International Journal of Combinatorial Optimization Problems and Informatics, 11(3), Sep-Dec 2020, 97-111. ISSN: 2007-1558.

www.editada.org

Analysis Optimization of wind farm turbines layout using an evolutionary algorithm

Abelardo Buentello Duque, Salvador Hernández González, José A. Jiménez García, Vicente Figueroa Fernández, Moisés Tapia Esquivias
Tecnológico Nacional de México en Celaya, Department of Industrial Engineering; Av. García Cubas 1200, Esquina Ignacio Borunda, Celaya, Gto. México. Tel. 01 (461)61 17575.
m1703073@itcelaya.edu.mx, salvador.hernandez@itcelaya.edu.mx, alfredo.jimenez@itcelaya.edu.mx vicente.figueroa@itcelaya.edu.mx, moises.tapia@itcelaya.edu.mx

Abstract. Wind energy is one of the most promising renewable energies in the coming years. The construction of wind farms is destined to increase in the U.S. and Mexico, only Mexico has a wind potential of 19,805 GWh. An important stage in the construction and design of wind farms is to solve the Wind Farm Layout Optimization Problem (WFLOP). The design of the wind farm involves many factors. One of the non-linear factors is the wake effect, which reduces the energy produced by wind turbines. To maximize the wind energy capture, in this paper an evolutionary algorithm is implemented in such a way that the placement of wind turbines within wind farm be optimized taking account the individual energy loss due to wake effects. The evolutionary algorithm is implemented in several instances. The results obtained by the evolutionary algorithm are compared with the results of the Generalized Reduced Gradient (GRG) algorithm.
Keywords: Artificial Intelligence, Evolutionary Algorithm, Meta-heuristic, Combinatorial Optimization Problem, Wind Farm, Wind Energy.

Article Info
Received Jul 10, 2019
Accepted Jan 11, 2020

1 Introduction

In recent years, one of the most challenging topics among the different kinds of renewable energy is wind energy. It seems that humans realized the benefit of the wind in 200 B.C., when Chinese first invented a windmill. The wind is used in a process to generate useful kind of energies like mechanical and electrical energies [1]. A wind turbine is a device which converts the wind's energy into electrical energy. This is achieved by blades, which are attached to a hub that rotates in response to the aerodynamic force of the wind on the blades. This rotation drives a generator which produces electricity that is transferred to the electrical power grid. A wind farm is a group of collocated wind turbines and may be thought of as wind-driven power stations [2]. One of the main advantages that make the energy produced in the wind farms is more competitive is that fixed costs (administration costs, costs related to the electricity network and project development costs) are distributed throughout the process investment. Wind energy offers other advantages compared to other types of renewable energy; higher conversion rate, clean and safety [3]. Wind energy is an alternative energy type to fossil fuel and has no negative effects on nature like fossils, which are the main sources of unsustainable energy and will be exhausted in the foreseeable future due to limited resources, rapid consumption, climate change, global warming, etc. Since wind is a sustainable energy source, wind energy has become widespread during the last 20 years [1,4]. Although wind energy also has several drawbacks, it seems that the positive aspects of this renewable energy are more attractive for the worldwide industry. However, one of the problems that must be faced in order to obtain the most energy from a wind farm is the Wind Farm Layout Optimization Problem (WFLOP). WFLOP is a problem, which has an objective function that tries to minimize the wake effects of turbines by each other.

The wake effect is the interference phenomenon for which, if two turbines are located one close to another, the upwind one creates a shadow on the one behind. This is of great importance in the design of the layout since it results in a loss of power production for the turbine downstream, that is also subject to possibly strong turbulence [5].
Therefore, the wind farm layout optimization problem consists in finding an optimal allocation of turbines in a given site (wind farm layout) that maximizes the expected power production. Finding high-quality solutions may ultimately lead to high profits for wind farm developers [6].

This strategic problem is extremely hard in practice, both for the sizes of the instances in real applications and for the presence of several nonlinearities to be taken into account, such as the wake effect [5]. In large wind farms wake results lead to considerable power loss [7], and thus is desirable to minimize them in order to maximize the expected power output [6].
It is estimated in [8] that in large offshore wind farms, the average power loss due to turbine wakes is around 10-20% of the total energy production. It is then evident that power production can increase significantly if the farm layout is designed so as to reduce the effect of turbine wake as much as possible [5].

Different models have been developed to calculate the loss of energy that near and far wake effect cause in wind farms. In wind farm layout optimization problem, the distant wake effect is more important than near wake effect. Wake models have been the subject of several researches and have been compared. The comparison of different far wake models shows that the Jensen´s far wake model is an excellent choice to solve the wind farm layout optimization problem due to its simplicity and a relatively high degree of accuracy [9]. Though the model is an approximation of the real context, it turns out to be accurate enough for turbines layout optimization in a wind farm [5].

In this paper, Jensen´s model is considered.

The wind farm layout optimization problem has been receiving increasing attention from the scientific community as this problem is classified as NP-Hard, which indicates that there is no algorithm that can solve it in a polynomial or reasonable computational time. Due to excessive computation time, the exact algorithms would flop. Because of the intricacy of layout optimization of wind farm, rigorous optimization approaches such as branch and bound, dynamic programming, backtracking and linear programming, etc., can be utilized to some extent [10].

There are many researches that show that using heuristics and meta-heuristics, it is possible to obtain wind farm layouts that provide a high-quality solution (energy) in a reasonable time. According to the literature, some of the algorithms or optimization techniques most commonly used to solve the WFLOP are; Genetic Algorithms (GA) [11,12],Simulated Annealing [13,14], Ant Colony Optimization (ACO) [15,16] and Particle Swarm Optimization (PSO) [17,18].

In the present work, the authors use the evolutionary algorithm (GA) of Excel Solver to optimize the layout of a given wind farm considering the Jensen´s model for computing the power loss between turbines due to wake effects. The results of the instances resolved through GA are compared with GRG (Generalized Reduced Gradient) algorithm. GRG is one of the more robust “classical” algorithms for solving non-linear problems [19]. The main criterion of comparison is the computational effort that each of the algorithms invested in solving a particular instance. The comparison between a meta-heuristic method and a classical algorithm as techniques to solve WLOP is something that has not been reported in the literature.

2 Experimental procedures

In the first part of this section, the technique of discretization to solve WFLOP is introduced. The details of the wake model used in this research are described in the second part. The GA method used in this paper is discussed in the third part of this section.

2.1 Discretization of wind farm

The technique of discretization is one of the most used strategies to solve WFLOP. Due to the complexity of the problem, especially when contemplating scenarios or instances of considerable size, discretization is an efficient way to obtain a high-quality design solution in reasonable computing times. The discretization consists of dividing a wind farm into small square or rectangular elements. Figure 1 shows a discrete wind farm with a total area of 0.454 km^2 and 36 grids. Each grid measures 90 m x 90 m (resolution). The centroid or center point within each grid represents the possible location where a turbine could be assigned or installed. If 12 wind turbines were installed in the wind farm shown in figure 1 with 36 possible location, the number of possible combinations for the wind farm layout would be 1,251,677,700. Discretization is very useful as, if it not discretized, the algorithm used would take a very long time to find a solution within a continuous solution space [11]. The large number of possible layouts of wind farms is the reason why the WFLOP is categorized as NP-Hard. Flat, rectangular and square wind farms are only considered in this paper.

