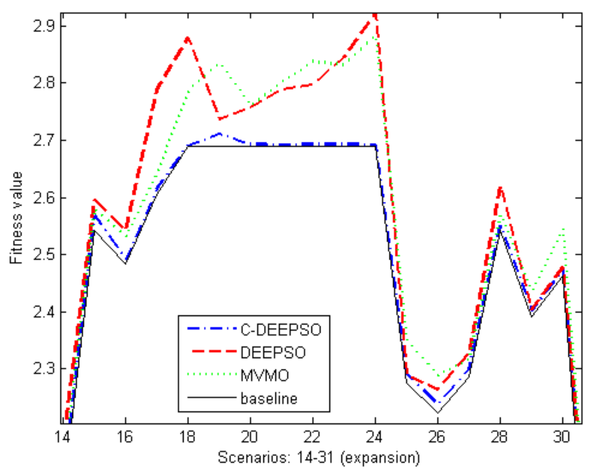

Fig. 5: Expansion (scenarios 14-31).

Table III provides another way to verify the algorithms results in each of 96 experimental test scenarios. In this table, the mean and standard deviation (std) values for each algorithm are listed.

Results indicate that, in terms of mean and std, C-DEEPSO performs better in 97% of experimental scenarios. However, an analysis based only on the mean and std represents a poor approach to compare results. In spite of the fact that mean and std values of C-DEEPSO are smaller than those of DEEPSO and MVMO, it is not possible to determine if these differences are statistically significant.

One-way analysis of variance, ANOVA [39], is a statistical technique used to verify whether there are any significant differences among the means of three or more samples. A hypothesis test is be used in the context of this experiment to verify the equality of means. Such test can be expressed formally as:

$$\begin{cases} H_0 & : \quad \mu_i = \mu_j, \forall \ i, j; \\ H_1 & : \quad \mu_i \neq \mu_j, \text{for} \quad \text{any } i. \end{cases}$$

