WIND FARM LAYOUT OPTIMIZATION IN COMPLEX TERRAIN WITH CFD WAKES

Jonas Schmidt
Fraunhofer IWES, Oldenburg, Germany
jonas.schmidt@iwes.fraunhofer.de

Bernhard Stoevesandt
Fraunhofer IWES, Oldenburg, Germany
bernhard.stoevesandt@iwes.fraunhofer.de

Abstract:

For a complex terrain site in Bahia, Brazil, 28 CFD-RANS simulations were carried out, representing the relevant states of a wind rose with three degrees resolution. The resulting wind fields provide the background wind for the layout optimization of a wind farm with 64 wind turbines based on the AEP. The underlying wake model was deduced from CFD-RANS simulation results of an isolated actuator disk. We find that a hybrid optimization algorithm that combines genetic and gradient-based optimizers and subsequently increases the size of the wind farm yields good optimization results.

Keywords: Wind farms, wake models, complex terrain, layout optimization, CFD

1 Introduction

Wind farm layout optimization is crucial for advancing wind energy, since the successful minimization of wake losses both increases the annual energy production (AEP) of a wind farm and also reduces turbine loads. While in densely populated regions, like Germany, layout optimization for on-shore sites may not always be essential due to the strong constraints and the relatively small wind farm sizes, this may be different for other regions of the world. One example is Brazil, where the wind farms are typically large, the terrain is complex and wide regions are sparsely populated. Currently many wind farms in Brazil have line-dominated layouts, since easterly winds strongly dominate the wind rose. However, this may not be the most effective land usage of a given area, and eventually one may have to face the issue of wind farm optimization in complex terrain.

For a recent review on the topic of wind farm optimization and more than 20 years of related research see [1], also [2,3]. A summary on the related topic of optimised wind farm control can be found in [4]. Examples for state-of-the-art software on the industry level are *WindFarmer* [5], *WindPRO* [6] and *OpenWind* [7], a recent comparison of *WindPRO* and *OpenWind* can be found in [8]. Examples from the scientific community are TOPFARM by the Technical University of Denmark [9] and *flapFOAM* by Fraunhofer IWES [10–12].

Most approaches in the literature that describe wind farm layout optimization focus on off-shore or flat terrain scenarios. Also details from full computational fluid simulations (CFD) are usually not included in the calculation process. However, due to non-linear and non-local flow phenomena in the presence of terrain features, the latter may be the key ingredient in situations where the flow physics are complex and all other modelling fails. This paper gives the proof-of-principle that AEP optimization of a wind farm of reasonable size in complex terrain including wind potential fields from CFD calculations for a realistic wind rose and wakes from CFD simulations is possible.

All wake and wind farm modelling for this work has been achieved within the framework of the software *flapFOAM*, which has been developed at Fraunhofer IWES since 2011. For the optimization the software has been coupled to the powerful optimization tool box Dakota [13] by Sandia National Laboratories, USA, on a c++ library level [12]. *flapFOAM* was inspired by the software *FLaP*, which had been developed earlier at the University of Old-

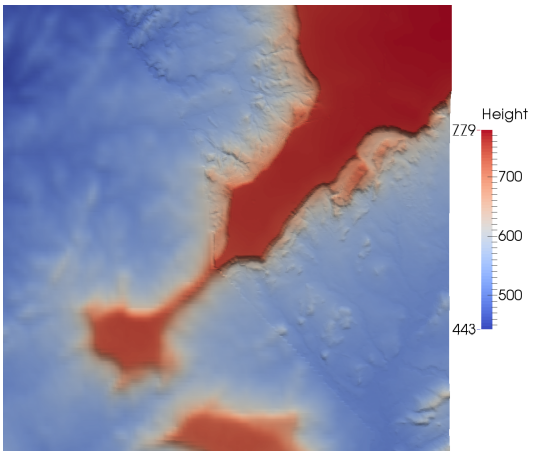

Figure 1: *The altitude of the terrain at the site.*

Figure 2: *The ground patch of the cylindrical fine mesh, and the wind farm boundary of size* 6×6 *km*^{2} *(black square).*

