24th ABCM International Congress of Mechanical Engineering
December 3-8, 2017, Curitiba, PR, Brazil

# COBEM-2017-1723
# WIND FARM LAYOUT OPTIMIZATION BASED ON CFD SIMULATIONS

**Luís Eduardo Boni Cruz**
**Bruno Souza Carmo**
University of São Paulo, Department of Mechanical Engineering, Av. Prof. Mello Moraes, 2231, São Paulo/SP
luis.boni@usp.br
bruno.carmo@usp.br

***Abstract.*** *The project consists of developing a tool for wind farm layout optimization based on Computational Fluid Dynamics (CFD) simulations of the atmospheric wind flow and inter-turbine interference using the open source code OpenFOAM and the Dakota optimization toolkit. Since it is not feasible to simulate a whole wind farm using the complete geometry of the wind turbines, the need for models to represent their effects on the wind flow and the interference of one turbine into the others arises, and the most commonly used model is the Actuator Disk model and its variations. Once defined the procedure for wind turbine behavior evaluation using a CFD model, the coupling between this model and Dakota optimization toolkit took place. With this new tool in hand, two configurations were tested in order to achieve the best wind farm layout in terms of power output that respects the imposed physical restrictions.*

***Keywords:*** *Layout optimization, Wind turbine, Actuator Disk, OpenFOAM, Dakota*

## 1. INTRODUCTION

Being one of the first stages of the project, the layout of a wind farm is a critical factor for its characterization. With the improvement of knowledge and competitiveness of this market, this factor is being highly exploited and taken to the extreme in order to maximize the profitability of each project. In most of the cases, there are few wind resource measurements in the area where the wind farm is being planned to be installed, normally between 1 and 10 meteorological masts for a many kilometers wide wind farm.

It is known that the commonly used tools on the market are not able to predict the wind flow behavior in regions with high topographic and roughness complexity, once they use linear models with many simplifications and limitations in addition to analytical wake models. For this reason, we chose to use a computation fluid dynamics (CFD) tool in this work, which despite having a higher complexity on its use, can achieve good results when correctly used (Tapia, 2009).

It was also noticed that there is a lack of CFD based optimization tools available on the wind market, so there is a need to have more accurate and reliable tools that can exploit as much as possible the available wind resource (Bonanni, *et al*., 2012).

The focus of this work will be on solving a simplified optimization problem considering a flat terrain with fixed roughness value and a predefined number of wind turbines. The objective of the algorithm is to find the best placement of the turbines so as to maximize the power output, taken into account some constraints regarding the distance between the turbines. The steady incompressible Navier-Stokes equations were solved and a genetic algorithm was used for the optimization task.

## 2. WIND TURBINE AND NUMERICAL FLOW MODEL SETUP

### 2.1 Actuator Disk Model

The actuator disk model has its origins on the 1D momentum theory (Mikkelsen, 2004). In this model, the swept area of the rotor is represented by a permeable disk in which the forces acting on the blades will be distributed. This distribution must be chosen accordingly to specific needs and the most common are the uniform and polynomial distribution.

As described in Simisiroglou, *et al*., 2016, a set of modifications on the classical actuator disk model was applied. The thrust force is applied according to the modified thrust coefficient curve, which is generated from the thrust coefficient definition $C_t = 4a(1-a)$.

By solving the definition above for the axial induction factor, we obtain Eq. (1).

$$a = \frac{1}{2}\left(1 - \sqrt{1 - C_t}\right) \quad (1)$$

The wind velocity on each point of the disc $U_1$can be estimated by combining the axial induction factor definition on Eq. (2) with Eq. (1) based only on the free wind velocity $U_\infty$ and $C_t$ curve.

$$a = \frac{U_\infty - U_1}{U_\infty} \quad (2)$$

$$U_1 = U_\infty\left(1 - \frac{1}{2}\left(1 - \sqrt{1 - C_t}\right)\right) \quad (3)$$

