Enhanced Wind Farm Layout Optimisation using Surrogate Models

Chen Zheng and Michael Mayo

Department of Computer Science, University of Waikato
Hamilton, New Zealand
cz90@students.waikato.ac.nz, mmayo@waikato.ac.nz

Abstract. Wind farm layout optimisation (WFLO) is the problem of finding the optimal geographical placement of wind turbines (WTs) inside a wind farm in order to maximise power generation while satisfying other constraints. Evolutionary Algorithms (EAs) are as powerful population based global optimiser, but WFLO problem requires very computational expensive fitness evaluations, thus even EAs are not efficient to solve the problem. On the other hand, surrogate models are also known as efficient meta-model for approximating expensive computation models. In this work, wind farm layout optimisation problem focuses on the single objective of energy cost of the wind farm. A standard analytical wake model has been used to calculate the velocity deficits in the downstream generated by individual turbines. Surrogate models are used as an approximation of the real fitness function in the evolutionary strategy to assist the optimisation. A previous presented blockcopy operation is used as the mutation method in the EA [17]. This paper carried out several tests to determine the various parameters of the surrogate models: what type of EA to use, which surrogate model to use, how to integrate the surrogate model into the EA, and how to ensure reliable approximation.

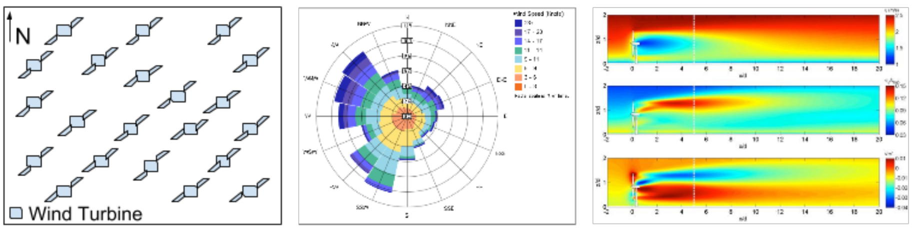
Keywords: wind farm layout optimisation, blockcopy, surrogate modelling, neural network

1 Introduction

With the rapid development of the global economy, wind power are increasingly more important around the world. The Global Wind Energy Council (GWEC) reported that globally wind power production reached 433 gigawatts (GW) by the end of 2015, a cumulative 17% increase. Furthermore, GWEC predicts that by the end of 2030, there will be about 2,000 GW of wind power spinning around the world [5]. It is clear that wind power is now a mainstream source of renewable energy supply and will play a leading role in the future. The wind industry is interested in using technical innovation to drive costs down, improve project reliability and predictability, and make it easier to integrate wind power into the main power grid.

Due to the limited energy generation of individual wind turbines, a wind farm normally is constructed by installing a large number of turbines on a given terrain. There is a trade off between two conflicting opposed economical factors: cost of construction and maintenance, and the profit of selling generated electricity. The optimal wind farm is considered to be one that maximises the power generation while minimising the total cost. In order to reduce cost and increase profit, wind farm designers are moving towards larger size farm, more turbines and other advanced capabilities. But installing numbers of turbines close together cause them to interfere each other due to the wake effect. Wind wake effect is a phenomena describes that when a wind turbine rotor extracts certain amount of kinetic energy from the wind flow and the downstream wind speed is reduced, also, there is turbulence and shear stress that will increase wind load fluctuation and cause turbine damage [14].

As an example shown in Fig. 1 (a), wind farm layout is the geographical placement of wind turbines inside a wind farm. Fig. 1 (b) is a typical wind rose diagram that illustrates the wind resource data such as wind speed, direction and percentage of time (frequency) around the farm. Fig. 1 (c) depicts three simulation results obtained using Large-Eddy Simulation with a Lagrangian scale-dependent dynamic model: averaged velocity (top), turbulence intensity (middle), kinematic shear stress (bottom) [25].