TABLE III: Results by C-DEEPSO, DEEPSO and MVMO

Sc	ALGORITHMS					
	C-DEEPSO		DEEPSO		MVMO	
	Mean	STD	Mean	STD	Mean	STD
1	**1.296709**	**0.0011**	1.300416	0.006464	1.299471	0.002149
2	**1.2674**	**0.001079**	1.27365	0.006849	1.270922	0.002452
3	**1.013952**	**0.000889**	1.018528	0.009137	1.017974	0.00618
4	**1.206103**	**0.001241**	1.210449	0.006677	1.214193	0.012536
5	**1.274215**	**0.001054**	1.277351	0.004858	1.277518	0.00302
6	**1.58931**	**0.001891**	1.590727	0.002382	1.595016	0.002961
7	**1.427447**	**0.001014**	1.430635	0.003885	1.431654	0.003422
8	**1.486853**	**0.001425**	1.490068	0.002738	1.491214	0.001765
9	**1.890275**	**0.004331**	1.920608	0.025138	1.912675	0.017726
10	**2.036265**	**0.006961**	2.059875	0.021007	2.068124	0.024814
11	**1.448917**	**0.001581**	1.460489	0.019159	1.466775	0.028831
12	**1.676151**	**0.003887**	1.687921	0.00919	1.696736	0.020164
13	**1.740549**	**0.004298**	1.75454	0.009227	1.765304	0.019683
14	**2.105806**	**0.007327**	2.138487	0.02316	2.131292	0.015799
15	**2.553849**	**0.007303**	2.576392	0.015664	2.583475	0.017772
16	**2.492254**	**0.00333**	2.52295	0.022583	2.526806	0.015743
17	**2.622195**	**0.011828**	2.763067	0.063241	2.752978	0.049748
18	**2.698968**	**0.01222**	2.837714	0.049846	2.832812	0.065309
19	**2.701435**	**0.012939**	2.838913	0.054696	2.832115	0.05962
20	**2.703978**	**0.014009**	2.829544	0.062866	2.836847	0.037857
21	**2.7048**	**0.013638**	2.826354	0.054087	2.824264	0.052248
22	**2.701397**	**0.011375**	2.840033	0.041991	2.8105	0.043337
23	**2.702657**	**0.011834**	2.84924	0.052075	2.813017	0.060382
24	**2.698104**	**0.010336**	2.824618	0.060398	2.830624	0.038445
25	**2.291002**	**0.004759**	2.320457	0.026687	2.328435	0.025966
26	**2.238042**	**0.006057**	2.279125	0.018528	2.273852	0.022695
27	**2.301127**	**0.007693**	2.332162	0.019854	2.326763	0.016357
28	**2.554652**	**0.008919**	2.584022	0.024599	2.588382	0.025195
29	**2.404226**	**0.004125**	2.434756	0.025572	2.435477	0.016747
30	**2.472276**	**0.002946**	2.504222	0.019501	2.510542	0.021636
31	**1.818557**	**0.004294**	1.837128	0.013342	1.845527	0.031906
32	**1.619751**	**0.003618**	1.632586	0.012615	1.638392	0.024243
33	**1.61713**	**0.000828**	1.620574	0.005501	1.621487	0.002227
34	**1.686769**	**0.000694**	1.688831	0.003948	1.690077	0.002368
35	**1.356638**	**0.001714**	1.35859	0.003192	1.359256	0.00347
36	**1.393209**	**0.001513**	1.396066	0.003154	1.397859	0.003853
37	**0.984658**	**0.000513**	0.989143	0.008027	0.986252	0.002255
38	**0.759614**	**0.00117**	0.765184	0.015427	0.761642	0.002352
39	**0.75597**	**0.001058**	0.760265	0.012586	0.757799	0.003082
40	**0.838926**	**0.000668**	0.840169	0.007057	0.84189	0.005286
41	**0.924513**	**0.000716**	0.927213	0.007331	0.92718	0.003881
42	**1.153131**	**0.001052**	1.155496	0.005161	1.155485	0.002304
43	**1.217101**	**0.001165**	1.220947	0.006493	1.221542	0.004354
44	**1.196428**	**0.001065**	1.201751	0.009862	1.199745	0.002207
45	**2.033573**	**0.004264**	2.06253	0.018563	2.07081	0.025702
46	**1.633785**	**0.002592**	1.644981	0.009954	1.655683	0.025572
47	**1.524769**	**0.001657**	1.535093	0.010909	1.552997	0.033933