After that, the thrust force $T_i$ acting on each cell can be calculated as follows:

$$T_i = (C_t)_i \frac{1}{2}\rho\left(\frac{U_{1,i}}{1 - a_i}\right)^2 A_i \quad (4)$$

where $(C_t)_i$ is the thrust coefficient found from the modified thrust coefficient curve, $U_{1,i}$ is the wind velocity on the disk and $A_i$ is the area normal to the flow direction (the subscript $i$ denotes each cell of the actuator disk). Our interest is on the power extracted from the wind by the wind turbine, so the output power can be calculated from the sum of each individual cell's power according to its definition:

$$P_{out} = \sum_{i=1}^{n} P_i = \sum_{i=1}^{n} T_i U_{1,i} \quad (5)$$

One of the native actuator disk models from OpenFOAM (*actuationDiskSource)* was modified in order to implement this model. Additionally, this native model requires an upstream point from which the free wind velocity will be gathered, but this point may be on a disturbed region, so the presented modification can be used even if the turbine is inside the wake of other wind turbines or of the terrain.

### 2.2 Boundary Conditions

In a problem with a length scale of a wind farm (normally on a kilometric scale) care must be taken regarding the velocity profile along the domain. As previously observed (Hargreaves and Wright, 2007), the atmospheric boundary layer (ABL) is not maintained on a fetch domain and some modifications on the boundary conditions are needed in order to maintain it. These modifications were initially proposed by Richards and Hoxey, (1993) and are already implemented on OpenFOAM by default for the $k - \epsilon$ turbulence model.

On the inlet, it is assumed that the vertical wind profile follows a logarithmic law according to Eq. (6).

$$U(z) = \frac{U^*}{\kappa}\ln\left(\frac{z - z_g + z_0}{z_0}\right) \quad (6)$$

$$U^* = \kappa\frac{U_{ref}}{\ln\left(\frac{z_{ref} + z_0}{z_0}\right)} \quad (7)$$

Where $U^*$ is the friction velocity calculated on Eq. (7), κ is the Von Karman's constant, $z_g$ is the minimum z coordinate, $z_0$ is the surface roughness height, $U_{ref}$ is the reference wind velocity at height $z_{ref}$. In the same fashion, the turbulent kinetic energy $k$ and the dissipation $\epsilon$ shall be modified as shown on Eq. (8) and Eq. (9).

$$k = \frac{(U^*)^2}{\sqrt{C_\mu}} \quad (8)$$

$$\epsilon = \frac{\left(U^*\right)^3}{\kappa\left(z - z_g + z_0\right)} \tag{9}$$

Finally, the wall function for the terrain surface will consider the ground to be fully rough. This boundary condition is designed to be used in conjunction with the ABL inlet conditions and the velocity parallel to the ground cells $U_w$ is calculated by Eq. (10).

$$U_w = \frac{U^*}{\kappa}\ln\left(\frac{z + z_0}{z_0}\right) \tag{10}$$

## 3. OPTIMIZATION PROBLEM

### 3.1 Dakota description

Dakota is an open source software developed and released by Sandia National Laboratories (USA) and it was originally developed to provide a common set of optimization tools for a group of engineers solving structural analysis and design problems at Sandia in 1994. Today it is constituted of a variety of iterative algorithms and the most important available classes are Parameters Studies, Optimization, Uncertainty Quantification, Design of Experiments and Calibration. It was also developed to be easily coupled with other simulation codes, especially ones that can be run on the Shell or Command Prompt.

In this work only the Optimization class will be used and, among the several optimization methods available, the most suitable one for this problem is a Genetic Algorithm. This method has been used by Bonanni, *et al*., 2012 and Schmidt and Stoevesandt, (2015) because it is a derivative-free method that can solve non-smooth problems.

### 3.2 Genetic Algorithm

The basics of every evolutionary algorithm is to emulate the natural process of evolution by the population reproduction, selection of the most fitted individuals and by applying some of natural process, such as crossover and mutation. Starting from an initial population, the objective function of each individual will be computed, and the best fitted will be selected for reproduction in order to generate a new population and repeat the process. Each individual has its characteristics represented by a finite set of parameters (commonly referred as a chromosome), which will be exchanged (cross over) and eventually changed (mutation).

Cross over is the main mechanism acting on the production of new individuals because on this phase the child chromosome will be formed from a mix between each parent's chromosomes. The probability of cross over is generally high (about 0.8 to 0.95). Mutation is the process where random parts of the chromosomes are changed to try to avoid sterilizing a population and getting trapped on a local optimum. This mechanism has a much smaller probability of occurring when compared to cross over (about 0.1).