enburg (cf. [14]), without including code of the latter. The software is based on the idea of single-wake superposition, fully written in c++, and can read OpenFOAM [15] simulation results. Its strictly modular structure allows the developer to extend and improve models independently of the core functionality of the code, and the user to select between a broad range of models and settings. The proof-of-principle of the numerical wake model based on CFD solutions of the Reynolds-averaged Navier-Stokes equations (RANS) in presence of an actuator disc (AD) was presented in [10], and progress on the inclusion of complex terrain effects was reported in [11]. The order of magnitude of the uncertainty due to the choice of wake model during layout optimization was estimated in [12], which also includes a brief summary of the basic calculation algorithms of *flapFOAM*. A detailed description of the software will be given elsewhere.

The paper is organized as follows. In Section 2 the site of interest is briefly introduced, for which wind field simulations have been performed as described in Section 3. Section 4 summarises the numerical wake model that is applied to these background wind fields during layout optimization in Section 5. The method and results are discussed in Section 6 before we conclude in Section 7.

2 Site description

We study a fictional wind farm in complex terrain at a site in Bahia, Brazil, that features steep slopes and plateau regions. The altitude varies over a range of 336 m, cf. Fig. 1.

The model wind farm consists of 64 wind turbines of identical rotor type, and is initially arranged in a regular 8×8 pattern of size 5.8×5.8 km^2. The available area for the layout optimization is a square of 6×6 km^2, with orography as shown in Fig. 2. The numerical wind turbine model that is considered in this work has $D = 120$ m rotor diameter, $H = 120$ m hub height and 2.5 MW nominal power. In what follows the effective wind speed at the rotor is obtained directly from the centre point of the disk.

3 Background wind fields

The wind rose from Fig. 3 contains 120 sectors and up to eight wind speed bins with 2 m/s width per sector. Since winds from east-south-east (ESE) are very dominant, as it is typical for north-eastern Brazil, only a subset of the sectors is relevant. By ignoring states with frequency below 1% we reduce the number of considered wind rose states to 28. For each of these states, consisting of the wind direction of the sector and the centre of the wind speed bin, a CFD-RANS simulation of the flow over the terrain in neutral stratification is performed. The simulation results provide the background wind fields for the relevant inflow conditions; they represent the input flow states for the AEP optimization of the wind farm.