48	**1.449359**	**0.0015**	1.464665	0.021088	1.466773	0.021983
49	**1.554853**	**0.000269**	1.573228	0.014294	1.593472	0.038212
50	**1.419097**	**0.000124**	1.42171	0.004049	1.429751	0.015539
51	**1.372646**	**0.000169**	1.37299	0.000638	1.379473	0.009745
52	1.280405	4.42E-05	**1.280345**	**3.92E-05**	1.280361	5.04E-05
53	**1.21604**	**4.49E-05**	1.216045	5.26E-05	1.216075	7.43E-05
54	1.257865	1.97E-05	**1.257835**	**2.25E-05**	1.257843	2.72E-05
55	1.261093	3.11E-05	**1.261064**	**4.58E-05**	**1.261064**	**4.53E-05**
56	**1.425905**	**0.000193**	1.427878	0.002874	1.435186	0.016387
57	**1.090003**	**0.000897**	1.100627	0.019237	6.477143	29.84215
58	**1.091811**	**0.000938**	1.107108	0.024072	1.12865	0.037902
59	**1.149323**	**0.000986**	1.162567	0.025166	1.18516	0.032974
60	**0.942244**	**0.001247**	0.967602	0.041148	485.9867	2700.31
61	**0.759047**	**0.000937**	0.768336	0.019544	0.764196	0.009798
62	**0.78431**	**0.017395**	0.785943	0.010336	0.788255	0.01326
63	**0.7726**	**0.00102**	0.781982	0.018497	0.779951	0.012635
64	**0.919392**	**0.00066**	0.924948	0.016801	0.923803	0.0095
65	**1.11818**	**0.000359**	1.12153	0.004301	1.12021	0.002402
66	**1.342056**	**0.001249**	1.344351	0.002214	1.344502	0.001449
67	**1.236404**	**0.001176**	1.23958	0.004837	1.239307	0.001269
68	**0.83218**	**0.000808**	0.838725	0.014454	0.834139	0.003636
69	**0.864326**	**0.000728**	0.868957	0.010561	0.865998	0.002148
70	**1.074571**	**0.000791**	1.076649	0.004521	1.075708	0.001544
71	**1.132337**	**0.000823**	1.13391	0.003072	1.134448	0.004129
72	**1.130509**	**0.00048**	1.132922	0.003447	1.132384	0.002756
73	**1.421856**	**0.001654**	1.424401	0.00346	1.42228	0.002085
74	**1.783861**	**0.017107**	1.789148	0.02496	1.783307	0.003791
75	**2.020105**	**0.02545**	75.50458	409.1587	2.022251	0.002981
76	**55954.25**	**273071.8**	1269600	5761404	2.637258	3.74E-08
77	**147776.7**	**416206.1**	923281.7	5118422	2.637258	2.50E-08
78	**148703.8**	**488445.5**	646388.5	3598924	7085.962	39438.28
79	**2.637259**	**4.08E-07**	20074.95	111757.9	2.637258	3.01E-08
80	**2.637258**	**1.83E-07**	2.637258	7.48E-08	2.637258	3.01E-08
81	**2.463194**	**0.004594**	2.47379	0.03726	2.463748	0.006993
82	**1.520329**	**0.001776**	1.522664	0.003022	1.524768	0.00223
83	**1.544186**	**0.001679**	1.546459	0.002979	1.549251	0.001964
84	**1.407015**	**0.00195**	1.409471	0.002567	1.409581	0.001261
85	**1.279572**	**0.001034**	1.282229	0.003461	1.281927	0.001248
86	**1.320585**	**0.001055**	1.323782	0.003268	1.322905	0.00182
87	**1.558975**	**0.001598**	1.560763	0.002548	1.563525	0.00138
88	**1.552458**	**0.001414**	1.554608	0.002166	1.557436	0.002001
89	**1.472436**	**0.00126**	1.47573	0.003377	1.476867	0.001657
90	**2.007656**	**0.001314**	2.011905	0.006837	2.022072	0.006272
91	**2.181026**	**0.000514**	2.192417	0.012609	2.198064	0.005255
92	**1.615984**	**0.001982**	1.617162	0.002416	1.622292	0.003239
93	**2.056128**	**0.001679**	2.064203	0.010235	2.071865	0.00516
94	**2.492259**	**0.004349**	2.516119	0.023582	2.512312	0.007086
95	**1.782919**	**0.001653**	1.784854	0.003648	1.79233	0.005024
96	**1.49785**	**0.001625**	1.500072	0.002245	1.502958	0.002782