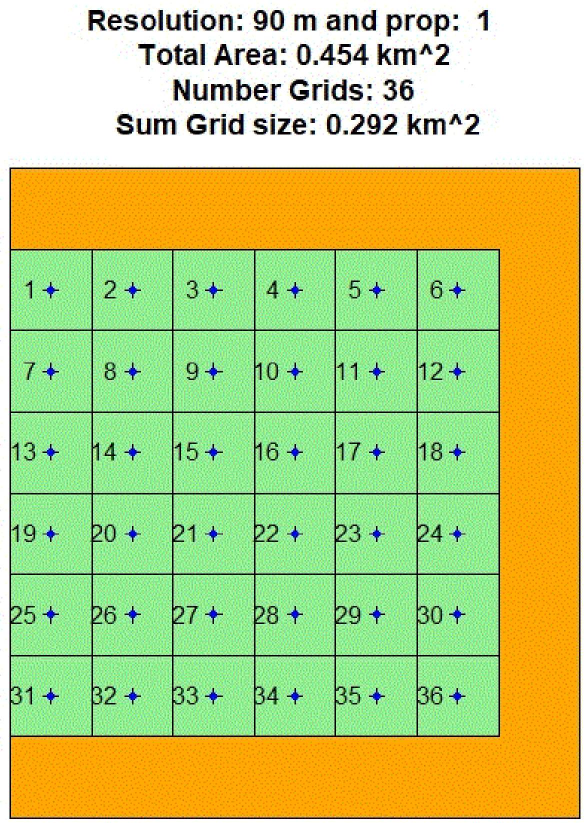


Fig. 1. Wind farm discretized

2.2 Wake effect

The wind efficiency of a turbine tends to be reduced after putting it in a wind farm with other turbines due to wake effects [20]. When the wind passes through the rotor of the wind turbine, a reduction in speed occurs behind the rotor due to turbine blades absorb kinetic energy from the ambient wind speed. It means the downwind air of the wind turbine has lower wind speed and higher turbulence, therefore, for downstream wind turbine energy production is not the same with the upstream wind turbine since there is a deficit in wind speed [21]. Wake effects refer to this wind speed reduction and diminish in energy production in a wind farm based on interactions between wind turbines [21]. Jensen's model proposed in [6] is implemented in this paper in order to calculate the power loss between turbines due to wake effects. Jensen´s model proposed in [6] is coherent to the ones used in [22-26], and it is equivalent to the one proposed by [27]. One of the main advantages of this model is the possibility to implicitly deal with a large number of wind scenarios, which is a must in practical cases. On the contrary, most of the alternative models turn out to be impractical, as they require the definition of a large number of additional variables and constraints [5]. To explain the model, Figure 2 is presented.

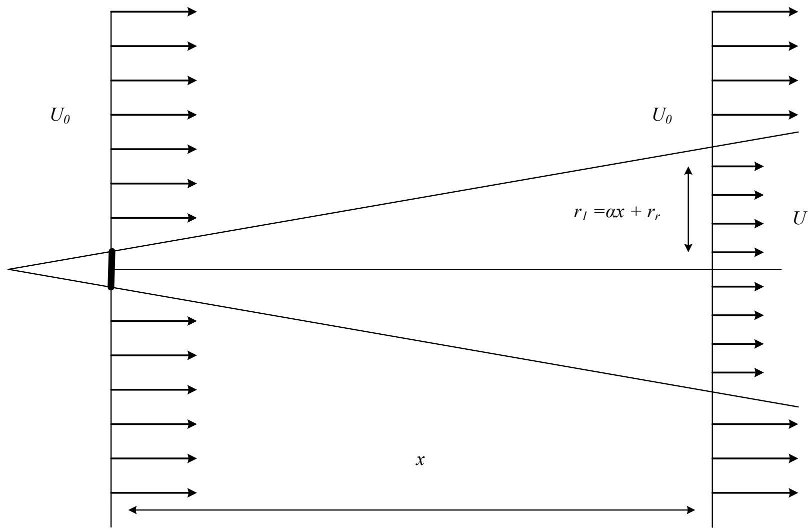


Fig. 2. Representation of the wake effect

Figure 2 shows wind from left to right at a certain speed U_0 and interact with a wind turbine (represented as a vertical line in bold on the left) whose rotor radius is r_r. At a particular distance x downwind, the wind speed is U and the radius of the wake (initially equal to r_r) becomes $r_1 = \alpha x + r_r$. As described in [6], the a-dimensional scalar α determines how quickly the wake expands with distance and it is defined in (1), where z is the hub height of the wind turbine producing the wake effect and z_0 is a constant, known as surface roughness, which depends on the surface characteristics.

$$\alpha = \frac{0.5}{\ln \frac{z}{z_0}} \tag{1}$$

Let be i the position of the wind turbine that generates the wake effect, j the position affected by position i, u_0 the ambient wind speed (without turbulence), and u_j the wind speed at j. Then:

$$u_j = u_0(1 - vd_{ij}) \tag{2}$$

where vd_{ij} is the velocity deficit that is induced at position j by the wake effect created by i. vd_{ij} is calculated by the following expression:

$$vd_{ij} = \frac{2a}{1 + \alpha\left(\frac{x_{ij}}{r_d}\right)^2} \tag{3}$$

The term x_{ij} that appears in the denominator is the distance between positions i and j. The term a that appears in the numerator is known as the axial induction factor and is computed as follows:

$$a = 0.5\left(1 - \sqrt{1 - C_T}\right) \tag{4}$$

In Eq.(4) the term C_T is the constant thrust coefficient, which estimates the proportion of energy captured when the wind goes through the blades of the wind turbine. Likewise, the term r_d that appears in the denominator in Eq. (3) is called downstream rotor radius and is calculated as follows:

$$r_d = r_r \sqrt{\frac{1-a}{1-2a}} \tag{5}$$

Due to many turbines are installed on a wind farm, wake effects can intersect and accumulate. These accumulations of wake effect might affect one or more downstream wind turbines at the same time. In the Jensen model, the total velocity deficit vd_{ij} at a position j that is affected by more wakes is gotten as follows:

$$v_{def(j)} = \sqrt{\sum_{i \in W(j)} vd_{ij}^{\;2}} \tag{6}$$

where $W(j)$ represents the set of wind turbines affecting position j with a wake effect. $v_{def(j)}$ is then substituted in Eq. (2) in place of vd_{ij} to compute u_j.

2.3 Genetic Algorithms

One of the most popular types of evolutionary algorithms are Genetic Algorithms (GA). Genetic algorithms are techniques used in combinatorial optimization, search and automatic learning [28]**.** These algorithms are robust search techniques that try to find the minimum or the maximum of a function based on principles inspired by the natural genetic and evolution mechanisms observed in nature [28,29]. GA use multiple paths of search instead of a single point, using encoded solutions to the problem. The main principle of GA is the maintenance of a set of encoded solutions (population) that evolves along with the generations, guiding the population towards the best solution [30]. To guide the population towards the best solution, GA uses three genetic operators: selection, crossover and mutation. As a global search tool, GA may escape from local optima by randomly generating solutions. An evolutionary algorithm for optimization is different from "classical" optimization methods in several ways [31]. In this paper, the authors use the evolutionary algorithm (GA) of Excel Solver (Microsoft Excel 2013) in order to optimize wind turbines placement of a given wind farm. The pseudocode of the Genetic Algorithm is presented in Table 1. The objective function used in Excel is to maximize the total energy output of a wind farm as shown in Eq. (7). Eq. (8) shows the constraint.

