

Adaptive Imperialist Competitive Algorithm Applied to Wind Farm Layout Optimization

Sinvaldo Rodrigues Moreno¹
Olicio da Silva Jr.¹
and Leandro dos Santos Coelho^{1,2*}

¹ Department of Electrical Engineering, Federal University of Parana
Zip code 81531-980, Curitiba, PR, Brazil,

² Industrial and Systems Engineering Graduate Program (PPGEPS)
Pontifical Catholic University of Parana (PUCPR)
Imaculada Conceicao, 1155, Zip code 80215-901, Curitiba, PR, Brazil

* corresponding author
Tel/fax numbers: (55) 41 32711345
Email: leandro.coelho@pucpr.br

Abstract: In this work, an adaptive scheme is incorporated into the Competitive Imperialist Algorithm (ICA). The proposed ICA is applied to find the best positioning of wind turbines into wind farm layout. The objective is to minimize the wake effect in a given turbine, due to interactions with other turbines in the wind farm. As result of the optimization layout, the annual energy production is maximized. Another approach adopted is related to the general formulation of the problem, where the wind farm area is considered continuous domain in the solution space of the optimization process, instead of dividing it into small cells, as usually found in the literature. Two different configurations of wind farms and wind scenarios were applied to validate the proposed approach. Due to the promising results, we can conclude that it is possible to minimize the wake effect loss with a small number of iterations.

Keywords: Wake effect, wind farm layout optimization, energy maximization, metaheuristics.

1 Introduction

The renewable energy sources, particularly wind power is becoming more popular because of ecological concerns and continuously rising prices of rapidly depleting conventional energy sources. Therefore, wind energy technology has evolved rapidly over the last three decades with increasing rotor diameter and the use of sophisticated power electronics to allow operation at variable rotor speed. In this context, many research works were carried out to maximize energy production of wind turbines. The main issues addressed in the recent works are related to wind farm layout optimization (WFLO), whereas the goal is minimizing the wake loss effect. The two fundamental impacts of a wake effect into wind farm can be cited as decreasing in the wind speed, which thusly diminishes the generation of the wind farm and an expansion on turbulence of the wind at downstream turbines [1].

For optimal wind turbines placement which is a combinatorial problem, many researchers proposed heuristic optimization techniques to determine the near optimal wind turbine positions into wind farm. The goal is increasing the potential of wind energy production [2, 3]. Mosetti et al. [4] were the first who addressed the optimal wind turbine placement into WFLO problem. They used

genetic algorithm (GA) for three different wind scenarios. Following Mosetti's work, Grady et al. [5] also employed GA to solve the wind farm layout optimization to achieve perfect locations in the wind farm. Marmidis et al. [6] applied Monte Carlo Simulation (MCS) considering only the case of constant wind speed and direction. A comparative study in [7], which applied GA to solve the wind farm problem, also consider these three previous results (Mosetti, Grady and Marmidis) as reference. Application of ANFIS (adaptive neuro-fuzzy inference system) for predicting the wake power and wind speed deficit was applied by [8] with promising result. The investigation conducted by [9] also applied ANFIS to predict the wake added turbulence.

In this work, we propose a modified version of an evolutionary algorithm known as Imperialist Competitive Algorithm (ICA), which comprises an adaptive actualization of the parameters. We named it as aICA (**a** stands for adaptive) and applied it to solve the WFLO problem. The ICA is a metaheuristic optimization method that is based on modeling the attempts of countries to dominate other countries [10]. The results from aICA are compared to other metaheuristics such as Particle Swarm Algorithm (PSO), Cultural Algorithm (CA) and the canonical version of ICA. By comparing the final cost, which is related to the optimal solution per unit of power, we can infer that the present approach is suitable and very promising to handle with wind turbine placement into WFLO.

The remainder of this paper is organized as follows. Section 2 presents the fundamentals of Jensen's wake model. In Section 3, we introduce the ICA optimization algorithm and the proposed approach. The numerical procedure is presented in Section 4. Results are presented in Section 5, final comments and conclusion in Section 6.