(a) Wind Farm Layout (b) Wind Rose Diagram (c) Wind Wake Effect

Fig. 1. Example of wind farm layout, wind rose diagram and wake effect

To develop practical solutions, in the computational intelligence community, related research works frequently focuses on the development of new wind farm design, optimisation and prediction techniques. The particular problem we are interested in this work is the wind farm layout optimisation (WFLO) problem [8, 18]. The main focus is to develop an approach that can still obtain the optimum but within a reasonable time for practical purposes.

In literature, evolutionary approaches are widely used in wind farm design problems [6, 13, 20], based on evolutionary approach, we propose a more sophisticated optimiser with the assistance of surrogate. Which surrogate to use is chosen based on a set initial experiments based on benchmark wind farm scenarios. Then the surrogate is integrated into our approach and it is trained during runtime based on runtime intermediate data. Inside the ES, we also used a highly efficient search operators in our optimiser [17].

The paper is organised as follows. In the next section is the background about the WFLO problem with detailed description. Then is the briefing of evolutionary strategies and surrogates. After that is out proposed optimiser for WFLO problem, and then followed by evaluation of out approach on four benchmark scenarios from the literature.

2 Background

2.1 Wind Farm Layout Optimisation Problem

Wind farm design and development is a highly complex process which involves multi-objective under various constrains such as technical, logistical, environmental, economical, legitimacy and social factors. Because of all these complicated factors, in many cases wind farm design problems are categorised as time consuming NP-hard problems [4]. So far, for optimisation problem that are NP-hard, no polynomial algorithms are proven to be efficient. In recent decades, many researchers have proposed approximate approach such as artificial neural networks and heuristic to solve them instead of traditional optimisation methods such as Linear Programming, Nelder-Mead Simplex, Lagrangian relaxation, Quadratic Programming, etc [1].

Generally speaking, heuristic approaches can be considered as a simple and fast way that provide satisfactory, but not necessarily optimal solutions to complex problems with large searching spaces. Meta-heuristic approaches are generalised heuristic methods that can be applied to specific problem with minor modifications. But in wind farm design problem, even heuristic and meta-heuristic approaches are not capable to obtain accurate solutions in reasonable runtimes due to the computational complexity of the wake model, which is at least $\mathcal{O}(n^2)$ where n is the number of turbines [8].

2.2 BlockCopy-based Evolutionary Strategies

The most common way to classify heuristic methods is based on trajectory methods vs. population-based methods [1]. More specifically, trajectory metaheuristics use a single solution during the search process the outcome is also a single optimised solutions. But due to limited capability of escaping local optima, populations based meta-heuristics are more commonly used. A population of candidate solutions which evolve during given iterations then return a population of solutions or until satisfied. The main population-based meta-heuristics include: Genetic Algorithms, Evolutionary Strategies, Scatter Search, Differential Evolution, etc.

Evolutionary strategies is a well known family of algorithms [2]. This paper utilised standard (μ, λ)-ES that was presented by Luke [16]. The (μ, λ)-ES uses a population of λ randomly generated individuals. Then the iteration begins with fitness assessment for all λ individuals. Then μ fittest ones are chosen to produce λ/μ children through mutation and crossover operations. The children generally replace the low fitness parents. After that there is a set of new λ individuals for next iteration of the algorithm . In the case of wind farm problem, the individuals are wind farm layouts that need to be optimised. The fitness assessment is the expensive computation for cost of wind farm energy output.

Recent studies show that a Turbine Replacement Algorithm (TDA) outperformed all other approaches including genetic algorithms, particle swarm optimisation, continuous development models in terms of finding satisfactory layouts across different benchmark scenarios [21, 24]. More specifically, the TDA algorithm simply moves a single turbine to a new valid location, followed by layout evaluation. If the new layout is worse than the original one, then the algorithm keeps modifying the original one until it finds a better layout and discard the original one.