One of the advantages of this method is that a final set of optimal designs will be reached, which means that there may be more than one option for the designer to choose, giving an additional flexibility for the optimization process.

Although it may seem a simple random search, genetic algorithms are randomized and exploit historical information of past populations in order to generate new search points (Goldberg, 1989).

### 3.3 Problem formulation

This work focused on a layout optimization, so the chromosome of each individual contained a set of coordinates that represents the locations of the wind turbines on the $xy$ plane and the objective function $F_{obj}$ was simply the total available power of the wind farm, which will be calculated using Eq. (5) and summing over all wind turbines. This objective function shall be maximized.

$$F_{obj} = \sum_{j=1}^{n} P_j \tag{11}$$

A constraint to assure a minimum distance between wind turbines must be applied in order to avoid having overlapping actuator discs or very close wind turbines, which can cause an ill treatment of the problem by the CFD calculations. As a market good practice, this minimum inter turbine distance must be between 2 and 3 rotor diameters in order to avoid high turbulence zones produced by the wind turbine wake, so if any individuals had not respected this constraint it was immediately discarded.

We also chose to solve a discrete problem due to the reduced search domain, which lead to a reduced computational cost when compared to a continuous problem. To do that, we selected an area of approximately 1.5km x 1.8km from the original 2km x 2km domain on the xy plane where the wind turbines could be placed and then this selected area was discretized on a 10 x 10 grid. The prior selection of a restricted area inside the domain aimed to avoid interferences of the mesh on the actuator disc region and the boundaries and also to keep a certain distance from the outlet boundary.

A representation of the domain, restricted area and discretized grid can be seen on Fig. 1, being the vertices the possible location of the wind turbines. This figure represents a horizontal slice of the domain and it is worth nothing that each grid cell has 165m x 200m.

The chosen wind turbine model for this work was a G114-2.1MW from Siemens Gamesa, which has a 114m rotor diameter and 2.1MW of rated power, with a chosen hub height of 120m. This hub height was fixed, while the quantity of wind turbines was specified for each case.

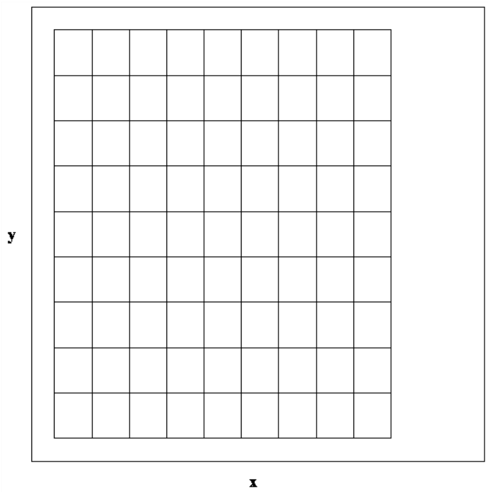


Figure 1. Optimization grid representation

## 4. WIND FARM OPTIMIZATION

### 4.1 General considerations

Once this is a problem in 3 dimensions, a domain size of 2km x 2km x 0.6km was considered. By doing so, we expected to avoid a strong interference on the boundaries that may be caused by the wind turbine regions and wakes.

For the following results, the quantities presented on Table 1 were not changed to try to keep each case's conditions as similar as possible. Two cases were examined maintaining the domain and increasing the number of wind turbines.

Table 1. Common parameters for the optimization cases.

| | |
|---|---|
| Wind direction (degrees) | 270 (positive x) |
| Wind velocity (m/s) | 8.0 |
| Reference height (m) | 120 |
| Surface roughness height (m) | 0.01 |
| Rotor diameter (m) | 114 |
| Huh height (m) | 120 |
| Air density (kg/m³) | 1.0 |
| Cross over probability | 0.95 |
| Mutation probability | 0.10 |

It can be noticed that the air density considered for both cases is different from the 1.225 kg/m³ of the standard air density. The incompressible solver *simpleFoam* solves all the equations divided by the fluid density, so an air density of 1kg/m³ is convenient, because no further calculation from the physical parameters must be done.