2 Jensen's Wake Model

The wake model proposed by Jensen [11] is based upon the assumption of a linearly expanding wake diameter. The wake speed U_i is the only variable in function of the downwind distance y , meaning that wake velocity profile along the crosswind direction at a certain downstream position is uniform. Figure 1 schematically illustrates the overview of Jensen's wake model. Through this figure can be noted that the wake expand linearly with downstream distance y . Which means, there is a relation between the diameter of the wake stream D_{wake} and y . Equation 1 gives the effective diameter of the wake stream D_{wake} at the downwind turbine. It is worth mentioning that the wind velocity at the downwind turbine and in the regions immediately beyond it, which are not covered by the wake effect, remains equals to U_0 (as presented on Figure 1),

$$D_{wake}(y) = \sqrt{\beta} D_{wt} \left(1 + 2k_d \frac{y}{D_{wt}} \right) \quad (1)$$

$$\beta = \frac{1}{2} + \frac{1}{2\sqrt{1-c_t}} \quad (2)$$

$$k_d = \frac{1}{2\ln\left(\frac{h}{z_0}\right)} \quad (3)$$

where y is the downstream distance of the WT_j (downwind turbine) to WT_i (upwind turbine); D_{wake} is the wake diameter at distance y behind the WT_i upwind turbine; D_{wt} is the turbine rotor diameter; β is the entrainment constant ($\beta > 1$), and it represents the initial expansion of the wake, which is

described as given Equation 2, where C_t is the thrust coefficient. The wake decreasing constant k_d is given by Equation 3, where h and z_0 are hub height and surface roughness, respectively.

The value of surface roughness is variable according to terrain characteristics. For plain terrains, the surface roughness assume the value equals to 0.3 [4]. The thrust coefficient C_t is given by Equation 4, where F_t is the thrust force ($kg.m/s^2$), ρ is the air density (kg/m^3), and r_0 is the rotor radius. Usually the datasheet of wind turbine provide all values of C_t . The final wake speed at the downwind location can be obtained by Equation 5. In this equation the wind velocity inside of the single turbine wake area is considered as function of the incoming wind speed U_0 , when downwind turbine is completely shadowed by upwind turbine.

$$C_t = \frac{F_t}{\rho \pi r_0^2 U_0^2} \quad (4)$$

$$U_i = U_0 \left(1 - \sqrt{1 - C_t}\right) \left(\frac{D_{wt}}{D_{wake}}\right)^2 \quad (5)$$

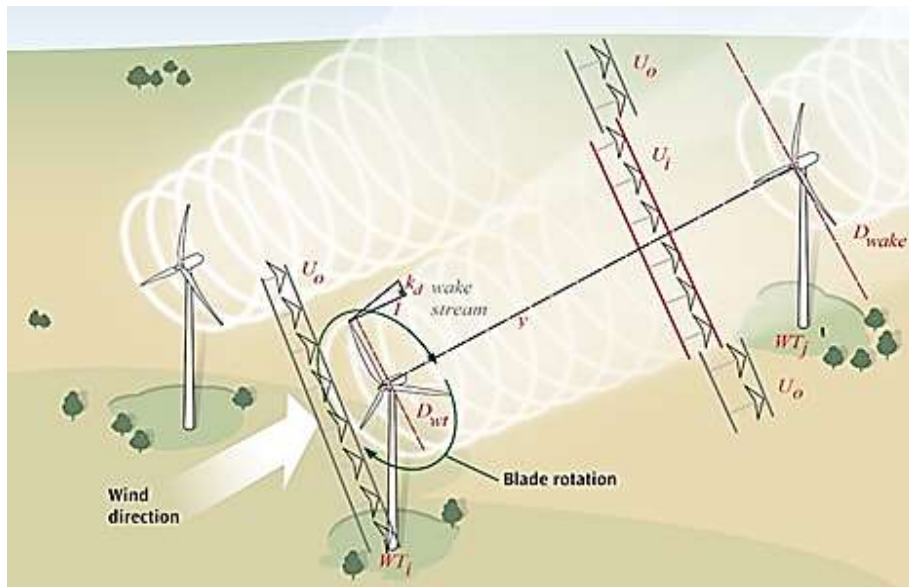


Figure 1 – Wake Scheme

If the downwind turbine is not completely immersed into wake, the equation that represents the wake model must be adjusted to represent the wake loss. Considering A_w as the partial rotor area that is inside the upwind turbine wake, and the rotor swept area A , so the wake speed at the down-wind location, locate at distance y from the upwind turbine, can be represented as function of the incoming wind speed U_0 , as Equation 6

$$U_i = U_0 \left(1 - \sqrt{1 - C_t}\right) \left(\frac{D_{wt}}{D_{wake}}\right)^2 \left(\frac{A_w}{A}\right) \quad (6)$$