Table 1. Pseudocode of Genetic Algorithm

Genetic Algorithm
1: *t* ← 0 % Iteration counter %
2: initialize (P) % Initialize the population %
3: **while** *there is no stopping criterion* (*t, P)* **do**
4: *Parents ← selection (P)* % Select parents %
5: *Children ← reproduction (Parents)* % Crossover %
6: *mutation(Children)* % Mutate the children %
7: *evaluate(Children)* % Evaluate the children %
8: *newGeneration = replacement (P, Children)* % replaces the population with the current %
9: *t* ← *t* + 1 % One more iteration %
10: **end while**
11: **Return**: best solution found.

$$Max\, Z = \sum_{i=1}^{N} P_i \, x_i \tag{7}$$

s.t

$$\sum_{i=1}^{N} x_i = \{0,1\} \qquad (8)$$

where; Z denotes the total energy output of the wind farm. P_i represents the energy output of a wind turbine. N indicates the number of wind turbines to be installed on the wind farm. In Excel, x_i corresponds to the binary decision variable which is defined in each of the cells, each Excel cell in turn indicates a possible location and this set of cells make up the wind farm. x_i is then used to show whether there is a wind turbine at location "i"or not. If x_i takes the value of 0 represents that there is no wind turbine installed in location"i". Value of 1 represents otherwise.

2 Results

During the optimization processes only one type of wind turbine was used. The turbine used has a hub height z = 60 m, diameter D = 40 m, and a constant thrust coefficient C_T = 0.88. The power curve of the turbine considered is shown in Figure 3. For the optimization of each instance only one direction (North-South) and wind speed (12.8 m/s) were considered. The surface roughness for all instances is z_0 = 0.3 m. Table 2 shows the results of the resolved wind farm instances by GA with 3 replicas. The first instance corresponds to the test instance. In this research, it was decided to replicate three times the optimization of each of the instance with GA in order to better estimate the computational effort (run time). Table 2 also shows a comparison among the solutions generated by GA and those obtained by the GRG algorithm. The GRG algorithm was configured to find the optimal global solution in each instance. In the case of optimization by the GRG algorithm, only one replica was enough to determine the computational effort trend as larger instances were optimized. Table 3 exhibits the parameters values in which the algorithms were adjusted for the optimization of all instances. Note that some parameters were adjusted according to the default values of Excel Solver 2013.

The hardware specifications of the computer where all the optimization processes were carried out are the following: Intel i5-7200U CPU @ 2.50 GHz, 2701 MHz, 2 main processors, 4 logical processors and installed physical memory (RAM) of 8GB.

Table 2. Results of resolved instances

Instance	Number of grids	Distance btw grids (meters)	Number of installed turbines	Best objective values by GA and run times (kWh, seconds)			Best objective value by GRG and run time (kWh, seconds)
				Replica 1	Replica 2	Replica 3	
1	3*4	500	8	4992.88,22.125	4992.88,20.48	4992.88,19.92	4992.88,48.656
2	4*4	200	8	4641.48,36.578	4641.48,37.46	4641.48,38.23	4641.48,164.328
3	5*5	200	15	7523.41,45.25	7523.41,50.34	7523.41,52.45	7523.41,443.062
4	6*6	200	12	7298.30,36.219	7298.30,35.96	7219.01,36.12	7318.26,2780.75
5	6*10	200	20	12018.56,33.875	11879.93,35.12	11979.18,36.17	12197.10,8910.54
6	7*7	200	21	11533.05,34.94	11578.07,36.06	11488.28,36.31	11818.92,18065.11
7	7*10	200	30	16229.16,44.609	16379.59,36.29	16204.03,36.53	16884.17,60268.625

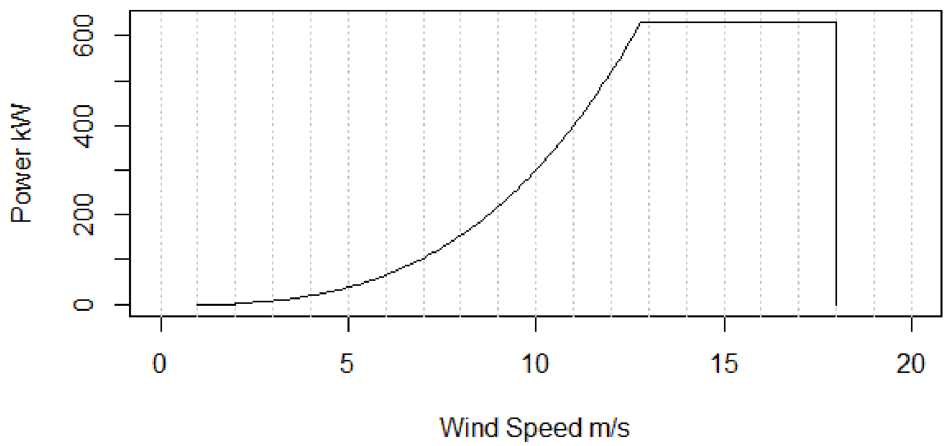


Fig. 3. Power curve for the wind turbine considered

Table 3. Parameter values used in Excel Solver

Parameters/Method	GA	GRG Non-linear
Convergence	0.0000001	0.000001
Integer optimality (%)	0	0
Maximum time (seconds)	Unlimited	Unlimited
Iterations	Unlimited	Unlimited
Accuracy of restrictions	0.000001	0.000001
Maximum of subproblems	Unlimited	Unlimited
Maximum viable solutions	Unlimited	Unlimited
Derivative method	-	Central
Multiple start	-	True
Mutation rate	0.075	-
Population size	100	100
Random initialization value	0	0
Maximum time without improvement (seconds)	30	-

Figure 4 shows the run times that GA invested in solving each of the instances proposed in this paper. The optimization process was replicated three times for each instance using GA. Likewise, Figure 4 indicates the mean run time obtained from the 3 replicas. The trend presented by this graph shows that although the number of variables increases (more turbines, more number of possible locations; larger instances) the algorithm is able to find good quality solutions in reasonable computing times.