All simulations were carried out for the same

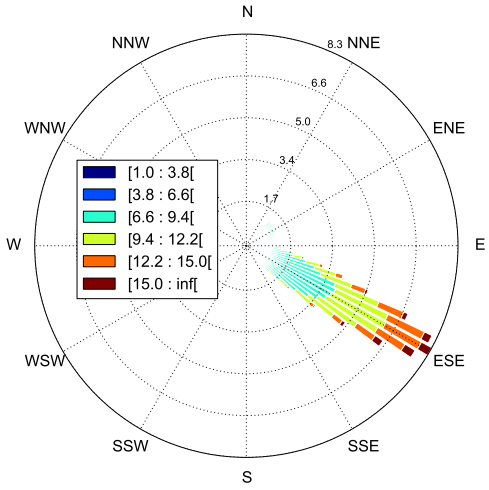

Figure 3: *The wind rose in 100 m height.*

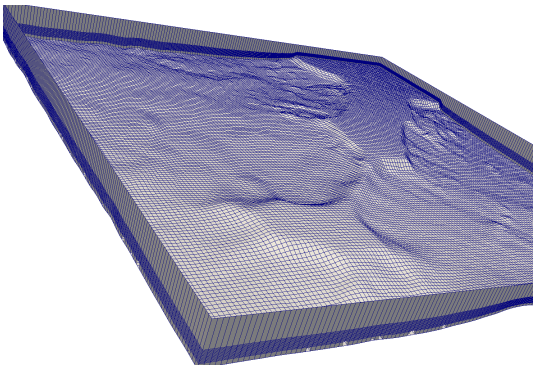

Figure 4: *The coarse mesh, not used for CFD simulations.*

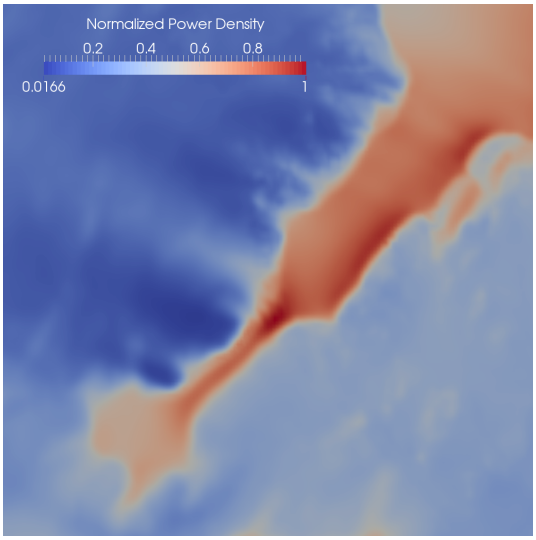

Figure 5: *The normalized mean wind power density in 120 m height above terrain.*

structured cylindrical mesh with 2.8 mio. cells, a radius of 10 km and 1 km height, called the fine mesh in the following. The terrain is resolved within a square of 10×10 km^2, cf. Fig. 2. The horizontal resolution in the central region is 50 m. The fine mesh has 50 levels in upward direction, with first cell height of 1 m and at least 10 m resolution within 200 m above ground. All meshes used for this work were created using the IWES in-house tool *terrainMesher*, which is a follow-up of the open-source *terrainBlockMesher* [16].

The OpenFOAM solver *simpleFoam* (version 2.3.1) was used to solve the RANS equations with standard k-ϵ turbulence model, with parameters adjusted for ABL simulations [17]. Wall functions were used at the ground, the roughness length was chosen uniformly as 5 cm.

The inflow profiles for the wind velocity field U and the turbulence fields k and ϵ were obtained by consistently solving a single column of cells with cyclic boundary conditions, given the mass flow according to a standard log-profile. The desired profile and the inflow wind speed at 120 m above ground were well matched by the results of this precursor simulation. For the different wind directions the inflowing velocity profiles were rotated accordingly. At the cylindrical boundary of the domain either the profiles or vanishing gradients were imposed, depending on the relation of the flow vector and the face normal. The whole procedure is fully automatized and parallelized, here 16 cores were used for each of the states. All simulations converged with residuals below 10^{-4} for pressure and below 10^{-5} for all other fields.

To speed up the interpolation of the background wind field results during optimization a second mesh with with 0.4 mio. cells was created afterwards, called the coarse mesh in the following. As shown in Fig. 4 it only covers 8×8 km^2 of the central region of interest. In the range of 50–190 m height over terrain the vertical resolution is 10 m, horizontally it is 50 m.

The 28 resulting fields are associated with frequencies, according to the wind rose. The mean wind power density can then be calculated by an integration, the result at 120 m height over terrain is shown in Fig. 5. Clearly the speed-up at the plateau and also its wake are visible. It can be

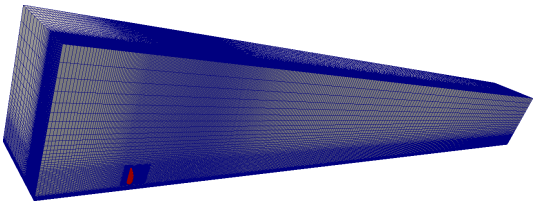

Figure 6: *Cut through the mesh and the actuator disk (red).*

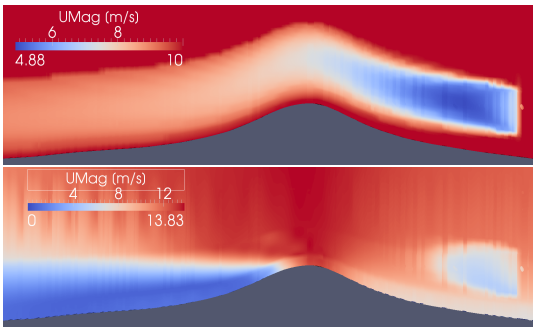

Figure 7: *Detail of a single CFD wake, in uniform background (top) and a CFD background solution (bottom), visualised in the coarse mesh. Notice that the flow direction is from right to left.*

expected that the optimal layout prefers the south-easterly borders of the elevation and tries to avoid the west-northern part of the domain. Note that the field shown in Fig. 5 is not used during optimization, instead the individual CFD results as stored in the coarse mesh enter the calculation.