This work considers all wind turbines having the same diameter $D_{wt} = 2r_0$, then the area of the turbine being shadowed can be calculated using the following basic trigonometric relationships [12] presented in Equations 7, 8 and 9

$$A_w = R_{wake}^2 \left(\gamma_R - \frac{\sin(2\gamma_R)}{2} \right) + r_0^2 \left(\gamma_r - \frac{\sin(2\gamma_r)}{2} \right) \quad (7)$$

$$\gamma_R = \cos^{-1} \left(\frac{R_{wake}^2 + x^2 - r_0^2}{2xR_{wake}} \right), \quad \forall x \in [R_{wake} - r_0, R_{wake} + r_0] \quad (8)$$

$$\gamma_r = \cos^{-1} \left(\frac{R_{wake}^2 - x^2 - r_0^2}{2xR_{wake}} \right), \quad \forall x \in [R_{wake} - r_0, R_{wake} + r_0] \quad (9)$$

In Equation 7 the value of A_w is dependent on x distance, which is the center distance between WT_j to WT_i . Therefore, when the distance $x \leq (R_{wake} - r_0)$ the shadowed area A_w is equals to rotor area A ; also if $x \geq (R_{wake} + r_0)$ the shadowed area becomes $A_w = 0$. Generalizing the Equation 6 to multiple wind turbines yield on Equation 10

$$U_j = U_0 \left(1 - C_r \frac{D_{wt}^2}{A} \left(\sum_{i=1}^n \frac{\Psi_{ij}}{D_{wake_{i,j}}^2} \right) \right) \quad (10)$$

$$C_r = 1 - \sqrt{1 - C_t} \quad (11)$$

where Ψ_{ij} represent the overlap area A_w , which is either less than or equal to the area A of the upwind turbine, and it will assume the following values: $\Psi_{ij} = 1$ for $\forall A_w = A$; $\Psi_{ij} = A_w$ for $\forall A_w \in 0 \leq A_w < A$; n is the total number of upwind turbines which affects the downwind turbine j .

3 Optimization Algorithms

Aiming to compare the final results from the proposed approach, three algorithms are applied in this work. The first is the well-known Particle Swarm Optimization (PSO), which was firstly introduced in 1995 by Eberhart and Kennedy [13] and has large application into engineering and science fields. The second is based on the work of Reynolds [14], named as Cultural Algorithm (CA), and the last, the canonical version of Imperialist Competitive Algorithm (ICA) [10]. In the following section detailed information about ICA algorithm and the proposed adaptive scheme is described. For those readers who desire full explanation about PSO and CA, we recommend the references [13-14]. Also Matlab™ source codes for all algorithms are available for download in Yarpiz Project's home page [15].

3.1 Imperialist Competitive Optimization Algorithm

ICA is a socio-politically inspired global search strategy that has been introduced in 2007 by Atashpaz-Gargari and Lucas [10]. It is good to mention that ICA is a robust method based on imperialism which is the policy of extending the power and rule of a government beyond its own borders. The main idea of the optimization process can be described in simplified form as: the algorithm starts with an initial population as initial countries; some of the best countries are selected to be the imperialist states and

the rest form the colonies of these imperialists. The colonies of initial countries are distributed among the mentioned imperialists regarding their power. The power of each country, the counterpart of fitness value in the genetic algorithm, is inversely proportional to its cost.

The imperialist states together with their colonies form some empires. Then, the imperialistic competition begins among all the empires. The weakest empire which cannot increase its power and is not able to succeed in this competition will be eliminated from the competition. As a result, all colonies move toward their relevant imperialists along with the competition among empires. Finally, the collapse mechanism will lead all the countries to converge to a state in which there exists just one empire in the domain of the problem, and all the other countries are colonies of that one empire. On the following sections, we describe two main steps performed by ICA algorithm during the optimization process. The flowchart of ICA is presented in Figure 2 [10], detailed information about ICA can be found in [11, 16, 17].