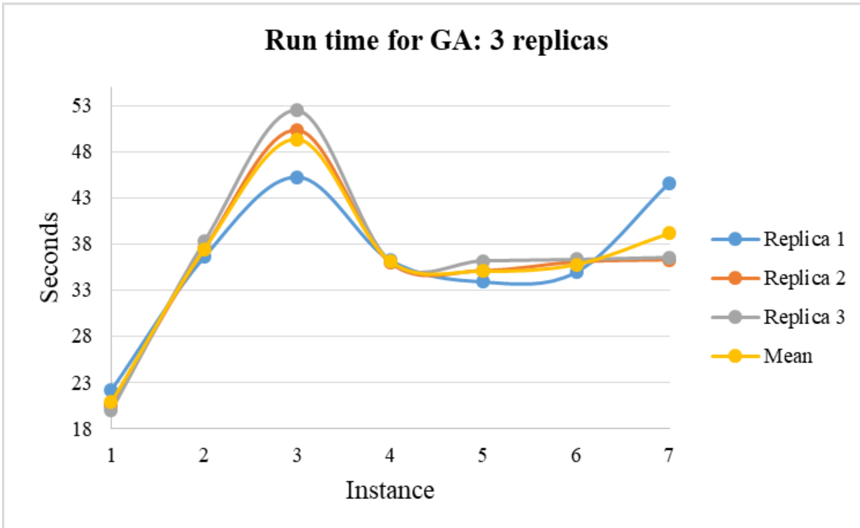


Fig. 4. Run time with 3 replicas and mean run time for every instance

Figure 5 shows the run times that the GRG algorithm invested in solving each of the instances. In comparison to the results obtained by GA, the run times obtained by the GRG algorithm increase as the size of the instance grows. Therefore, the figure indicates exponential growth as more variables are added to the problem. This non-linear trend of computational effort is typical when classical "exact" algorithms for solving NP-Hard problems are implemented.

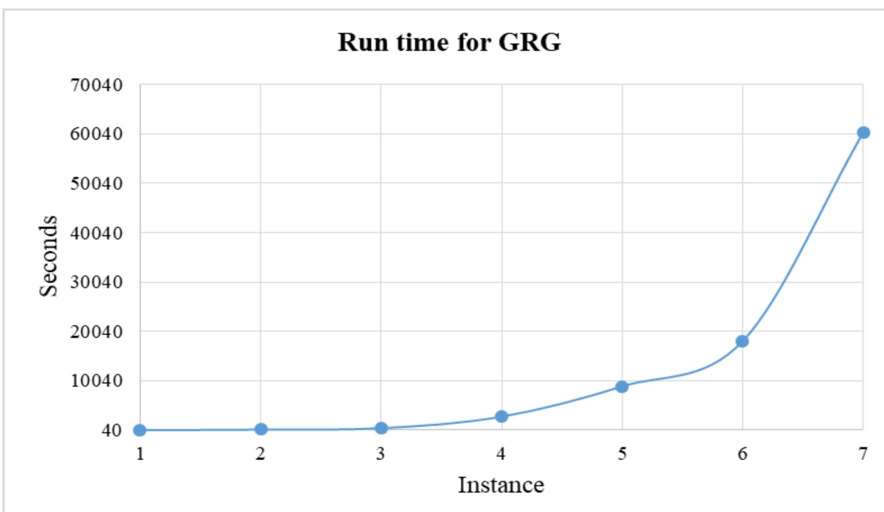


Fig. 5. The run time that GRG algorithm invested in solving each instance

The solutions reported in Table 2 are presented schematically below. The figures that schematically show the layout solutions obtained by GA correspond to the first replica.

The layout solution found for the first instance (12 possible locations and 8 wind turbines) is shown in Figure 6. This optimal global solution was found both by GA and by the GRG algorithm. Wind turbines installed in the center of the grids are represented by filled black points. The value that is positioned above the filled black point represents the amount of energy (kW) produced by said turbine installed in that location. It is observed that the wind turbines downstream produce less energy due to the wake effects generated by the turbines upstream. The energy solution found by both algorithms was 4992.88 kW with an efficiency of 99.20%. The computational time that the GRG algorithm invested in finding this solution was 48.656 seconds while GA invested 22.125 seconds.

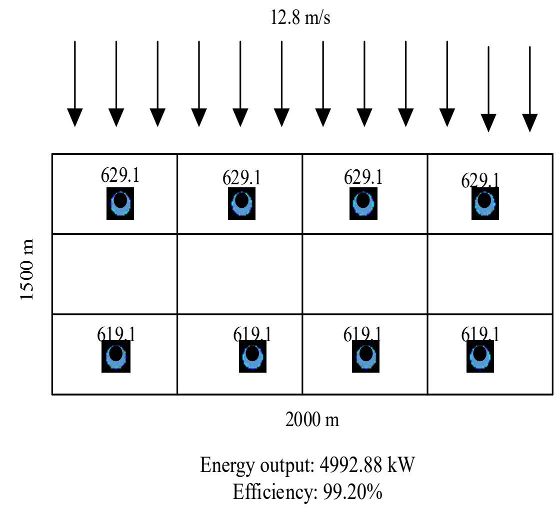


Fig. 6. Optimal layout generated by both GA and the GRG algorithm for instance #1

Figure 7 shows the optimal placement of wind turbines which was found by both the GRG algorithm and the Genetic Algorithm. This solution corresponds to instance #2 (16 possible locations and 8 wind turbines). Likewise, the figure indicates the total energy output and the efficiency of the wind farm. The GRG algorithm invested 164.328 seconds while GA invested 36.578 seconds.

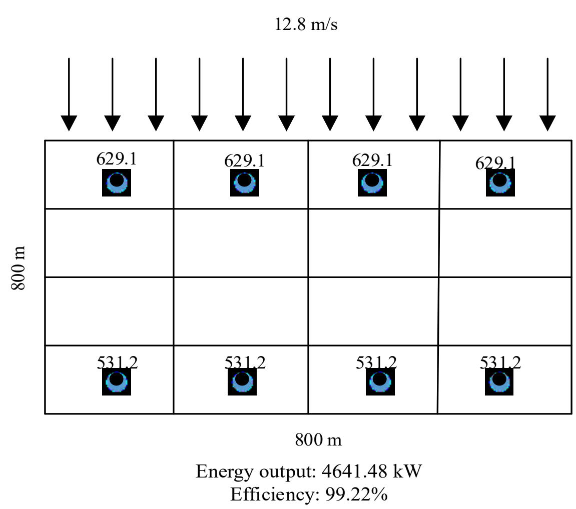


Fig. 7. Optimal layout generated by both GA and the GRG algorithm for instance #2

Figure 8 shows the optimal layout generated by both GA and the GRG algorithm for instance #3 (25 possible locations and 15 wind turbines). Likewise, figure 8 indicates the total energy output and the efficiency of the wind farm. To find this solution, the GRG algorithm invested 443.062 seconds while GA invested 45.25 seconds.