4 Numerical wake model

Basically the 3D-RANS equations applied to an isolated actuator disk define a (4+x)-equation wake model, where x represents the turbulence model equations. Due to the complexity of CFD simulations they are obtained before run time of *flapFOAM* and span the range of inflow wind speeds of interest. Details of the implementation of a numerical wake model based on pre-calculated CFD-RANS results are given in [10]. Here we apply the wake model from our previous work [12], which is briefly summarised below.

Eight CFD-RANS simulations of a single uniform actuator disk in neutral stratification were run with OpenFOAM's *simpleFoam* solver (version 2.3.1), at inflow wind speeds 3, 5, 8, 10, 12, 15, 18 and 20 m/s at hub height 120 m. For intermediate inflow wind speeds, local second order interpolation is applied.

The mesh has dimensions $8.8 \times 1.5 \times 1.0$ km^3. It consists of 2.05 million cells, including the actuator disk with 1892 cells, cf. Fig. 6. The first cell height at the ground is 1 m and standard wall functions with roughness length 5 cm were used. Both grading and refinement were applied to improve the resolution of the wake and the near-disk region.

We applied the k–ϵ–f_P turbulence model [18] with parameters as recommended there. Compared to the standard k–ϵ model this version includes a correction of turbulent viscosity that depends on the change of velocity gradients due to the presence of the actuator disk, enhancing the wake deficit. All boundary conditions at the inlet were obtained by a one-dimensional cyclic precursor run, as described before. All variables converged to residuals below 10^{-5} in all simulations.

The addition of wake deficits is preformed quadratically under the square root, and no partial wake or meandering models are applied. The total wake deficit is then added to the pre-calculated CFD background wind field in terrain following manner, cf. Fig. 7. An additional deformation of the wake due to the presence of complex terrain as discussed in [11] is not included in the current study and left for future work, we refer to Section 6 for further discussion.

5 Layout optimization

The objective function that is used in throughout this work is the total wind farm AEP, normalized by the product of the number of turbines and the maximal AEP of the turbine model. Note that this quantity never exceeds the value one. The optimization variables are the horizontal positions of the wind turbines. The optimization constraints are defined by the rectangular boundary and the requirement of a minimal distance of 2 D between any two turbines.

We apply a hybrid of the genetic algorithm *soga*, which is part of the JEGA library [19], and the gradient based optimizer *conmin* [20], both as available trough Dakota [13] (version 6.0.0). Our algorithm is sketched in Fig. 8 and described in the following. The idea of subsequent turbine optimization has been applied before, for example in *WindPRO* [6] and *OpenWind* [7] (for a summary see [8]). We re-

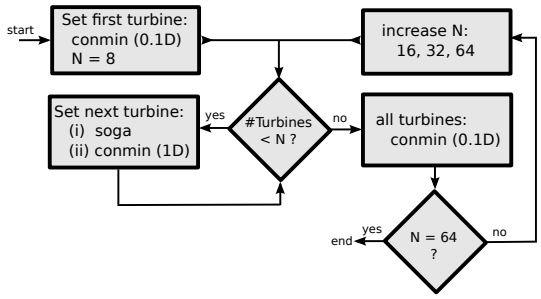

Figure 8: *Sketch of the optimization algorithm.*

Parameter	Value
Population size	10
Initialization type	unique random
Mutation type	replace uniform
Mutation rate	0.05
Replacement type	elitist
Constraint penalty	50
Max. function eval.	500
Convergence type	best fitness (20 gen., 1%)

Table 1: *Parameters of the* soga *algorithm, for details see [13].*

fer to Section 6 for a further discussion of the algorithm.