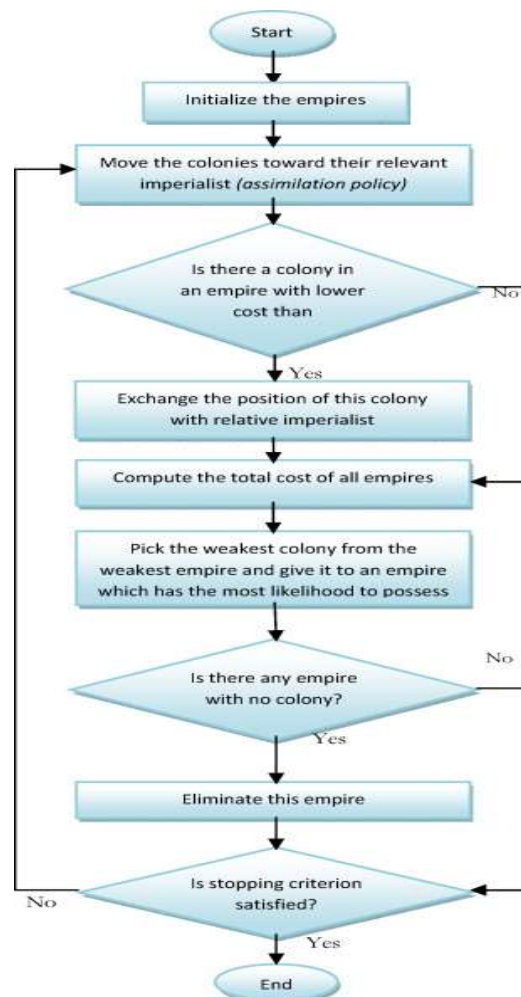


Figure 2 - Flowchart of ICA [10]

3.1.1 Assimilation Step

Each colony moves toward the imperialist by x -units, and this movement is known as assimilation step. The movements toward the one specific imperialist can be performed only the colonies that belong to this empire. So, they are encouraged to move toward as described in Equation 12, where r is a random variable with uniform distribution between 0 and 1; d is the distance between colony and imperialist; and β is one of ICA control parameter called assimilation coefficient, which causes the colony to get closer to the imperialist. Usually $1 < \beta \leq 2$, high values can result in oscillatory behavior.

$$x_{new} = x_{old} + U(0, \beta * d) \quad (12)$$

3.1.2 Revolution Step

Revolution brings sudden random changes in the position of some the colonies in the search space, as its counterpart in genetic algorithm, mutation, and it is modeled as a random change in one or more coordinates of a country. Revolution step in ICA is controlled by 3 different parameters: revolution probability pR , which indicates the probability of the revolution occurrence, and its value usually, is less than 1; revolution rate Rr , which specifies how many of the coordinates of a country are affected by the revolution. A very high value of revolution decreases the exploitation power of algorithm and can reduce its convergence rate. Usually, the revolution rate is a small percentage (10 a 30%) value of the total empires [17]; the last parameter, the revolution effect σ (or step length for some authors) specify the domain in which the country coordinate can change in the revolution process, and its value is often considered to be equal to $0.1 * (x_{max} - x_{min})$ as presented in [10].

3.2 Adaptive Scheme Proposed

There are two parameters of ICA algorithm, β in Assimilation step and pR in Revolution step, which can affect significantly the exploration and convergence capability of the ICA. Considering the fact that these two parameters can increase the algorithm's ability of exploration, which helps the algorithm escape from local minima [10, 16, 17], we propose and adaptive scheme to actualize these parameters, which is described as follows.

3.2.1 Assimilation Coefficient β - Actualization Scheme

The assimilation coefficient β usually is as constant parameter into ICA algorithm, and it has value assigned equals to 1.5 or 2 on literature [10, 16, 17]. However, β value has impact direct on the movements towards the imperialist direction, and this movement has dependency of the actual distance d between colonies and Imperialist, which varies for each Imperialist. Therefore, we proposed a random tuning scheme, which follows a normal distribution with mean μ equals to the initial value assigned to β , and constant standard deviation δ . The procedure of actualization occurs in step 7 of the Pseudocode 1, just after the first movement towards an Imperialist in Assimilation step, as presented below:

- 1: **For** each Imperialist i until N do the following steps
- 2: **For** each colony $j \in$ Imperialist i do the following steps
- 3: Move the colony towards the Imperialist applying Eq. (12)

```

4:      Evaluate the Cost of the Imperialist  $i$  with new position of the colony  $j$ 
5:       $j = j + 1$ 
6:  End For
      % Adaptive process of  $\beta$  actualization
7:  Actualize  $\beta_{i+1} = \text{normrnd}(\beta, \delta)$ , where  $\delta = 0.25$  and  $\beta = 1.5$ 
8:   $i = i + 1$ 
9: End For