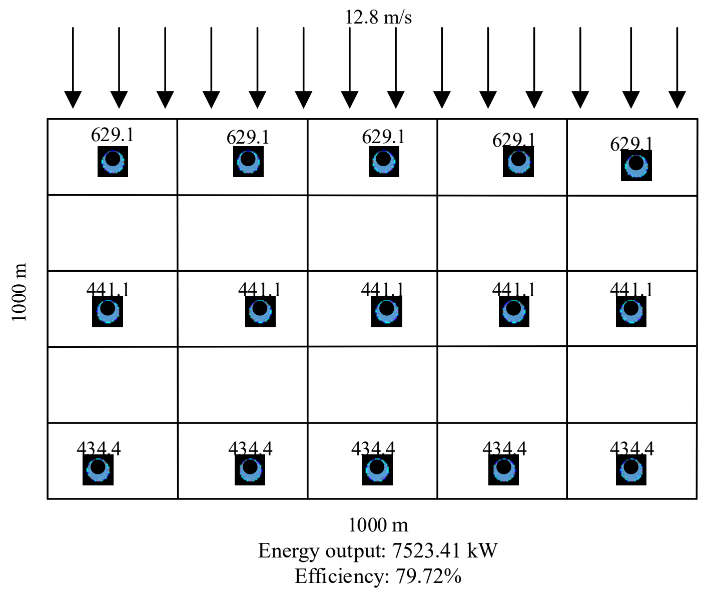


Fig. 8. Optimal layout generated by both GA and the GRG algorithm for instance #3

Figure 9 shows the optimal layout generated by the GRG algorithm for instance #4 (36 possible locations and 12 wind turbines). Likewise, figure 9 shows the total energy output and the efficiency of the wind farm with this configuration. To find this solution, the GRG algorithm invested 2780.75 seconds.

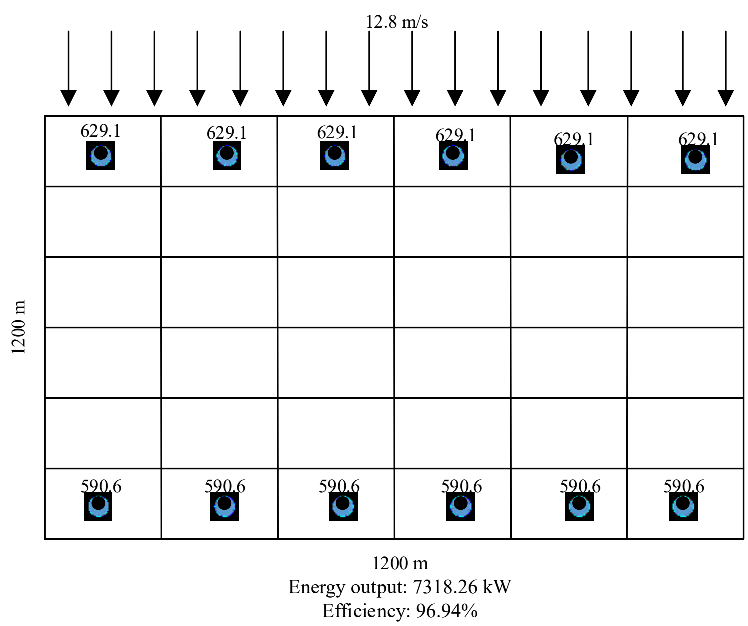


Fig. 9. Optimal layout generated by the GRG algorithm for instance #4

Figure 10 shows the best distribution solution found by GA for instance #4. Also, figure 10 exhibits the total energy output and the efficiency of the wind farm with this layout. To find this solution, the Genetic Algorithm invested 36.219 seconds. The difference in efficiency among the solution found by the GRG algorithm and the solution generated by GA for instance #4 is 0.34% equivalent to 19.93 kW.

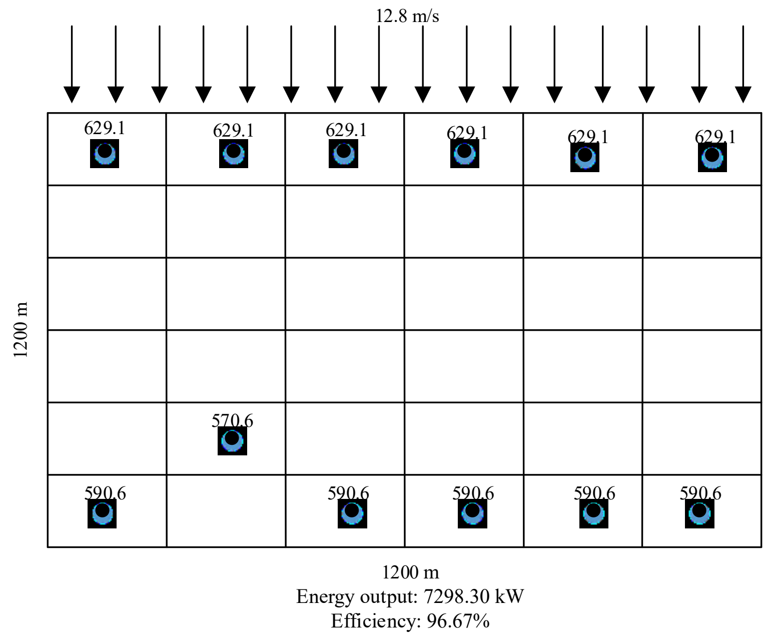


Fig. 10. Best distribution solution found by GA for instance #4

Figure 11 shows the optimal layout generated by the GRG algorithm for instance #5 (60 possible locations and 20 wind turbines). Likewise, figure 11 indicates the total energy output and the efficiency of the wind farm with this configuration. To find this solution, the GRG algorithm invested 8910.54 seconds.

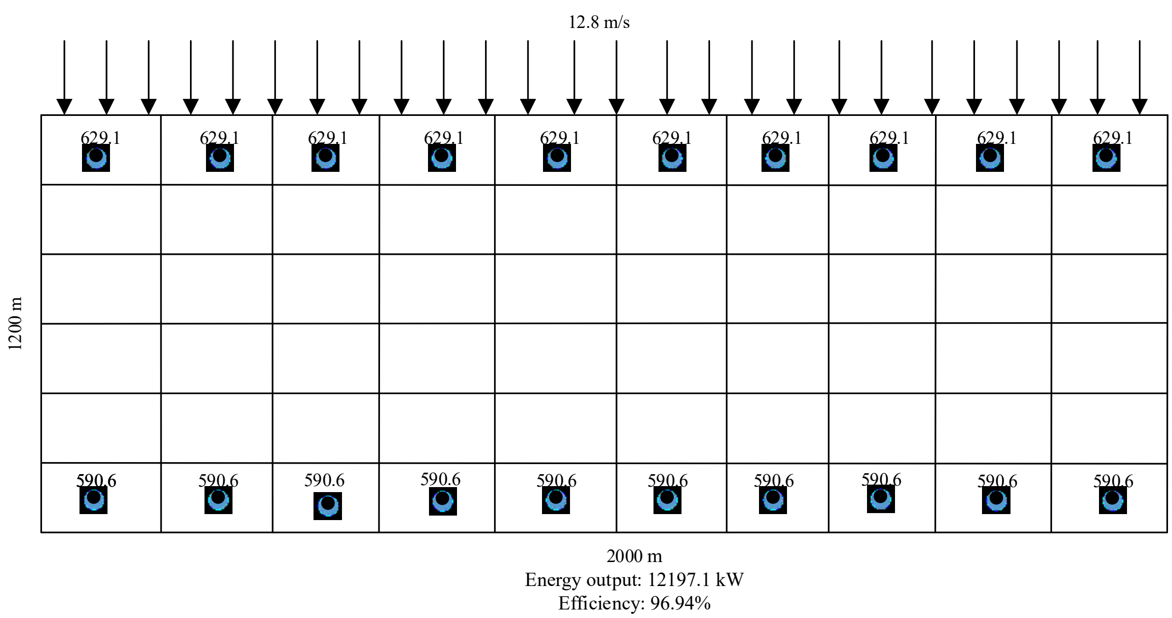


Fig. 11. Optimal layout generated by the GRG algorithm for instance #5

Figure 12 shows the best distribution solution found by GA for instance #5. Also, figure 12 shows the total energy output and the efficiency of the wind farm with this distribution. To find this solution, the Genetic Algorithm invested 33.875 seconds. The difference in efficiency between the solution generated by the GRG algorithm and the solution found by GA for instance #5 is 1.42% equivalent to 178.6 kW.