The null hypothesis, H_0, assumes the equality of the means and the alternative hypothesis, H_1, indicates that there are at least one mean which is not equal to the others. P-value is defined as the lowest level of significance that leads to rejection of H_0, with the provided data. It is useful for reporting results of a hypothesis test, since it carries out more information than simply the rejection or the failure to reject H_0. When One-Way ANOVA shows a significant result, this indicates that at least one group is different from the others.

The significance level adopted to verify whether there are statistical differences among C-DEEPSO, DEEPSO and MVMO algorithms is set to 5%. Considering each scenario, if the P-value in the ANOVA is less than 0.05, then it is possible to say that there is sufficient statistical evidence to reject H_0 meaning that there is a statistical difference between the means. Otherwise, the null hypothesis can not be rejected. Although One-Way ANOVA can determine whether there are significant differences among the means of three or more samples, it does not have an indication of which group is different. A simple paired comparison technique, known as Tukey Test or Honestly Significant Difference (HSD), can be used to find means that are different from the others [40].

Table IV shows the obtained P-values for each scenarios using One-Way ANOVA and a classification given by the Tukey Test, when necessary. The results show that, in 17 out of 96 scenarios, or in 17.7% of the cases. Since P-value is higher than 0.05, it can be said that the algorithms have the same performance for solving the problem. However, in 79 out of 96 scenarios, or in 82.3% of the scenarios, One-Way ANOVA is able to identify that there are statistical differences among the algorithm means. In those cases, Tukey test is performed to rank the algorithms.