```

Pseudocode 1 – Actualization scheme proposed for the assimilation coefficient β

3.2.2 Revolution Probability pR - Actualization Scheme

The revolution increases the exploration of the algorithm and prevents the early convergence of countries to local minimums. The revolution probability pR in the algorithm works as a trigger to start the revolution in each iteration, where the colonies can change their position randomly. Aiming to increase the algorithm performance, we introduce an active scheme of actualization, presented in Pseudocode 2, which occurs at each iteration of the ICA algorithm.

```

1: For each iteration  $k$  until  $m$  do the following steps
2:    $c \sim \text{rand}(0,1)$ ;
3:    $cr = \text{normrnd}(\mu, \delta)$ ; where  $\mu = 0.05$  and  $\delta = 0.01$ 
4:    $sCR = \text{randn}(1, nEmp)$ ; where  $nEmp$  is the number of Empires
5:    $pR = (1 - c) * cr + \text{mean}(sCR)$ ;
6: End For

```

Pseudocode 2 – Actualization scheme proposed for the Revolution Probability pR

4 Numerical Procedures and Problem Setup

As mentioned before, this work compare results from PSO, CA, ICA and aICA algorithms. Thereby the objective function is given by

$$F = \min (f_{xy}) = \frac{N_t(\frac{2}{3} + \frac{1}{3}e^{-0.00174 t^2})}{P_{total}} \quad (13)$$

$$P_{total} (kWh \text{ year}) = 8760 \sum_{k=0}^{360} \sum_{i=1}^{N_t} f_k(\theta) P_i(U_i(\theta)) \quad (14)$$

Subject to:

$$\begin{aligned} \sum_{i=1}^{N_t} \sum_{\substack{j=1 \\ j \neq i}}^{N_t} \max\left(6D_{wt} - \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}, 0\right) &\leq 0 \\ x_i &\geq D_{wt} \\ y_i &\geq D_{wt} \\ x_i &\leq 2000 - D_{wt} \\ y_i &\leq 2000 - D_{wt} \end{aligned} \quad (15)$$

where f_{xy} is the Cost of annual Energy (CoE), given by Eq. 13 and the term N_t is the maximum number of wind turbines that make up the wind farm, P_{total} is the power taking out from N_t turbines at certain free of wake loss wind speed $U_i(\theta)$ - given by Eq. 10 - blowing at direction θ , following the power curve of the wind turbine as per Equation 16, it means the summation of the terms $P_i(U_i(\theta))$ as Eq. 14. The $f_k(\theta)$ term represents the probability of occurrence of wind speed to be blowing at direction θ . Therefore, considering this probability as discrete bin of 10° , as Figure 3, its necessary sum up until 360° and multiplies by the total hours of the year to get the annual production. The following constrains in Eq. 15, which the Eq. 13 is subject to, comprises the wind farm bounds, to ensure the placement of the wind turbines within the fixed size wind farm, and safe constraints in wind turbine operations (roads, transposing land frontiers from different owners, etc.).

The wind farm parameters are presented in Table 1. Two wind speed scenarios were applied in this work: Scenario A considers 12m/s at multidirectional direction and different probability of occurrence in each direction. The second, Scenario B is most complex scenario, and considers wind speed varying between 8, 12 and 17m/s, with different probability of occurrence and directions, as per Figure 3. The wind farm optimization scheme carried out in this study follows the scheme found in Mosetti et al. [4] and Grady et al. [5], which considers the number of wind turbines for each scenario as Table 2. Table 3 present all algorithm's parameters applied in this work.

Table 1 – Wind Farm Characteristics

Parameter	Value/Unit
Hub height h	60 [m]
Rotor Diameter D_{wt}	40 [m]
Thrust coefficient C_t	0.88
Rated Wind Speed U_0	12 [m/s]
Maximum Wind Farm Site Area	4 [km ²]

Table 2 – Wind Farm Configurations

Scenario	Number of Wind Turbines
Scenario A	Mosetti et. al [4]
	19
Scenario B	Grady et. al [5]
	39
Scenario B	Mosetti et. al [4]
	15
Scenario B	Grady et. al [5]
	39

Table 3 – Parameters for Optimization Algorithms

Algorithm	Parameter	Value
PSO	C_1, C_2	1.42
	w_{max}	0.9
	w_{min}	0.4
	Population size	$2N_t$
	Iteration number	100
	Acceptance ratio	0.35