Additionally, the $k - \epsilon$ turbulence model was used with its standard coefficient on OpenFOAM, except for $\sigma_\varepsilon$ that was changed according to Richards and Hoxey, (1993) and is shown on Table 2.

Table 2. k - ϵ turbulence model constants.

| $C_\mu$ | $C_1$ | $C_2$ | $C_3$ | $\sigma_\varepsilon$ | $\sigma_k$ |
|---|---|---|---|---|---|
| 0.09 | 1.44 | 1.92 | -0.33 | 1.11 | 1.0 |

Regarding Dakota setup, the optimization method used was SOGA (Single Objective Genetic Algorithm) using discrete variables, with a randomly generated initial population and without any imposed constraint within Dakota. The inter turbine distance constraint was verified by an external script that set the objective function to zero if the minimum inter turbine spacing of two rotor diameters was not fulfilled.

### 4.2 Case 1: Wind farm with 12 wind turbines

The first case to be tested was a wind farm with 12 wind turbine generators with the aforementioned characteristics for a population of 30 individuals. Fig. 2 shows the objective function obtained for each layout evaluation carried out by the CFD model. From this result it is possible to notice that just a few individuals respected the imposed constraint until evaluation 1000 and after that this constraint was almost always fulfilled. It can also be noted that from evaluation 2000 onwards there were no significant changes on the best individuals and the populations tended to be more even.

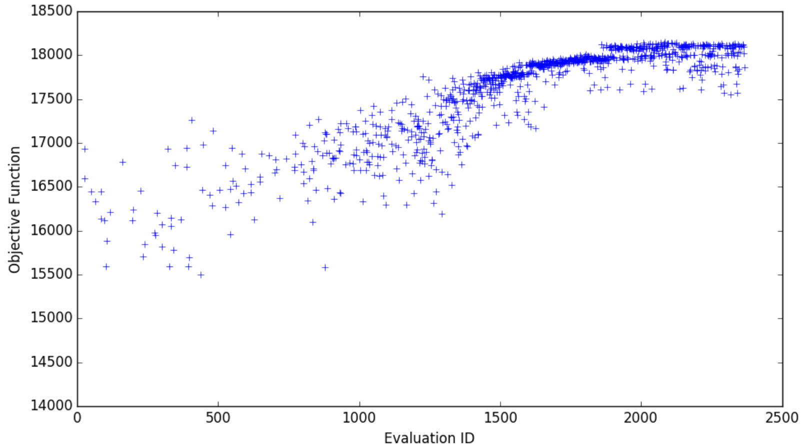


Figure 2. Objective function of every evaluation from Case 1

On the sequence, the best fitted individual of each population is plotted on Fig. 3a, showing the same behavior as Fig. 2 and indicating that the convergence was obtained at about population 100. Fig. 3b shows a horizontal slice of the whole domain colored by the wind velocity magnitude. In this figure, the best individual of the optimization is shown and it can be seen that the algorithm correctly avoided the wakes generated by the upstream wind turbines, reducing the power losses of the wind farm.