The blockcopy operation is a novel approach to optimising a wind farm layout uniformly by *reusing* small regions within layouts by copying them [17]. The main difference is that blockcopy operations focus on a larger pattern of turbines rather than individual turbines [22]. The idea behind this approach is that a small part of a layout is optimised first, and then another small part of the layouts is replaced by the optimised one. To simplify the pattern, the blocks are limited to size $1km \times 1km$ for *BlockCopy mutation* and *BlockCopy crossover* operations. An extensive evaluation using evolutionary strategies in conjunction with Blockcopy operators shows that Blockcopy-based evolutionary strategies are more efficient.

2.3 Surrogate Modeling of Expensive Evaluation Functions

There are several issues involved with using approximating in evolutionary algorithms. First is to decide what type of fitness approximation to use. Surrogate-assisted evolutionary optimisation is more application driven. The wind farm design and optimisation problem generally requires very expensive computational function evaluations. Some studies show that using approximation is a viable alternative instead of using actual function evaluation [3]. However, in most cases, no analytical fitness function exists for accurately evaluating the fitness of a candidate solution [11]. Therefore, a functional approximation of the real evaluation function is introduced to reduce the cost evaluation. More specifically, an alternate and explicit expression is constructed for the objective function as the surrogate model.

Then is to determine what approximation model to use. It is hard to compare the different types of approximation in terms of performances as them can be problem dependent. In [10] Jin has suggested that approximation models should start from simple ones, for example, lower order polynomial models to see if the given training and testing data can be fitted with reasonable accuracy. If the simple model fails to fit high dimensional problems, then higher order polynomial or neural networks should be considered. In this work, due to the high dimensional nature of a wind farm optimisation problem, the initial experiments in conjunction with simpler models such as J48, M5P and SMO [7] are both failed. After that, the more sophisticated multi-layer perceptron (MLP) is integrated with a traditional evolutionary algorithm as the surrogate.

The next problem is to integrate the built surrogate model into EA. Surrogate can be applied to almost all the operations of evolutionary algorithms, such as population initialisation, mutation, crossover and fitness evaluation [3]. For example, surrogate models can be used to filter out poor candidate solutions among the new population generated by initialisation, crossover and mutation. Using approximation based fitness evaluation in order to reduce the number of real fitness evaluations is another promising approach [12]. Also, a surrogate model is employed to pre-screen poor solutions by using estimated fitness before real fitness evaluation [15].

Techniques for using surrogates to assist fitness evaluation can generally be divided into individual-based, generation-based and population based [10]. By individual-based, the real fitness function is used for actual evaluations for partial individuals in a generation whereas population-based approaches use the surrogate for fitness evaluation in part of the generations and real fitness function for the rest. In population-based approaches, different surrogates can be used for more than one sub-population during their co-evolution process. These methods above might suffer from using surrogate fitness values for selection rather than real fitness function value.

After the integration, the remain issue is the reliability of the approximation. The idea of using surrogates in evolutionary strategy is to reduce runtimes. But there is a tradeoff between accuracy and computational cost. Using the approximation along with actual experimental evaluation is extremely important to achieve reliable performance by the surrogate assisted EA. One of the difficulties in improving the approximation accuracy is due to high-dimensionality in the search space. The major difficulty is to determine the parameter and structure of the surrogate model. While the initial experiments is the key, in [9, 19], the surrogate can also co-evolve with the original problem for a optimal structure and parameter.

3 Proposed Surrogate-based Wind Farm Layout Optimiser

There are two closely related methods for surrogate based ESes called pre-selection strategy and best strategy [11]. As shown in Fig. 2 (a) and (b) respectively, assume that the size of population is λ and the size of parent is μ, in a (μ, λ)-ES using pre-selection strategy, $\lambda^{\star} \geq \lambda$ offspring are generated and then evaluated using the surrogate. Among them, λ best individuals are selected and re-evaluated using the expensive real fitness function, then they are passed down as μ parents of next generation. By contrast, the (μ, λ)-ES best strategy evaluates all the λ offspring by using the surrogate, then $\lambda^{\star} \leq \lambda$ best individuals are chosen based on surrogate fitness value, these individuals are re-evaluated using the expensive real fitness function.