CA	α	0.3
	β	0.5
	Population size	$2N_t$
	Iteration number	100
ICA, aICA	Revolution Probability pR	0.05
	β	1.5
	α	1
	ζ	0.2
	Number of Empires $nEmp$	10
	Population size	$2N_t$
	Iteration number	100

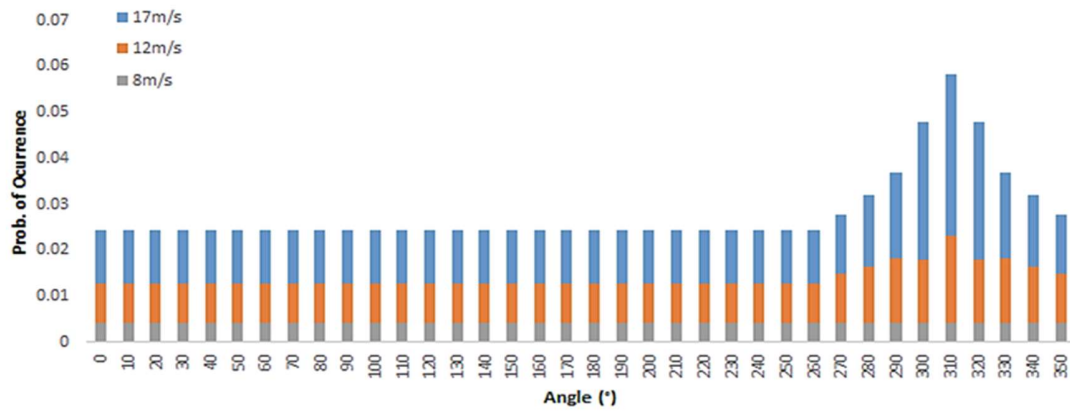


Figure 3 – Wind Speed Scenarios

$$P(U_i) = \begin{cases} 0 & \text{for } U_i \leq 2.3 \text{ m/s} \\ 0.3U_i^3 & \text{for } 2.3 < U_i \leq 12.8 \text{ m/s} \\ 630 & \text{for } 12.8 < U_i \leq 18 \text{ m/s} \\ 0 & \text{for } U_i > 18 \text{ m/s} \end{cases} \quad (16)$$

5 Results and Discussion

Tables 4 and 5 compare the results from proposed approach and those found in literature, which were conducted by Mosetti et al. [4] applying Genetic Algorithm (GA), and Grady et al. [5], which also applied GA. The adaptive approach increase the ICA ability of explore the search space, outperforming all four algorithms (PSO, CA, ICA and GA). Even considering the small number of iterations applied, the final cost and the Annual Energy Production (AEP) achieved by aICA outperform the canonical ICA, PSO and CA. Analyzing Figure 4 we can conclude that aICA ability of exploring the search space was improved, which leads outperform the canonical ICA algorithm.

The results show that the wind turbines were placed efficiently, decreasing the wake loss. Reducing the wake loss, means that the wind speed faced by the wind turbine is approximately equals to the wind speed into free stream, as counterpart the wind farm efficiency achieved (relation between the AEP with no loss and AEP experiencing wake loss) values closed to one.

