Response to Reviewer 1

Andrew PJ Stanley and Andrew Ning

October 28, 2019

Thank you for your thorough review of the manuscript and for your comments! We will address each of your comments and questions individually.

Question/Comments are in black.
The corresponding responses are immediately below in blue.

The reviewer very much appreciates the effort in presenting results in a clear, concise, and visually appealing way, as well as the availability of the computational codes. These two efforts contribute enormously to the understanding, and reproducibility of the results, which should set a precedent to all authors of this journal.
Regarding content, in spite of the massive oversimplification of the layout optimisation that could quickly lead to infeasible designs due to the high number of constraints industry faces in practice, it is very valuable to see that AEP-wise the direct and parameterised approaches are not that different. It is furthermore acknowledged that this academic effort to benchmark three design procedures so robustly with three different energy densities, shapes and windroses, provides high value and further evidence of AEP behaviour for this tremendously complicated optimisation problem.
Nevertheless, there are a few points for discussion, that should further improve the understanding of the proposed procedure and make the approach more transparent as well.

The optimal distribution of the turbines on the site boundary will depend on the windrose and shape of the site. Have you tested a more sophisticted algorithm to fill the perimeter with variable spacing according to wind direction and direction of the sites edges? Is a higher AEP expected than if placing them using a uniform spacing?

Yes, we have tested the parameterization where the boundary turbines are spaced equally perpendicular to the dominant wind direction. The idea was that this would avoid layouts where turbines are very close together parallel to the dominant wind direction. However, with the cases we tested we found that equally spacing the boundary turbines around the perimeter performed better. The following text will be added to Section 2.1 to explain this:
"During development of our parameterization method, we tested various strategies of spacing the turbines around the boundary. However, we found that equally spacing the turbines around the perimeter consistently provided the best results."

The authors suggest placing 45% of turbines on the boundary, when feasible. This sounds too case-specific. While I understand that the gradient-based optimisation algorithm requires a smooth function, and that letting the optimiser vary the spacing of the turbines on the perimeter and thus moving turbines inside the site would lead to "jumps" in the AEP response surface, I believe that fixing the number of turbines arbitrarily does not help the design space either. Would you suggest to re-run your method with different spacings/number of turbines on the perimeter? Is this done at all in the 100 randomly initialised runs of section 5?

This is an excellent point. Yes, 45% is a specific number which may be slightly sub-optimal for some specific cases. At length, we looked into the performance of BG parameterization for different numbers of turbines on the perimeter for different wind resources, wind farm boundary shapes and sizes. Our original goal was to find a relationship between some non-dimensional wind farm metric and the best ratio of turbines to place on the boundary. However, in every case we considered, placing 45% of the turbines performed the best or very close to the best compared to other amounts. This consistent good performance, along with the simplicity of having a this number as a constant led us to recommend the number of turbines of the boundary as a constant 45%.

Given sufficient computational resources, yes we would suggest this. However, if resources or time is limited, we would suggest using 45%. The following paragraph was added to section 2.2 to explain this:

"The process outlined to select the discrete variables used in the parameterization is recommended as a starting point, and when computational resources or time is limited. We tested many different methods of how to determine the discrete values, but found that the method shown above consistently produced wind farm layouts with high energy production. With sufficient resources, some scenarios may benefit from optimizing with a different ratio of boundary turbines, or different initializations of the boundary grid. However, the results discussed in this paper were produced with the method given in this section."

Why not make the spacing the design variable, and let the number of turbines on the boundary be variable. AEP surface would be too discontinuous?

Exactly. Discrete variables are not favorable for a gradient-based optimization. If desired, a user could certainly include the number of perimeter turbines as a design variable with a gradient-free approach. The following text was added to section 2.2 discussing this:

"Because these variables are discrete, they cannot be included as design variables when using a gradient-based optimization method, because the function space would be discontinuous. But, a gradient-free optimization may benefit from including some of these discrete variables as design variables in the optimizations."