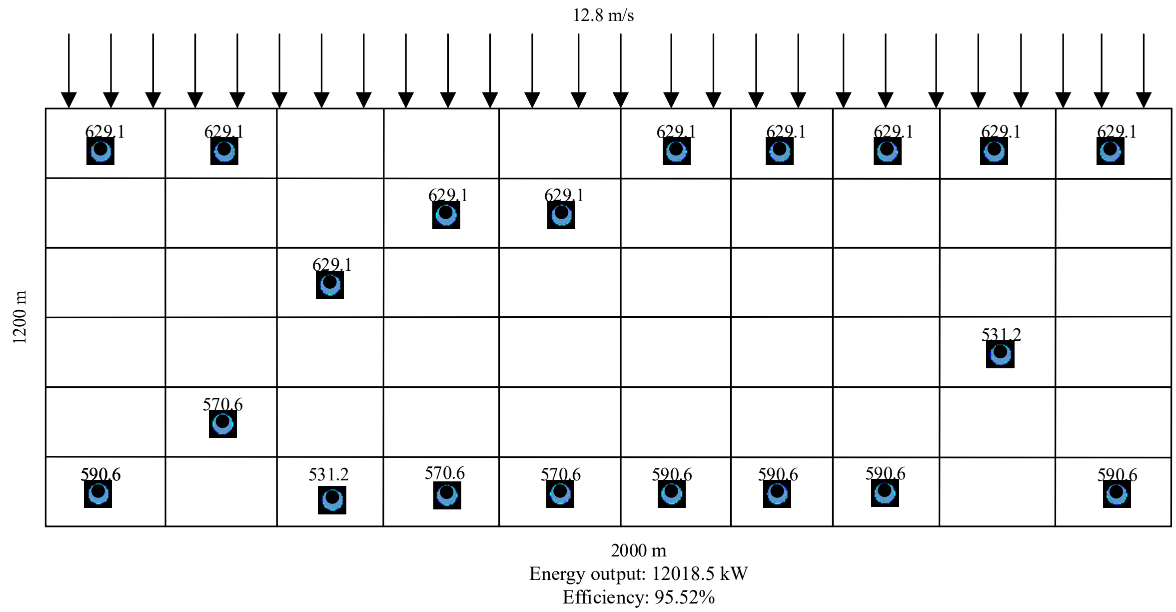


Fig. 12. Best layout solution generated by GA for instance #5

Figure 13 shows the optimal placement of wind turbines which was generated by the GRG algorithm for instance #6 (49 possible locations and 21 wind turbines). Likewise, figure 13 indicates the total energy output and the efficiency of the wind farm with this distribution of wind turbines. To find this solution, the GRG algorithm invested 18065.11 seconds.

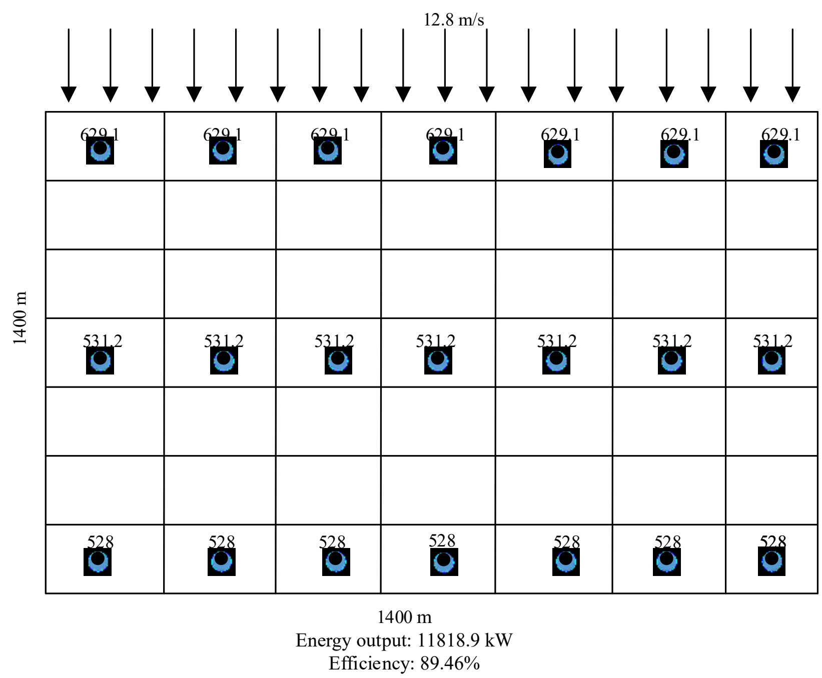


Fig. 13. Optimal layout generated by the GRG algorithm for instance #6

Figure 14 shows the best distribution solution generated by GA for instance #6. Likewise, figure 14 indicates the total energy output and the efficiency of the wind farm with this layout. To find this solution, the Genetic Algorithm invested 34.94 seconds. The difference in efficiency among the solution found by the GRG algorithm and the solution generated by GA for instance #6 is 2.17% equivalent to 285.9 kW.

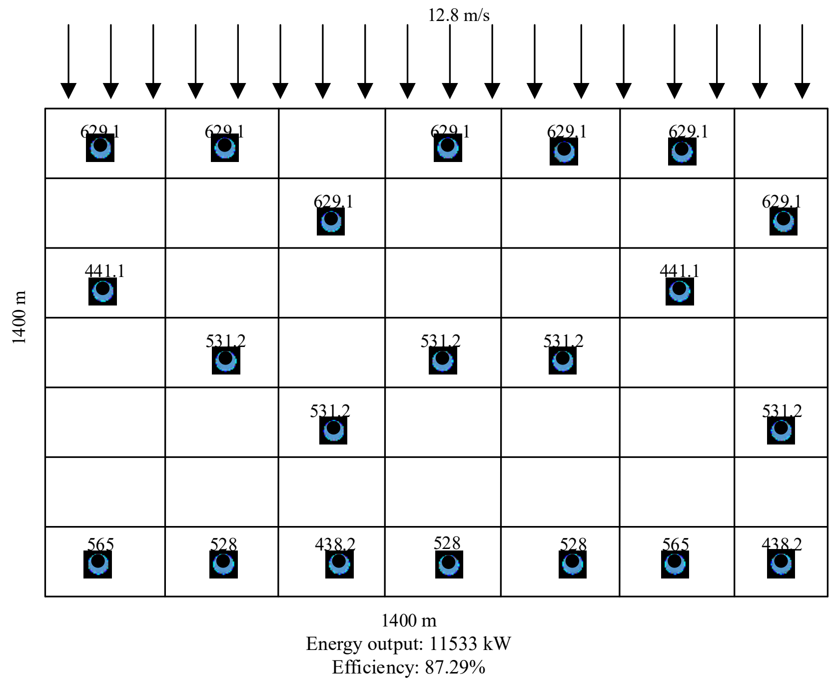


Fig. 14. Best distribution solution found by GA for instance #6

The optimal layout solution generated by GRG for instance #7 (70 possible locations and 30 wind turbines) is shown in Figure 15. Likewise, figure 15 indicates the total energy output and the efficiency of the wind farm with this configuration of wind turbines. To find this solution, the GRG algorithm invested 60268.625 seconds.