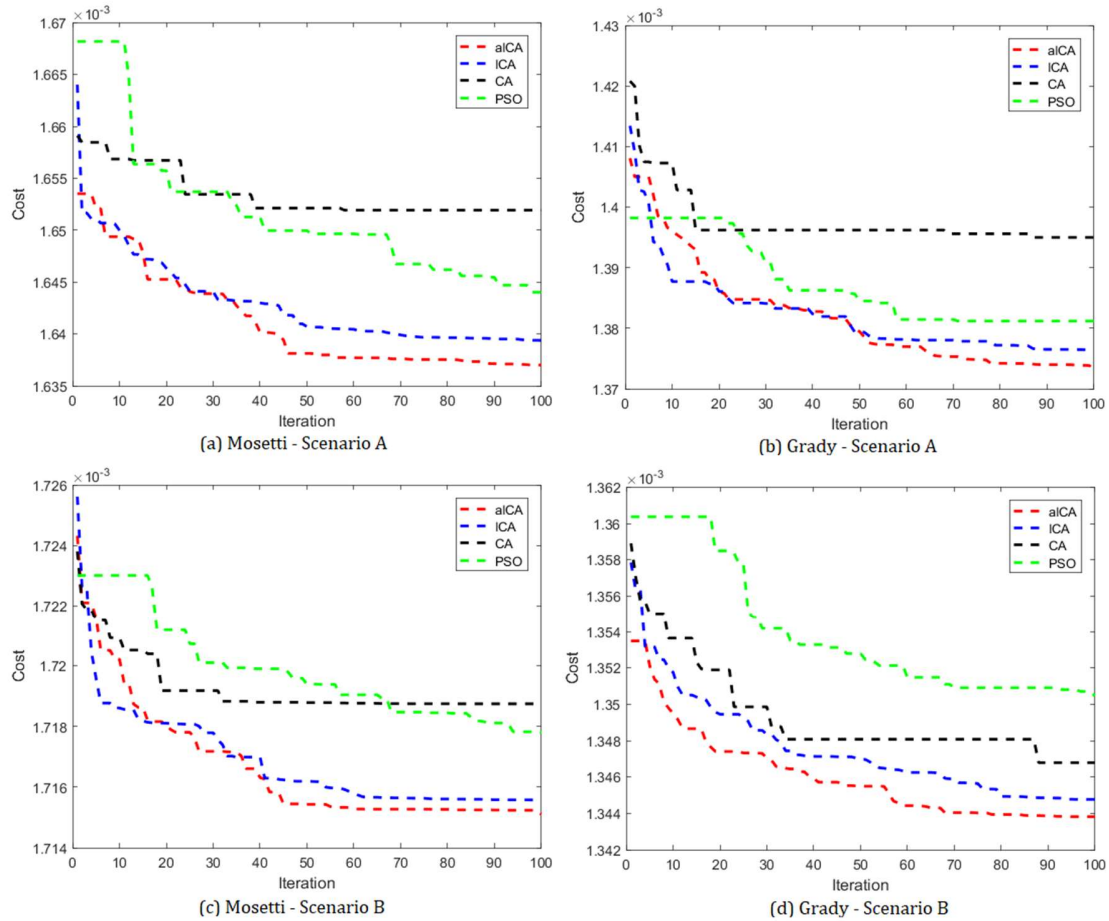


Figure 4 – Results comparison

Table 4 – Results Scenario A - Wind Speed at 12 m/s and variable direction

Approach	Efficiency	AEP (kWh year)	Cost ($\times 10^{-3}$)
Mosetti (GA)	0.9385	9,275	1.735
Present Study (aiICA)	0.9952	9,802	1.637
Present Study (ICA)	0.9937	9,787	1.639
Present Study (CA)	0.9862	9,713	1.651
Present Study (PSO)	0.9909	9,760	1.644
Grady (GA)	0.9217	18,639	1.444
Present Study (aiICA)	0.9695	19,601	1.373
Present Study (ICA)	0.9674	19,559	1.374

Present Study (CA)	0.9545	19,298	1.395
Present Study (PSO)	0.9641	19,492	1.381

Table 5 – Results Scenario B – Wind Speed at 8, 12 and 17 m/s - and variable direction

Approach	Efficiency	AEP (kWh year)	Cost ($\times 10^{-3}$)
Mosetti (GA)	0.9462	7,359	1.818
Present Study (aICA)	0.9992	7,801	1.715
Present Study (ICA)	0.9989	7,799	1.716
Present Study (CA)	0.9971	7,784	1.718
Present Study (PSO)	0.9976	7,789	1.717
Grady (GA)	0.9217	18,636	1.444
Present Study (aICA)	0.9869	20,034	1.343
Present Study (ICA)	0.9862	20,020	1.344
Present Study (CA)	0.9847	19,990	1.346
Present Study (PSO)	0.9820	19,935	1.350

6 Conclusion

In this work, we proposed an adaptive scheme of ICA parameters actualization aiming to optimize the wind farm layout. The proposed approach were compared with well-known optimization algorithms and the results confirm that the adaptive scheme increase the ability of exploration, avoiding get trapped into local minima, decreasing the final energy losses due to wake effect into wind turbines. The proposed approach can be applied as optimization tool during wind farm micro-siting process, especially due to the good results achieved with low computational cost. Future work will comprises applying aICA into WFLO considering multiobjective approach and wind turbines with different hub height.