As a final result, after the application of One-Way ANOVA and Tukey Test, in 60.4% of the scenarios C-DEEPSO is classified into the first position when compared to DEEPSO and MVMO. In 18.8 % of the scenarios, C-DEEPSO is tied at first position with other algorithm. In 3.1% of the scenarios, C-DEEPSO is classified at the third position, being worst than the other algorithms. It is worthwhile to notice that in 79.2% of the scenarios, C-DEEPSO has performed better than or equal to the state-of-art algorithm, MVMO. It can be seen that C-DEEPSO is a competitive algorithm, which is able to perform optimal control of the operation of a WPP for a 24h period, minimizing transmission losses and ensuring adjustment of all variables to meet the reactive power requirements at PCC.

V. Conclusion

The problem, known as Optimal Power Flow (OPF-like) for wind power generation, can be expressed as the minimization of the cost of production of a power plant, a Wind Power Plant. In order to maintain a clean and sustainable electric energy system and since the hydraulic power alone is not able to expand the renewable energy supply, wind power has become a viable energy source in Brazil. Despite the great progress made in this sector, considering off-shore and on-shore wind

TABLE IV: Results of ANOVA and Tukey tests.

Sc	P-value	CLASSIFICATION		
		First	Second	Third
1	1.20E-03	C-DEEPSO	DEEPSO; MVMO	-
2	5.93E-07	C-DEEPSO	DEEEPSO	-
3	1.11E-02	C-DEEPSO	DEEPSO; MVMO	-
4	1.00E-04	C-DEEPSO	DEEPSO	-
5	2.00E-04	C-DEEPSO	DEEPSO; MVMO	-
6	2.15E-14	C-DEEPSO; DEEPSO	MVMO	-
7	1.06E-06	C-DEEPSO	DEEPSO; MVMO	-
8	1.36E-12	C-DEEPSO	DEEPSO; MVMO	-
9	4.88E-09	C-DEEPSO	DEEPSO; MVMO	-
10	8.54E-09	C-DEEPSO	DEEPSO; MVMO	-
11	2.60E-03	C-DEEPSO; DEEPSO	MVMO	-
12	8.60E-08	C-DEEPSO	DEEPSO	MVMO
13	1.70E-10	C-DEEPSO	DEEPSO	MVMO
14	2.16E-11	C-DEEPSO	DEEPSO; MVMO	-
15	2.88E-12	C-DEEPSO	DEEPSO; MVMO	-
16	6.49E-14	C-DEEPSO	DEEPSO; MVMO	-
17	9.87E-22	C-DEEPSO	DEEPSO; MVMO	-
18	3.07E-21	C-DEEPSO	DEEPSO; MVMO	-
19	3.50E-21	C-DEEPSO	DEEPSO; MVMO	-
20	1.28E-22	C-DEEPSO	DEEPSO; MVMO	-
21	3.12E-20	C-DEEPSO	DEEPSO; MVMO	-
22	1.95E-27	C-DEEPSO	MVMO	DEEPSO
23	3.19E-21	C-DEEPSO	MVMO	DEEPSO
24	1.57E-23	C-DEEPSO	DEEPSO; MVMO	-
25	1.52E-09	C-DEEPSO	DEEPSO; MVMO	-
26	1.02E-15	C-DEEPSO	DEEPSO; MVMO	-
27	4.39E-12	C-DEEPSO	DEEPSO; MVMO	-
28	5.32E-09	C-DEEPSO	DEEPSO; MVMO	-
29	5.38E-11	C-DEEPSO	DEEPSO; MVMO	-
30	2.05E-14	C-DEEPSO	DEEPSO; MVMO	-
31	3.26E-06	C-DEEPSO	DEEPSO; MVMO	-
32	4.77E-05	C-DEEPSO	DEEPSO; MVMO	-
33	6.49E-06	C-DEEPSO	DEEPSO; MVMO	-
34	2.44E-05	C-DEEPSO	DEEPSO; MVMO	-
35	1.70E-03	C-DEEPSO	DEEPSO. MVMO	-
36	1.42E-07	C-DEEPSO	DEEPSO	MVMO
37	1.60E-03	C-DEEPSO; MVMO	DEEPSO	-
38	**0.0539***	-	-	-
39	**0.0832***	-	-	-
40	**0.0773***	-	-	-