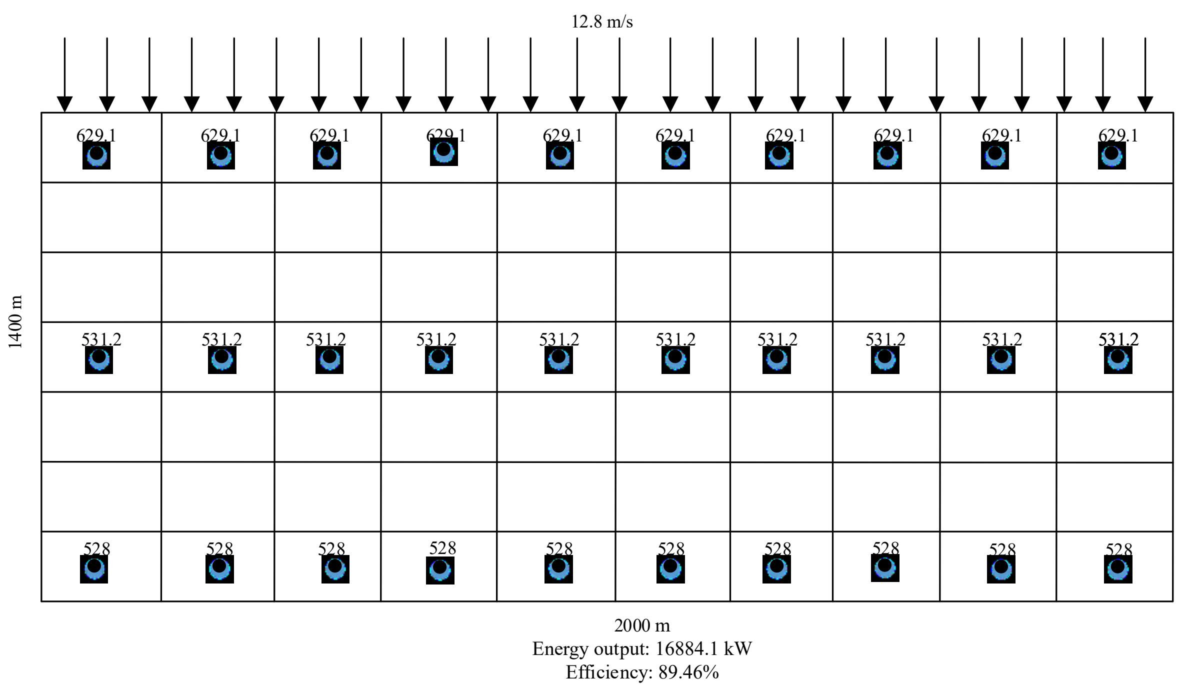
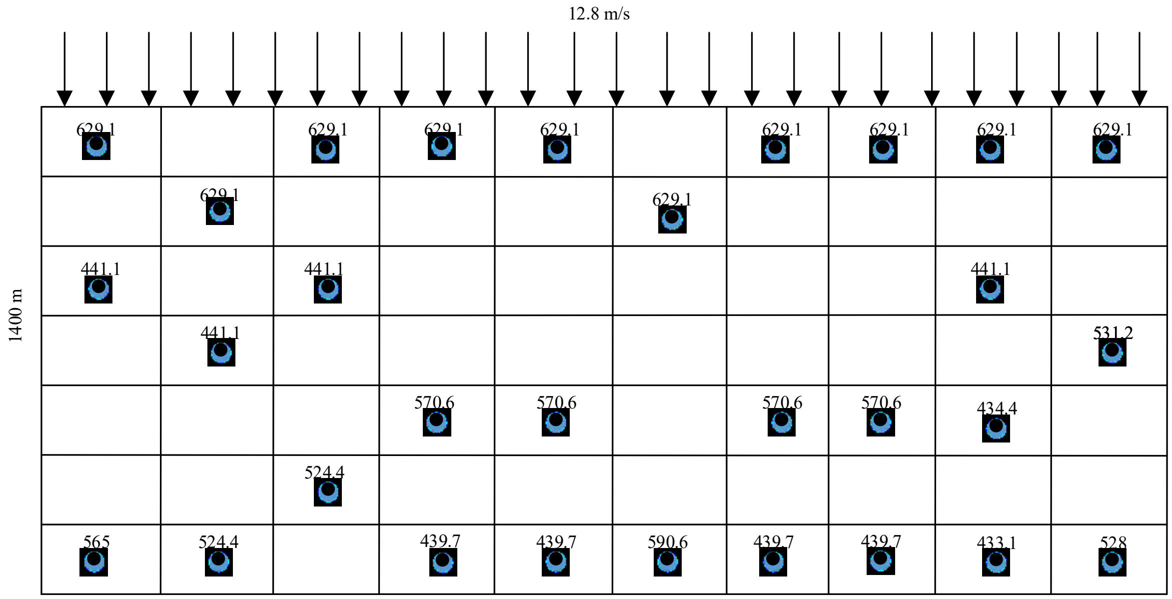


Fig. 15. Optimal configuration generated by the GRG algorithm for instance #7

The best layout solution generated by GA for instance #7 is shown in Figure 16. Likewise, figure 16 exhibits the total energy output and the efficiency of the wind farm with this layout. To find this solution, the Genetic Algorithm invested 44.609 seconds. The difference in efficiency between the solution generated by the GRG algorithm and the solution found by GA for instance #7 is 3.47% equivalent to 655 kW.



2000 m
Energy output: 16229.1 kW
Efficiency: 85.99%

Fig. 16. Best layout solution found by GA for instance #7

4 Conclusions

The complexity of the wind farm layout optimization problem has been discussed and an evolutionary algorithm to optimize proposed scenarios of wind farms has been implemented. The computational results obtained by GA have been compared with the results obtained by the GRG algorithm. From this comparison, it is concluded that for some instances both the genetic algorithm and the classical algorithm generated the same solutions; however, solving times for GRG algorithm are much larger than GA. As more variables were added to the problem, the classical algorithm spent more computation time while the meta-heuristic method invested a reasonable computation time. This type of comparison, in which a classical algorithm and a meta-heuristic are used to solve the WFLOP has not been found in the literature. This is very important for the wind farm developers since a developer needs to decide if it is worth spending more time looking for the optimal layout or only a high-quality solution is enough. Undoubtedly, the decision made by the developer will mainly affect economic aspects; production cost, profit, investment, etc.