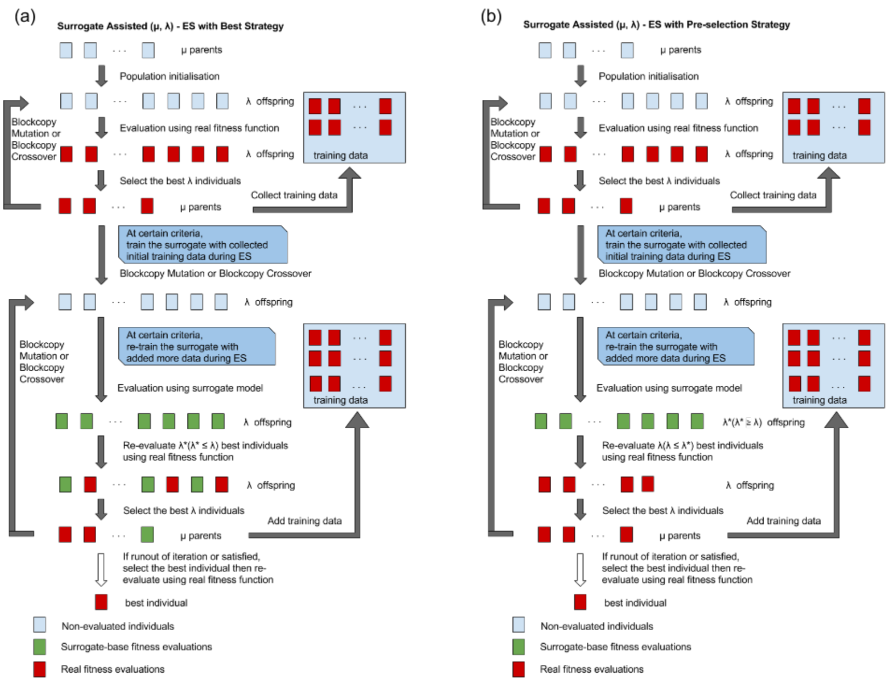


Fig. 2. Surrogate Assisted (μ, λ) - ES with Best Strategy and Pre-selection Strategy

The idea is to re-evaluate the individuals that potentially have a good fitness. The surrogate used in the evolutionary algorithm is to give a prediction of the individual's fitness. According to the predicted fitnesses, several winners are choose to be re-evaluated using the real fitness function. Then the parents for next generations are chosen from the population.

As shown in Fig. 2 (a), we propose the first approach that using surrogated assisted (μ, λ)-ES using best strategy with surrogate re-training procedure for wind farm layout optimisation problems. First is the complete random initialisation of μ parents to generate λ offspring. At the first certain generations, only real fitness function is used to evaluate the individuals, meantime, the initial training data is collected based on the real fitness value during this process. Then, at certain criteria, such as desired amount of training data or certain iteration of evolution, a surrogate is built using the collected training data. Then the evolution procedure continues in conjunction with the surrogate model. From now on, at each generation all λ individuals are first evaluated using the surrogate model, then, based on surrogate fitness value $\lambda^\star \leq \lambda$ best individuals are chosen to be re-evaluated using the real fitness function. At the same time, new training data is collected from the re-evaluated individuals. If runout of iteration or satisfied, the best individual is selected based on fitness value regardless of real or estimated, then it's re-evaluated using real fitness function.

Fig. 2 (b) shows the second approach called surrogated assisted (μ, λ)-ES using pre-selection strategy, the only difference from best strategy is that there are $\lambda^\star \geq \lambda$ offspring generated each generation, all of them are evaluated using surrogate model but only λ individuals with better estimated fitness value are picked for re-evaluation. Then the parents are only selected from re-evaluated individuals.