The first turbine is initially located near the south-eastern boundary of the domain, as shown in grey colour in Fig. 9. Starting from this position, a straight forward *conmin* search with step size 0.1 D finds the ideal position with maximal wind potential, compare Fig. 5 (red disk) and Fig. 9. The normalized AEP increases from 79.1% to 97.7%. However, for general initial positions a local optimization algorithm is not sufficient, due to many local maxima of the objective function and flat regions in the domain. Hence the need for a global optimizer, in our case a genetic algorithm, which is combined with subsequent local optimization for best results.

We subdivide the task of optimizing the layout or 64 turbines into sub tasks consisting of $N = 8, 16, 32$ and finally $N = 64$ turbines. Note that these numbers are coincidentally chosen as powers of 2, but that is no requirement for the algorithm. Until the total turbine number of wind turbines has reached the current N, turbines are subsequently added to the wind farm. The position of the new wind turbine is determined by the genetic *soga* algorithm with parameters as listed in Table 1, followed by the gradient-based *conmin* algorithm with step size 1 D.

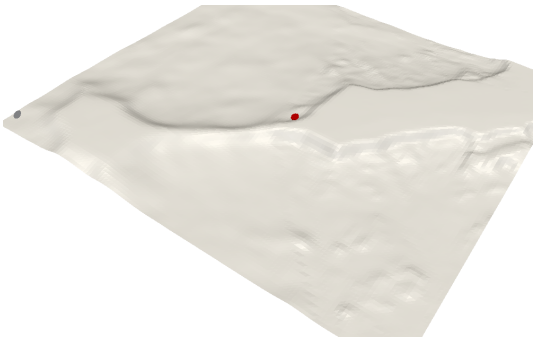

Figure 9: *The initial (grey) and optimized (red) first turbine position. Notice that the flow direction is from bottom to top of the image.*

If the new wind farm size equals N the complete layout is optimized again with the local optimizer and step size 0.1 D. Finally N is increased, or the algorithm stops if the maximal number of turbines has been reached.

The resulting layouts are shown in Fig. 10. For $N = 1, 8, 16, 32$ all wind turbines are placed on top of the plateau, as expected from the wind potential in Fig. 5. Also the narrow transition region between the two plateaus in the south-west and the north-east has been populated. The restriction of a minimal distance of 2 D between the turbines is apparent from the solution. Wakes effects are minimized according to the choice of representing the disk only by its centre point. The downstream area behind the elevation is avoided, upstream only hill tops are chosen for some of the turbines of the last optimization step $N = 64$.

Table 2 lists the normalized AEP values of the different steps after optimization. Clearly the single turbine case has the highest normalized AEP, since it can occupy the global maximum of the objective function. Up to limits of the genetic algorithm the turbines one after the other fill up the preferred regions of the wind potential in Fig. 5, yielding subsequently smaller AEP contributions. Finally, at large wind farm sizes, the wake effect further reduces the energy output of the wind farm.

6 Discussion

Flow over complex terrain in general is a complex phenomenon. It affects both the background wind and the wakes, in fact it remains to be shown that the superposition approach is even applicable in all

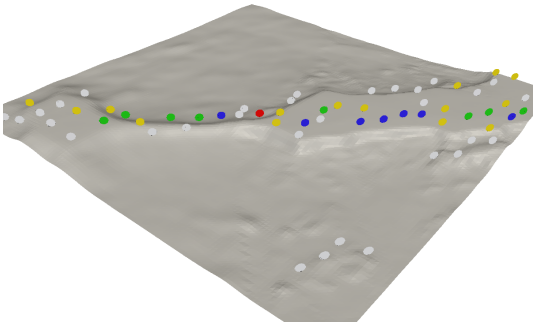

Figure 10: *The final optimized wind farm layout, with new turbines in red ($N = 1$), blue ($N = 8$), green ($N = 16$), yellow ($N = 32$) and grey ($N = 64$). Notice that the flow direction is from bottom to top of the image.*

N	1	8	16	32	64
AEP [%]	97.7	96.7	96.1	94.4	89.2

Table 2: *Optimization results in terms of the normalized AEP.*

cases. In this work we fully represent the effect on the background wind field by CFD-RANS simulations, which model the involved physics within the limits of the mesh and the turbulence model. Despite the fact that for realistic cases these simulations have to be validated before starting the optimization, the method is potentially more accurate for complex orography than other engineering methods.