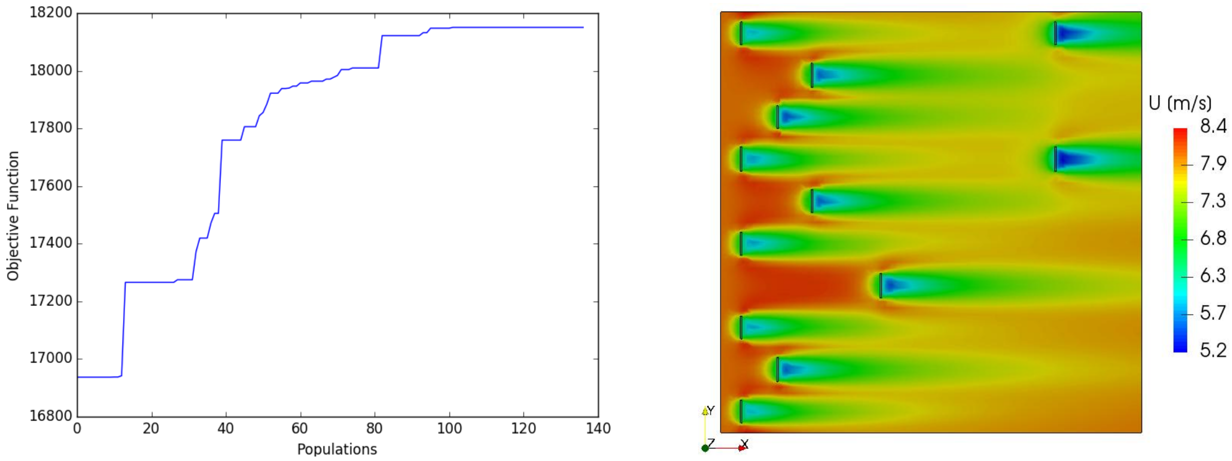


Figure 3. (a) Objective function of each population from Case 1; (b) Best individual from Case 1

One way to analyze the results is by means of the specific power, defined by the ratio between the total extracted power and the number of wind turbines. The extracted power calculated for a single wind turbine on a free stream using the described actuator disk model and the mentioned conditions is 1.546MW, thus, from Fig. 3(a) the calculated gain in specific power was about 7.2% and the specific power of the wind farm is 1.513MW, which gives a configuration with 97.8% of efficiency.

### 4.3 Case 2: Wind farm with 16 wind turbines

The second case consists of a wind farm with 16 wind turbines, but this time with 20 individuals on each population. In the same way, Fig. 4 shows the same behavior of Fig. 2, but for this case Dakota took more evaluations to find individuals that respected the imposed constraints due to the increase in the number of wind turbines. After evaluation 1500 the objective function kept steady, indicating a probable convergence.

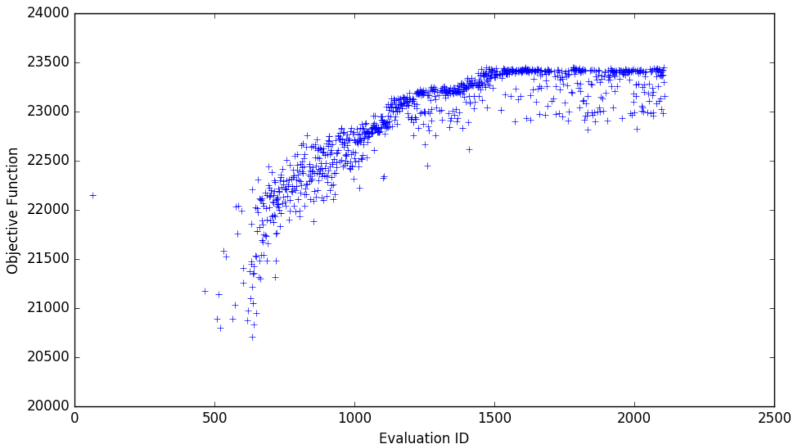


Figure 4. Objective function of every evaluation from Case 2

Fig. 5(a) and Fig. 5(b) present resembling results of Fig. 3(a) and Fig. 3(b), but, although the number of populations needed to reach convergence was about 200, the total number of objective function evaluations was very close to Case 1.