In both pre-selection and best strategy, to train a multilayer perceptron as the surrogate model, the initial training data is collected during first certain iterations of layout search in the format of raw cartesian coordinates of each turbine, then the coordinates are transformed into polar coordinates. Then the training data is feed into 20 hidden layers with random weight initialisation alone with a fixed learning rate, both are determined based on early experiments. After first training, the MLP is used as a surrogate to predictor layout fitness during search, meanwhile, good fitness layouts are also recorded to re-training the MLP to improve prediction accuracy.

4 Evaluation

4.1 Scenarios

To evaluate our WFLO optimiser, four benchmark scenarios are selected from the recent literature. The first two simpler scenarios are proposed by Kusiak & Song in [13]. Kusiak & Song's first scenario is an artificial example in which wind blows predominantly in a single direction while the second scenario is a more realistic one that describes the wind speed direction distribution at an actual industry wind farm [13]. In both scenarios, the size is limited to 4km by 4km and fixed 100 turbines.

The next two scenarios are selected from the 2014 Wind Farm Layout Optimisation competition [23]. With a larger rectangular area for more turbines and obstacles where turbines can not be installed. With 220 and 710 turbines, these two scenarios are far more challenging than the first two scenarios.

In table 1 is the basic basic descriptions of four scenarios, including dimensions, fixed number of blocks (across and down) for use with the BlockCopy operator. In each scenario, the entire site is divided into approximate 1km by 1km blocks. The Weibull k parameter is also differ from each scenario to provide an accurate estimate of wind speed at various turbine hub height.

Fig. 3 shows the wind speed and wind directions in each scenario. Each wind rose gives the proportionate expected wind speed in each direction. Directions are discretised into 15° bins. m/s in all directions. The rectangles in Fig. 4 indicate the obstacles configuration in scenario comp 1 and comp 3, respectively.

The cost function was originally proposed by Kusiak & Song in [13]. This paper uses an extended version from 2015 Wind Farm Layout Optimisation Competition [23]. By calculating the total cost of the farm (including construction and yearly operating costs) and dividing that by the total power output of the wind farm, the cost is defined as the expected cost of per kilowatt energy output. More precisely, the cost function is shown in Fig.1 where $c_t = 750,000$ is the cost of a turbine in USD; $c_s = 8,000,000$ is the cost of subsection in USD; $m = 30$ is the number of turbines per subsection; $r = 0.3$ is the interest rate; $y = 20$ is the lifetime of the farm in years; $C_{OM} = 20,000$ is the cost of operations and maintenance in USD; n is the number of turbines; and P is the total energy output of the farm. For more detailed and complete information please see [13] and [23].

$$cost = \frac{(c_t \times n + c_s \times \lfloor \frac{n}{m} \rfloor)(\frac{2}{3} + \frac{1}{3} \times e^{-0.00174n^2}) + C_{OM} \times n}{(1-(1+r)^{-y}) \big/ n} \times \frac{1}{8760 \times P} + \frac{0.1}{n} \quad (1)$$