As indicated in Fig. 7, the background wind solution captures the wake region behind hills and phenomena like flow separation. However, the wake transformation that has been applied in this work may be a very simple model to represent the real flow. A promising and more advanced CFD based approach has been studied in earlier work [11], and its generalization from isolated idealised hills to realistic orography is work in progress.

In principle the flow behaviour and especially the detachment of the wake at hill tops depends on stratification, and the strictly terrain following model that is applied here may only be relevant for modelling highly stable conditions. However, more research is needed to test this hypothesis, and generally to validate wake transformation functions in complex geometry. This is beyond the scope of the work presented here. Nevertheless, the flow accuracy in the wake of the plateau is not crucial for the studied layout optimization, since the wind rose clearly prefers south-easterly winds. Hence for the presented virtual wind farm one may argue that simple terrain following wakes may be sufficient, assuming that upstream and on top of the plateau the influence of model details is less significant, but again, this remains to be shown by comparing to measurement data.

Our optimization algorithm is a combination of a genetic and a gradient-based local optimizer, cf. Fig. 8. The turbines are added subsequently, and adding turbine number n comprises 2 optimization variables and $n-1+b$ constraints, where $b = 4$ is the number of constraints due to the wind farm boundary. As stated in Table 1 the population size of the evolutionary algorithm for the two variables is 10, and the maximally allowed number of objective function evaluations is 500, hence each optimization problem is relatively small and fast. We also evaluated both the genetic and the local optimizers also individually for the complete wind farm with 64 wind turbines. Both algorithms did not find satisfying solutions, in the latter case this is due to the complexity of the objective function. For the genetic algorithm a very large number of required function evaluations is needed for good results, in the studied case 10000 evaluations at population size 100 were not sufficient. Note that in that case the number of variables is 128 and the number of constraints is 2272. On a single core of a work station computer this required less than 48 hours, the algorithm from Fig. 8 less than 24.

As described in Section 5 and sketched in Fig. 8, our algorithm optimizes the complete layout with a local optimizer only when specific wind farm sizes N have been reached. This is a trade-off that has been made in order to speed-up the optimization as a whole, but in principle one may perform this step also after each turbine insertion. Furthermore it is straight forward to generalise the algorithm such that it ends after reaching the optimal number of wind turbines that complies with the optimization constraints.

The final layout from Fig. 10 reflects pure AEP optimization. For a realistic application more constraints need to be included, for example representations of the soil conditions and their suitability for realising the turbine foundation. Such constraints would possibly significantly influence some of the turbine positions, especially in the narrow transition region between the two plateaus at the site.

7 Conclusion

We demonstrated how CFD simulations of wind flow over complex terrain and CFD simulations of the flow through isolated rotors can be combined to realise wind farm layout optimization in complex orography based on CFD results. We found that a combination of a genetic algorithm for subsequently placing new turbines and gradient-based local optimization yields satisfying results.

All calculations were performed within the *flapFOAM* software framework. Once the precalculated CFD simulations were available, the computational time of the complete optimization on a single core of a workstation computer for a wind farm with 64 wind turbines was less than 24 hours.

One open issues is the validation of wake transformation functions in complex terrain. Also stratification and its impact on wakes at complex sites has to be included in the calculation. Furthermore the objective function and the optimization constraints need to be extended, for example to represent cable costs and other economic considerations. This is work in progress, as is the inclusion of turbulence intensity and wind turbine loads into the wind farm optimization.

Acknowledgements

The work is conducted in the scope of the project Smart Wind Farms funded by the German Federal Ministry for Economic Affairs and Energy (FKZ 0325851B). We would like to thank Casa dos Ventos Energias Renovaveis S.A. for providing the orography and typical wind rose data. We also thank Jerome Feldhaus for help with hybrid optimization solvers, and the computer time provided by the Facility for Large-scale Computations in Wind Energy Research (FLOW) at the University of Oldenburg, Germany.