In this paper, the replication allowed to estimate better the computational effort (run time) that GA invested in solving each one of the instances as the evolutionary algorithm required a reasonable computational time to solve larger instances. Likewise, the 3 replicas allowed to analyze and determine the capacity (robustness) of the algorithm to find a higher quality energy solution when running a new optimization. Therefore, the effectiveness and performance of genetic algorithms for solving the wind farm layout optimization problem has been demonstrated.

In the present work only a discrete domain dividing the total wind farm area space into small cells for wind turbine positioning has been considered. As future work, it would be interesting to consider a continuous space search where a turbine can be installed at any points in a certain geographical area.

Although in this paper wind farms of interest have been solved, the wind farm developers can be supported to optimize large practical scenarios due to the simplicity of implementation of the procedures addressed. Moreover, the optimization procedures could be used in the design of both onshore and offshore wind farms by introducing of specific requirements for variables in the Jensen model such as the surface roughness coefficient.

Finally, it is clear that one of the main contributions of this research is the use of Excel Solver to optimize large scenarios of wind farms, this being an alternative to the programming languages and software which usually require extensive knowledge.

References

1. Yousefbeigi, S., & Akmandor, I.S. (2015). Wind Farm Optimization. In *International Symposium on Innovative Technologies in Engineering and Science. ISITES* (pp. 3-5).
2. Burton, T., Sharpe, D., & Jenkins, N. (2001). *Handbook of wind energy*. John Wiley & Sons.
3. Chen, Y., Li, H., Jin, K., & Song, Q. (2013). Wind farm layout optimization using genetic algorithm with different hub height wind turbines. *Energy Conversion and Management*, *70*, 56-65.
4. Mostafaeipour, A., Sedaghat, A., Dehghan-Niri, A.A., & Kalantar, V. (2011). Wind energy feasibility study for city of Shahrbabak in Iran. *Renewable and Sustainable Energy Reviews*, *15*(6), 2545-2556.
5. Fischetti, M., & Monaci, M. (2016). Proximity search heuristics for wind farm optimal layout. *Journal of Heuristics*, *22*(4), 459-474.
6. Samorani, M. (2013). The wind farm layout optimization problem. In *Handbook of wind power systems* (pp. 21-38). Springer, Berlin, Heidelberg.
7. Méchali, M., Barthelmie, R., Frandsen, S., Jensen, L., & Réthoré, P.E. (2006, February). Wake effects at Horns Rev and their influence on energy production. In *European wind energy conference and exhibition* (Vol. 1, pp. 10-20). Citeseer.
8. Barthelmie, R.J., Hansen, K.S., Frandsen, S.T., Rathmann, O., Schepers, J.G., Schlez, W., Phillips, J., Rados, K., Zervos, A., Politis, E. & Chaviaropoulos, P. K. (2009). Modelling and measuring flow and wind turbine wakes in large wind farms offshore. *Wind Energy*, *12*(5), 431-444.
9. Shakoor, R., Hassan, M.Y., Raheem, A., & Wu, Y.K. (2016). Wake effect modeling: A review of wind farm layout optimization using Jensen′ s model. *Renewable and Sustainable Energy Reviews*, *58*, 1048-1059.
10. Sahni, S., & Horowitz, E. (1978). *Fundamentals of computer algorithms*. Computer Science Press.
11. Chen, Y., Li, H., Jin, K., & Song, Q. (2013). Wind farm layout optimization using genetic algorithm with different hub height wind turbines. *Energy Conversion and Management*, *70*, 56-65.
12. Mittal, A. (2010). *Optimization of the layout of large wind farms using a genetic algorithm* (Doctoral dissertation, Case Western Reserve University).
13. Rivas, R.A., Clausen, J., Hansen, K.S., & Jensen, L.E. (2009). Solving the turbine positioning problem for large offshore wind farms by simulated annealing. *Wind Engineering*, *33*(3), 287-297.
14. Kirkpatrick, S., Gelatt, C.D., & Vecchi, M.P. (1983). Optimization by simulated annealing. *science*, *220*(4598), 671-680.
15. Colorni, A., Dorigo, M., & Maniezzo, V. (1992, December). Distributed optimization by ant colonies. In *Proceedings of the first European conference on artificial life* (Vol. 142, pp. 134-142).
16. Eroğlu, Y., & Seçkiner, S.U. (2012). Design of wind farm layout using ant colony algorithm. *Renewable Energy*, *44*, 53-62.
17. Eberhart, R., & Kennedy, J. (1995, November). Particle swarm optimization. In *Proceedings of the IEEE international conference on neural networks* (Vol. 4, pp. 1942-1948).
18. Wan, C., Wang, J., Yang, G., Gu, H., & Zhang, X. (2012). Wind farm micro-siting by Gaussian particle swarm optimization with local search strategy. *Renewable Energy*, *48*, 276-286.
19. Frontline Solvers. Standard Excel Solver-Limitations of Nonlinear Optimization. Available: https://www.solver.com/standard-excel-solver-limitations-nonlinear-optimization (2019). Accessed: 28 May 2019
20. Sorensen, P., & Nielsen, T. (2006, February). Recalibrating wind turbine wake model parameters—validating the wake model performance for large offshore wind farms. In *European Wind Energy Conference and Exhibition, EWEA*.
21. Donovan, S. B. (2008). *Whither the wind blows: wind flow modelling and wind farm layout optimisation* (Doctoral dissertation, University of Auckland).
22. Ozturk, U. A., & Norman, B.A. (2004). Heuristic methods for wind energy conversion system positioning. *Electric Power Systems Research*, *70*(3), 179-185.
23. Grady, S.A., Hussaini, M.Y., & Abdullah, M.M. (2005). Placement of wind turbines using genetic algorithms. *Renewable energy*, *30*(2), 259-270.
24. Huang, H.S. (2007, November). Distributed genetic algorithm for optimization of wind farm annual profits. In *2007 International Conference on Intelligent Systems Applications to Power Systems* (pp. 1-6). IEEE.
25. Mosetti, G., Poloni, C., & Diviacco, B. (1994). Optimization of wind turbine positioning in large windfarms by means of a genetic algorithm. *Journal of Wind Engineering and Industrial Aerodynamics*, *51*(1), 105-116.
26. Şişbot, S., Turgut, Ö., Tunç, M., & Çamdalı, Ü. (2010). Optimal positioning of wind turbines on Gökçeada using multi-objective genetic algorithm. *Wind Energy: An International Journal for Progress and Applications in Wind Power Conversion Technology*, *13*(4), 297-306.
27. Lackner, M. A., & Elkinton, C. N. (2007). An analytical framework for offshore wind farm layout optimization. *Wind Engineering*, *31*(1), 17-31.
28. Goldberg, D. E., & Holland, J. H. (1988). Genetic algorithms and machine learning. Addison Wesley.
29. Spillman, R. (1993). Genetic algorithms. *Dr. Dobb's Journal*, *18*(2), 26-30.
30. González, J. S., Rodriguez, A. G. G., Mora, J. C., Santos, J. R., & Payan, M. B. (2010). Optimization of wind farm turbines layout using an evolutive algorithm. *Renewable energy*, *35*(8), 1671-1681.
31. Frontline Solvers. Genetic Algorithms and Evolutionary Algorithms-Introduction. Available: https://www.solver.com/genetic-evolutionary-introduction#Population (2019). Accessed: 21 June 2019