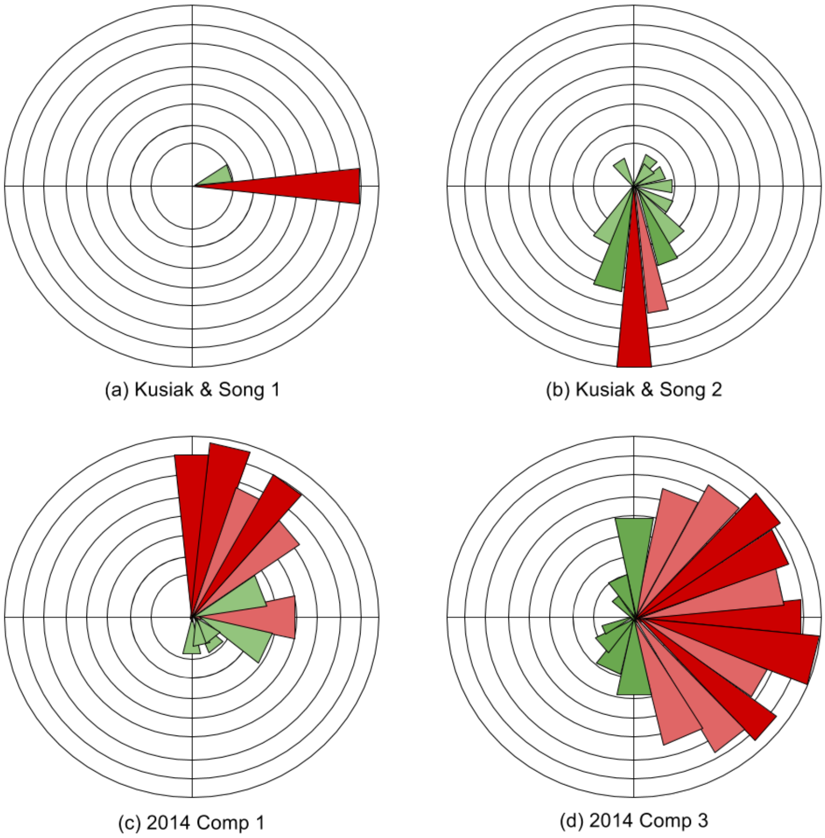


Fig. 3. Wind rose used in each scenario.

4.2 Results

// TODO

5 Conclusion

// TODO
To conclude, this paper has presented two approaches for the WFLO problem. Using the blockcopy operator in both mutation and crossover context, the experiments of have shown that the performance of (μ, λ)-ES best strategy is highly depend on the quality of surrogate while the (μ, λ)-ES pre-selection is ...

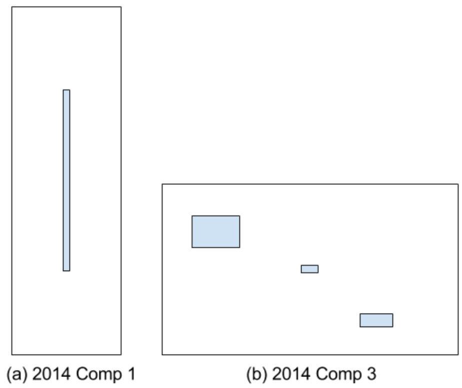


Fig. 4. Obstacles in scenario Comp 1 and Comp 3. Layouts are not shown to scale.

Scenario	Width (km)	Height (km)	# Turbines	Width (blocks)	Height (blocks)	Obstacles?	k
Kusiak & Song 1 [13]	4.0	4.0	100	4	4	No	2.0
Kusiak & Song 1 [13]	4.0	4.0	100	4	4	No	2.0
2014 Comp 1 [23]	3.5	16.1	220	3	16	Yes	2.187-3.624
2014 Comp 3 [23]	15.8	11.3	710	16	11	Yes	2.016-4.473

Table 1. Wind Scenario dimensions, number of turbines, number of blocks and k parameter