References

[1] J. F. Herbert-Acero, O. Probst, P.-E. Rethore, G. C. Larsen, and K. K. Castillo-Villar. A review of methodological approaches for the design and optimization of wind farms. *Energies*, 7(11):6930–7016, 2014.

[2] M. Samorani. The wind farm layout optimization problem. Technical report, PowerLeeds School of Buisness, 2010.

[3] A. Tesauro, P.-E. Rethore, and G. C. Larsen. State of the art of wind farm optimization. In *EWEA*, Copenhagen, Denmark, 2012.

[4] Torben Knudsen, Thomas Bak, and Mikael Svenstrup. Survey of wind farm control-power and fatigue optimization. *Wind Energy*, 18(8):1333–1351, 2015.

[5] DNV GL. `http://www.gl-garradhassan.com/en/software/GHWindFarmer.php`, 2014. [Online; acc. 05-May-2014].

[6] EMD. `http://help.emd.dk/knowledgebase`, 2015. [Online; acc. 08-Oct-2015].

[7] AWS Truepower. `http://software.awstruepower.com/openwind`, 2015. [Online; acc. 06-Oct-2015].

[8] Eftun Yilmaz. *Benchmarking of Optimization Modules for Two Wind Farm Design Software Tools*. PhD thesis, Gotland University, 2012.

[9] P.-E. Rethore, P. Fuglsang, G. C. Larsen, T. Buhl, T. J. Larsen, and H. A. Madsen. Topfarm: Multi-fidelity optimization of wind farms. *Wind Energy*, 17(12):1797–1816, 2014.

[10] J. Schmidt and B. Stoevesandt. Wind farm layout optimisation using wakes from computational fluid dynamics simulations. In *EWEA conference proceedings*, Barcelona, Spain, 10–13 March 2014.

[11] Jonas Schmidt and Bernhard Stoevesandt. Modelling complex terrain effects for wind farm layout optimization. *Journal of Physics: Conference Series*, 524(1):012136, 2014. The Science of Making Torque from Wind, Copenhagen, Denmark.

[12] Jonas Schmidt and Bernhard Stoevesandt. The impact of wake models on wind farm layout optimization. *Journal of Physics: Conference Series*, 625(1):012040, 2015. Wake Conference, Visby, Sweden.

[13] B.M. Adams, L.E. Bauman, W.J. Bohnhoff, K.R. Dalbey, M.S. Ebeida, J.P. Eddy, M.S. Eldred, P.D. Hough, K.T. Hu, J.D. Jakeman,

L.P. Swiler, and D.M. Vigil. Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 5.4 user's manual. Technical Report SAND2010-2183, Sandia Technical Report, 2009.

[14] B. Lange, H. P. Waldl, A. G. Guerrero, D. Heinemann, and R. J. Barthelmie. Modelling of offshore wind turbine wakes with the wind farm program flap. *Wind Energy*, 6(1):87–104, 2003.

[15] OpenFOAM. http://www.openfoam.org, 2013. [Online; accessed 05-November-2013].

[16] Jonas Schmidt. `https://github.com/jonasIWES/terrainBlockMesher.git`, 2013. [Online; acc. 13-March-2013].

[17] A Crespo et al. Numerical analysis of wind turbine wakes. In *Delphi Workshop on Wind Energy Applications, Delphi, Greece*, 1985.

[18] M. Paul van der Laan, Niels N. Sorensen, Pierre-Elouan Rethore, Jakob Mann, Mark C. Kelly, Niels Troldborg, J. Gerard Schepers, and Ewan Machefaux. An improved k-epsilon model applied to a wind turbine wake in atmospheric turbulence. *Wind Energy*, 18(5):889–907, 2015.

[19] John Eddy and Kemper Lewis. Effective generation of pareto sets using genetic programming. In *ASME Design Engineering Technical Conference*, 2001.

[20] G. N. Vanderplaats. Conmin - a fortran program for constrained function minimization user's manual. Technical Report NASA TM X-62,282, NASA Technical Memorandum, 1973.