Acknowledgements

The authors would like to thank National Council of Scientific and Technologic Development of Brazil - CNPq (Grant numbers: 303908/2015-7-PQ, 404659/2016-0-Univ, 204893/2017-8-PDE, and 150501/2017-0-PDJ) for its financial support of this work.

References

- [1] S. Shamshirband, D. Petković, R. Hashim, S. Motamedi, N. Badrul Anuar, *An appraisal of wind turbine wake models by adaptive neuro-fuzzy methodology*, International Journal of Electrical Power & Energy Systems, Volume 63, December 2014, Pages 618-624, ISSN 0142-0615.
- [2] S. Pookpant, W. Ongsakul, *Design of optimal wind farm configuration using a binary particle swarm optimization at Huasai district, Southern Thailand*, Energy Conversion and Management, Volume 108, 15 January 2016, Pages 160-180, ISSN 0196-8904.
- [3] S. A. Khan, S. Rehman, *Iterative non-deterministic algorithms in on-shore wind farm design: A brief survey*, Renewable and Sustainable Energy Reviews, Volume 19, March 2013, Pages 370-384, ISSN 1364-0321.

- [4] G. Mosetti, C. Poloni, B. Diviacco, *Optimization of wind turbine positioning in large windfarms by means of a genetic algorithm*, Journal of Wind Engineering and Industrial Aerodynamics, Volume 51, Issue 1, January 1994, Pages 105-116, ISSN 0167-6105.
- [5] S.A. Grady, M.Y. Hussaini, M.M. Abdullah, *Placement of wind turbines using genetic algorithms*, Renewable Energy, Volume 30, Issue 2, February 2005, Pages 259-270, ISSN 0960-1481.
- [6] G. Marmidis, S. Lazarou, E. Pyrgioti, *Optimal placement of wind turbines in a wind park using Monte Carlo simulation*, Renewable Energy, Volume 33, Issue 7, July 2008, Pages 1455-1460, ISSN 0960-1481.
- [7] S. Rajper, I. J. Amin, *Optimization of wind turbine micro siting: A comparative study*, Renewable and Sustainable Energy Reviews, Volume 16, Issue 8, October 2012, Pages 5485-5492, ISSN 1364-0321.
- [8] V. Nikolić, S. Shamshirband, D. Petković, K. Mohammadi, Ž. Čojbašić, T.A. Altameem, A. Gani, *Wind wake influence estimation on energy production of wind farm by adaptive neuro-fuzzy methodology*, Energy, Volume 80, 1 February 2015, Pages 361-372, ISSN 0360-5442.
- [9] S. Shamshirband, D. Petković, N. B. Anuar, A. Gani, *Adaptive neuro-fuzzy generalization of wind turbine wake added turbulence models*, Renewable and Sustainable Energy Reviews, Volume 36, August 2014, Pages 270-276, ISSN 1364-0321.
- [10] E. Atashpaz-Gargari and C. Lucas, *Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition*, 2007 IEEE Congress on Evolutionary Computation, Singapore, 2007, pp. 4661-4667. doi: 10.1109/CEC.2007.4425083
- [11] Jensen NO. *A note on wind generator interaction*. Technical report Riso-M-2411; 1983.
- [12] R. Shakoor, M. Y. Hassan, A. Raheem, Y.-Kang Wu, *Wake effect modeling: A review of wind farm layout optimization using Jensen's model*, Renewable and Sustainable Energy Reviews, Volume 58, May 2016, Pages 1048-1059, ISSN 1364-0321.
- [13] J. Kennedy, R. Eberhart. *Particle swarm optimization*. In: IEEE int conference on neural networks; 1995.
- [14] R. G. Reynolds, *An Introduction to Cultural Algorithms*, in Proceedings of the 3rd Annual Conference on Evolutionary Programming, World Scientific Publishing, pp 131–139, 1994.
- [15] Yarpiz, The Yarpiz Project: Machine Learning, Metaheuristics, Multiobjective Optimization and Fuzzy System: available in <http://yarpiz.com>
- [16] M. Abdollahi, A. Isazadeh, D. Abdollahi, *Imperialist competitive algorithm for solving systems of nonlinear equations*, Computers & Mathematics with Applications, Volume 65, Issue 12, 2013, Pages 1894-1908, ISSN 0898-1221.
- [17] K. Kiamehr, S. K. Hannani, *Wind farm layout optimization using imperialist competitive algorithm*, Journal of Renewable and Sustainable Energy 6, 043109 (2014).