6 Notes

References

1. Baños, R., Manzano-Agugliaro, F., Montoya, F., Gil, C., Alcayde, A., Gómez, J.: Optimization methods applied to renewable and sustainable energy: A review. Renewable and Sustainable Energy Reviews 15, 1753–1766 (2011)
2. Beyer, H., Schwefel, H.: Evolution strategies: A comprehensive introduction. Journal Natural Computing 1(1), 3–52 (2002)
3. Bhattacharya, M.: Evolutionary approaches to expensive optimisation. International Journal of Advanced Research in Artificial Intelligence 2(3), 53–59 (2013)
4. Garey, M.R., Johnson, D.S.: Computers and intractability, vol. 29. wh freeman New York (2002)
5. Global Wind Energy Council: Global Wind Energy Outlook 2015 (2015)
6. González, J.S.and González, A., Mora, J., Santos, J., Payan, M.: Optimization of wind farm turbines layout using an evolutive algorithm. Renewable Energy 35(8), 1671–1681 (2010)
7. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: Weka 3: Data mining and open source machine learning software in java (2013)
8. Herbert-Acero, J.F., Probst, O., Réthoré, P.E., Larsen, G.C., Castillo-Villar, K.K.: A review of methodological approaches for the design and optimization of wind farms. Energies 7(11), 6930 (2014), `http://www.mdpi.com/1996-1073/7/11/6930`
9. Hüsken, M., Jin, Y., Sendhoff, B.: Structure optimization of neural networks for evolutionary design optimization. Soft Computing 9(1), 21–28 (2005)

10. Jin, Y.: A comprehensive survey of fitness approximation in evolutionary computation. Soft computing 9(1), 3–12 (2005)
11. Jin, Y.: Surrogate-assisted evolutionary computation: recent advances and future challenges. Swarm and Evolutionary Computation 1, 61–70 (2011)
12. Jin, Y., Olhofer, M., Sendhoff, B.: A framework for evolutionary optimization with approximate fitness functions. Evolutionary Computation, IEEE Transactions on 6(5), 481–494 (2002)
13. Kusiak, A., Zheng, H., Song, Z.: Power optimization of wind turbines with data mining and evolutionary computation. Renewable Energy 35(3), 695–702 (2010)
14. Leung, D.Y., Yang, Y.: Wind energy development and its environmental impact: A review. Renewable and Sustainable Energy Reviews 16(1), 1031 – 1039 (2012), `http://www.sciencedirect.com/science/article/pii/S1364032111004746`
15. Loshchilov, I., Schoenauer, M., Sebag, M.: A mono surrogate for multiobjective optimization. In: Proceedings of the 12th annual conference on Genetic and evolutionary computation. pp. 471–478. ACM (2010)
16. Luke, S.: Essentials of Metaheuristics. Online version 2.1 edn. (2014)
17. Mayo, M., Zheng, C.: Blockcopy-based operators for evolving efficient wind farm layouts. In: IEEE Congress on Evolutionary Computation, CEC '16. (2016)
18. Samorani, M.: The wind farm layout optimization problem. In: Pardolas, P. (ed.) Handbook of Wind Power Systems, pp. 21–38. Springer-Verlag (2013)
19. Schmidt, M.D., Lipson, H.: Coevolution of fitness predictors. IEEE Transactions on Evolutionary Computation 12(6), 736–749 (2008)
20. Vanaret, C., Durand, N., Alliot, J.M.: Windmill farm pattern optimization using evolutionary algorithms. In: Proceedings of the 2014 Conference Companion on Genetic and Evolutionary Computation Companion. pp. 181–182. GECCO Comp '14 (2014)
21. Wagner, M., Day, J., Neumann, F.: A fast and effective local search algorithm for optimizing the placement of wind turbines. Renewable Energy 51, 64–70 (2013)
22. Wagner, M., Day, J., Neumann, F.: A fast and effective local search algorithm for optimizing the placement of wind turbines. Renewable Energy 51, 64–70 (2013)
23. Wilson, D.: `http://www.irit.fr/wind-competition/`. URL (2015)
24. Wilson, D., Cussat-Blanc, S., Veeramachaneni, K., O'Reilly, U.M., Luga, H.: A continuous developmental model for wind farm layout optimization. In: Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation. pp. 745–752. ACM (2014)
25. Wu, Y.T., Porté-Agel, F.: Large-eddy simulation of wind-turbine wakes: evaluation of turbine parametrisations. Boundary-layer meteorology 138(3), 345–366 (2011)