41	**0.0543***	-	-	-
42	7.20E-03	C-DEEPSO	DEEPSO; MVMO	-
43	4.00E-04	C-DEEPSO	DEEPSO; MVMO	-
44	2.30E-03	C-DEEPSO	DEEPSO; MVMO	-
45	6.59E-12	C-DEEPSO	DEEPSO; MVMO	-
46	3.04E-06	C-DEEPSO	DEEPSO	MVMO
47	2.57E-06	C-DEEPSO; DEEPSO	MVMO	-
48	3.00E-04	C-DEEPSO	DEEPSO; MVMO	-
49	3.63E-08	C-DEEPSO	DEEPSO	MVMO
50	4.84E-05	C-DEEPSO; DEEPSO	MVMO	-
51	3.69E-06	C-DEEPSO; DEEPSO	MVMO	-
52	2.96E-06	DEEPSO; MVMO	C-DEEPSO	-
53	**0.0543***	-	-	-
54	5.02E-06	DEEPSO; MVMO	C-DEEPSO	-
55	7.20E-03	DEEPSO; MVMO	C-DEEPSO	-
56	6.00E-04	C-DEEPSO; DEEPSO	MVMO	-
57	**0.3689***	-	-	-
58	1.32E-06	C-DEEPSO; DEEPSO	MVMO	-
59	3.20E-07	C-DEEPSO; DEEPSO	MVMO	-
60	**0.3719***	-	-	-
61	1.79E-03	C-DEEPSO; MVMO	DEEPSO	-
62	**0.5379***	-	-	-
63	1.36E-02	C-DEEPSO; MVMO	DEEPSO	-
64	**0.1229***	-	-	-
65	5.98E-05	C-DEEPSO	DEEPSO; MVMO	-
66	4.89E-08	C-DEEPSO	DEEPSO; MVMO	-
67	5.70E-05	C-DEEPSO	DEEPSO; MVMO	-
68	1.13E-02	C-DEEPSO; MVMO	DEEPSO	-
69	1.53E-02	C-DEEPSO; MVMO	DEEPSO	-
70	1.66E-02	C-DEEPSO; MVMO	DEEPSO	-
71	1.94E-02	C-DEEPSO; DEEPSO	MVMO	-
72	9.00E-04	C-DEEPSO	DEEPSO; MVMO	-
73	3.00E-04	C-DEEPSO; MVMO	DEEPSO	-
74	**0.3578***	-	-	-
75	**0.3721***	-	-	-
76	**0.2426***	-	-	-
77	**0.5235***	-	-	-
78	**0.5544***	-	-	-
79	**0.3719***	-	-	-
80	**0.7213***	-	-	-
81	**0.1093***	-	-	-
82	8.56E-10	C-DEEPSO	DEEPSO	MVMO
83	8.35E-13	C-DEEPSO	DEEPSO	MVMO
84	8.83E-07	C-DEEPSO	DEEPSO; MVMO	-
85	7.61E-06	C-DEEPSO	DEEPSO; MVMO	-
86	6.33E-70	C-DEEPSO	DEEPSO; MVMO	-
87	3.40E-14	C-DEEPSO	DEEPSO	MVMO
88	3.56E-16	C-DEEPSO	DEEPSO	MVMO
89	4.82E-11	C-DEEPSO	DEEPSO; MVMO	-
90	6.21E-17	C-DEEPSO	DEEPSO	MVMO
91	1.44E-12	C-DEEPSO	DEEPSO	MVMO
92	1.15E-15	C-DEEPSO; DEEPSO	MVMO	-
93	8.16E-14	C-DEEPSO	DEEPSO	MVMO
94	3.32E-09	C-DEEPSO	DEEPSO; MVMO	-
95	1.86E-16	C-DEEPSO; DEEPSO	MVMO	-
96	4.75E-13	C-DEEPSO	DEEPSO	MVMO

generation, it is necessary to guarantee optimal uses of the wind potential. The solution of the Optimal Power Flow problem to optimize the active power losses of the transmission network within wind farms for wind power generation is proposed using a hybrid metaheuristic. This work proposes a new hybrid algorithm, C-DEEPSO, which corresponds to a single-objective metaheuristic incorporating some features of Evolutionary Computation, Particle Swarm Optimization and Differential Evolution. C-DEEPSO is applied to a well-studied real world large-scale problem at the power systems industry. The proposed algorithm is compared to the baseline algorithm, DEEPSO, and to the reference algorithm, MVMO. The results indicate that the proposed algorithm is efficient and competitive, capable to tackle this difficult problem. The experimental results also show that the new approach exhibits better results, when compared to the reference algorithm.