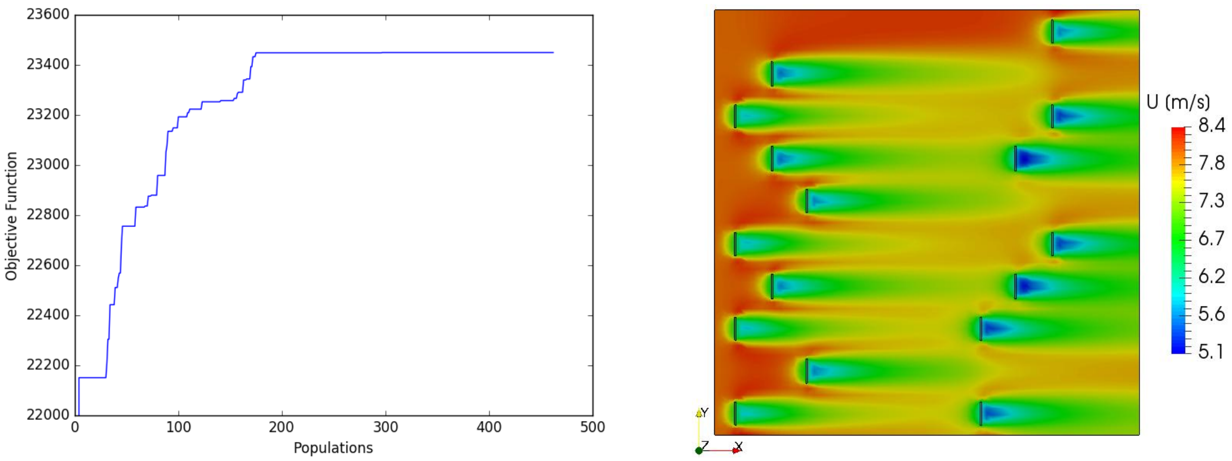


Figure 5. (a) Objective function of each population from Case 2; (b) Best individual from Case 2

Following the same interpretation of results, the calculated gain in specific power was about 5.9% and the specific power of the wind farm is 1.466MW, resulting on a configuration with 94.8% of efficiency. Table 3 presents a summary of the results from both cases.

Table 3. Summary of results from Case 1 and 2.

| | Number of wind turbines | Specific power (MW) | Gain with optimization | Efficiency |
|---|---|---|---|---|
| Case 1 | 12 | 1.513 | 7.2% | 97.8% |
| Case 2 | 16 | 1.466 | 5.9% | 94.8% |

## 5. CONCLUSIONS

We developed a tool for wind farm layout optimization making use of CFD for better wake modeling and the Dakota toolkit for optimization. Although there is a high computational cost when compared to semi empirical and approximate analytical models, CFD techniques for solving the flow field can give an expressive improvement of knowledge on the wake impact on the optimization process, justifying its use.

Two simplified scenarios were optimized using a Genetic Algorithm for a single objective function (total extracted power of the wind farm) and their results were analyzed in terms of gain in specific power and efficiency of the final solution. In both scenarios the results were considered promising and worth continuing the research.

The next steps of this work would be to fully validate the power curve obtained from the actuator disk model on OpenFOAM through either a comparison of the power obtained on each wind velocity with the manufacturer power curve or by comparing the wind velocity profile downstream the turbine using field measurements.

Regarding the optimization, the next steps for future developments would be to input an initial random population that already fulfills the spacing constraint so Dakota does not need to keep looking for a valid layout. By doing so, we believe that convergence could be reached faster. In addition, the wind conditions could be changed to more wind sector and wind speeds, so that real conditions of a wind farm could be taken into account on the optimization. Finally, a real terrain could be used instead of a flat terrain, also considering variable roughness all over the wind farm area.

## 6. ACKNOWLEDGEMENTS

The authors would like to thank Siemens Gamesa for supporting this work and providing information about the wind turbine chosen for the analysis.

## 7. REFERENCES

Bonanni, A. *et al*, 2012. “Wind farm optimization based on CFD model of single wind turbine wake.” In *Proceedings of the European Wind Energy Conference and Exhibition*. p. 1-10.

Goldberg, D. E., 1989. *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley Publishing Company, Boston.

Hargreaves, D. M. and Wright, N. G., 2007. *On the use of the k–ε model in commercial CFD software to model the neutral atmospheric boundary layer*. “Journal of Wind Engineering and Industrial Aerodynamics”, Vol. *95*(5), p 355-369.

Mikkelsen, R. F., 2004. *Actuator Disc Methods Applied to Wind Turbines*. PhD Thesis, Technical University of Denmark, Denmark.

Richards, P. J. and Hoxey, R. P., 1993. *Appropriate boundary conditions for computational wind engineering models using the k-ϵ turbulence model*. “Journal of wind engineering and industrial aerodynamics”, Vol. 46, p 145-153.

Schmidt, J. and Stoevesandt, B., 2015. “Wind farm layout optimisation in complex terrain with cfd wakes”. In *EWEA conference proceedings, Paris, France*, p. 17-20.

Simisiroglou, N. *et al*, 2016. “The actuator disc concept in PHOENICS”. *Energy Procedia*, Vol. 94, p 269-277.

Tapia, X. P., 2009. *Modelling of wind flow over complex terrain using OpenFoam*. MSc Thesis, University of Gävle, Sweden.

## 8. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.