Acknowledgment

The authors would like to thank CEFET-MG and INESC TEC for the infrastructure used for this project, and CAPES, CNPq and FAPEMIG for financial support. They also thank to all researchers at CEFET-MG and INESC TEC that contributed to this work. This work is financed too by the ERDF - European Regional Development Fund through the Operational Programme for Competitiveness and Internationalization - COMPETE 2020 Programme within project POCI-01-0145-FEDER-006961, and by National Funds through the FCT - Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) as part of project UID/EEA/50014/2013.

References

[1] D. E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley, 1989.
[2] M. Dorigo, V. Maniezzo, and A. Colorni. The ant system: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics, 26(1):29-41*, 1996.
[3] J. Kennedy and R. Eberhart. Particle swarm optimization. *Proc. of IEEE International Conference on Neural Networks, volume 4, pages 1942-1948*, 1995.
[4] C.D. Gelatt S. Kirkpatrick and M.P. Vecchi. Optimization by simulated annealing. *Science Magazine, 220(4598):671*, 1983.
[5] K. Price, R. Storn, and J. Lampinen. Differential evolution: A practical approach to global optimization. natural computing series. *Springer*, 2005.
[6] Rainer Storn and Kenneth Price. Differential evolution: a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization 11 (4), pages 341-359*, 1995.
[7] C. Coello, D. Veldhuizen, and G Lamont. Evolutionary algorithms for solving multi-objective problems. *Kluwer Academic Publishers, New York, USA*, 2002.
[8] Y. Liu, X. Yao, Q. Zhao, and T. Higushi. Scaling up fast evolutionary programming with cooperative coevolution. *Proc. of IEEE Congress on Evolutionary Computation, pages 1101-1108*, 2001.
[9] X. Li, K. Tang, M. Omidivar, Z. Yamg, and K. Qin. Benchmark functions for the cec'2013 special session and competition on large-scale global optimization. Technical report, IEEE CEC, 2013.
[10] W. Zhang and X. Xie. Depso: Hybrid particle swarm with differential evolution operator. *Proc: In IEEE conference on Systems, Man and Cybernetics*, 2003.
[11] Z. Hao, G. Guo, and H. Huang. A particle swarm optimization algorithm with differential evolution. *Proc: In 6Th International Conference on Machine Learning and Cybernetics. pp: 1031-1035*, 2007.
[12] V. Miranda and R. Alves. Differential evolutionary particle swarm optimization (deepso): a successful hybrid. In *Proc: 11th Brazilian Congress on Computational Intelligence (BRICS-CCI). pp 368-374*, 2013.
[13] S. Frank, I. Steponavice, and S. Rebennack. Optimal power flow: a bibliographic survey (i) - fodeterministics methods and deterministic methods. *Energy Systems. Springer. V.3:221-258.*, 2012.
[14] M.A. Abido. Optimal power flow using particle swarm optimization. *Eletrical Power and Energy Systemns - ELSEVIER*, 24:563–571, 2002.
[15] M. Bhaskar, S. Muthyala, and S. Maheswarapu. Security constraint optimal power flow (scopf) - a compreensive survey. *International Journal of Computer Applications. V.2- N6. Pg: 42-52*, 2010.
[16] R. Jabr and B. Pal. Intermittent wind generation in optimal power flow dispatching. *IET Generation, Transmission & Distribution. Vol. 3, No. 1, pp. 66-74*, 2009.
[17] M. Georgescu, L. Barote, C. Marinescu, and L. Clotea. Smart electrical energy storage system for small power wind turbines. In *Proc in: 12th International Conference on Optimization of Electrical and Electronic Equipment, OPTIM, pp: 1192-1197*, 2010.
[18] C. Marcelino, P. Almenda, E. Wanner, L. Carvalho, and V. Miranda. Application of evolutionary multiobjective algorithms for solving the problem of energy dispatch in hydroelectric power plants. *Lecture Notes in Computer Science: 9019. Springer. pp. 403-417*, 2015.
[19] S. Frank, I. Steponavice, and S. Rebennack. Optimal power flow: a bibliographic survey (ii) - non-deterministics and hybrid methods. *Energy Systems. Springer. V.3:259-289.*, 2012.
[20] K. Pandya and S. Joshi. A survey of optimal power flow methods. *Journal of Theoretical and Applied Information Technology. JATIT: 450-458*, 2008.
[21] N. Rau. Issues in the path toward an rto and standard markets. *IEEE Transactions on Power Systems. 18(2), 435-443*, 2003.
[22] Jardim J. Alsac O. Stott, B. Dc power flow revisited. *IEEE Transactions on Power Systems. 24(3), 1290-1300*, 2009.
[23] Pardalos P. Thoai N.V Horst, R. Introduction to global optimization. *2nd edn. Springer, Berlin*, 2000.
[24] Happ H. Wirgau K Burchett, R. Large scale optimal power flow. *IEEE Transactions Apparatus and Systems 101(10), 3722-3732*, 1982.
[25] Liu Y.H Teng, J.H. A novel acs-based optimum switch relocation method. *IEEE Transactions on power systems 18(1), 113-120*, 2003.
[26] Allaoua B. Gasbaoui, B. Ant colony optimization applied on combinatorial problem for optimal power flow solution. *Leonardo Journal of Science 14, 1-17*, 2009.
[27] Rahman T. Musirin I. Aminudin, N. Optimal power flow for load margin improvement using evolutionary programming. In *In: The 5th Student Conference on Research and Development (SCOReD2007), Malaysia. pp:1-6*, 2007.
[28] Chung C. Wong K. Duan X. Tse C Liang, C. Study of differential evolution for optimal reactive power flow. *IET Gener. Transm. Distrib. 1(2), 253-260*, 2007.
[29] Ramirez J. Coello C. Yumbla, P. Optimal power flow subject to security constraints solved with a particle swarm optimizer. *IEEE Transactions on Power Systems, 23(1), 33-40*, 2008.
[30] M Georgescu, M. Barote, C. Marinescu, and L. Clotea. Smart electrical energy storage system for small power wind turbines. In *12th International Conference on Optimization of Electrical and Electronic Equipment, OPTIM 2010. V.1. pp 1192- 1197*, 2010.
[31] L. Shi, C. Wang, L. Yao, N. Yixin, and M. Bazargan. Optimal power flow solution incorporating wind power. *IEEE Systems Journal, VOL. 6, NO. 2, JUNE, pp: 233-241*, 2012.
[32] L. Shi and G. Xu. Self-adaptive evolutionary programming and its application to multi-objective optimal operation of power systems. *Electric Power Systems Research. ELSEVIER, 57, pp: 181-187*, 2001.
[33] S Joseph, E. Skariah, T. Joseph, S. Sreedharan, C. Chittesh, P. Vipin, and J. Vishnu. Pso based controller algorithm for optimal allocation & setting of fuel cell in a wind - pv integrated power system for maximizing loadability. In *ICAGE, December*, 2014.
[34] W. Bai, I. Eke, and Lee. Heuristic optimization for wind energy integrated optimal power flow. In *IEEE Power & Energy Society General Meeting, V1, pp: 1-5*, 2015.
[35] H. Pham, J. Rueda, and I. Erlich. Online optimal control of reactive sources in wind power plants. *IEEE Transactions on Sustainble Energy, Vol.5, NO.2*, 2014.
[36] 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 (CEC), vol. 2, pp. 10801085*, 2002.
[37] S. Frank, I. Steponavice, and S. Rebennack. Optimal power flow: a bibliographic survey. *Energy Systems, Springer*, 3:221–258, 2012.
[38] I. Erlich, K. Lee, J. Rueda, and S. Wildenhues. Competition on application of modern heuristic optimization algorithms for solving optimal power flow problems. Technical report, 2014.
[39] D. Montgomery. *Design and analysis of Experiments*. 8th edition, 2012.
[40] H. Abdi and L. William. Tukey's honestly significant difference (hsd) test. Technical report, The university of Texas at Dallas, 2010.