A constant CT is assumed by the wake modelling, is there a noticeable difference in AEP compared to using a Ct curve?

The AEP with a constant CT is lower than that with a CT curve. A constant CT does not reduce the thrust after rated power is reached, making the predicted wakes stronger than reality. Although not of vital importance to the purpose of our paper, we are already rerunning the wind farm optimizations to make a correction in the mean wind speed, so the results of our revised submission will include a CT curve.

During the initialisation procedure suggested, dy is 4 times dx, is there empirical evidence for it?

The authors tested several different initialization methods for dy, and this method gave the most consistent good results. For some specific cases, a different initialization method may be desirable. However, for the cases we tested, this provided the best results overall. We added text to section 2.2 discussing this (shown in a response above).

However, if I understand correctly, dy is varied later to fit the desired number of turbines inside the site area, is this initial ratio not lost then?

Correct. This ratio only applies to the initialization of the discrete design variables, which are adjusted during optimization.

Also, the b variable is initialised to offset rows by 20 deg, is there a reason for this seemingly arbitrary value? Why not stagger rows by one half dx?

The authors agree, there is some arbitrariness to the initialization of b. We tested several different combinations of discrete variable selection, and included the rules that provided the most consistent and best results for us. Although for specific cases there may be a better method, in general the rules we provide worked well. Again, the paragraph we added for the revised paper in section 2.2 (shown in a response above) discusses this.

The initialisation procedure is meant to fix the number of rows and columns across the optimisation. The last paragraph of section 2.2 implies that the optimisation does not allow turbines to "jump" between rows, or to trade columns for rows. Is this what varies between the 100 runs of section 5?

That is correct. The following text will be added to Section 4 to clarify this idea:

"The random initialization was performed by fully randomizing the rotation variable θ and the boundary start location s, and defining the discrete and other design variables as defined in Sec. 2.2. The design variables dx, dy, and b are then randomly perturbed by plus or minus 10%. This random initialization method allows the number of rows and columns in the inner grid to differ between optimization runs."

How are the authors checking which turbines are inside the area? Can you share what algorithm you are using for that matter?

Certainly, we'll give a quick summary here and point to the code where the boundary calculations are made.
The wind farm boundary is defined with a set of sequential points, we assume straight lines between each of the points. Also note that the boundaries that we consider in this work are all convex. For a single turbine to one of the boundary lines:

1. Calculate the unit normal to the boundary line.
2. Calculate the vector defining the perpendicular distance between the turbine and the boundary line.
3. The constraint is then calculated as the dot product of these two vectors.

This is repeated for every turbine, to every boundary line. For a concave boundary, a slightly more complicated algorithm would be necessary, but this suffices for the current work. The boundary constraint code we used can be found here:

10.5281/zenodo.3261037

byuflowlab/stanley2019-variable-reduction/code/position_constraints.py

The function name is `calculate_distance`

We will add a note in the text in Section 4 that a link to the project code is included at the bottom of the paper.

"A link for the code used in this project is included at the end of this paper. Please refer to the code for specific details about how these constraints were enforced."

How are the authors defining the inner area in which the grid turbines must lie? Is there a uniform buffer spacing from the perimeter enforced?

There only thing defining where the inner turbines lie are the boundary and spacing constraints discussed in Section 4. There is no uniform buffer spacing. The following text was added to the revised paper in Section 4:

"No bound constraints, or additional constraints were used to define where the turbines must lie."

How do the authors foresee they will deal with prohibited zones inside the area?

This issue is beyond the scope of the presented research, however we have a few ideas on how this could be addressed. The following paragraph was added to Section 6 discussing this:

"Often, there are prohibited areas within a wind farm. This could be for many reasons, such as natural geography, roads or shipping lanes, or a variety of other reasons. Although beyond the scope of this paper, and not addressed in the results shown in Sect. 5, we have a few ideas on how this would be handled with BG parameterization. Many prohibited zones, such as shipping lanes, roads, or cable lines, are easily managed with a grid turbine layout, as these could easily be designed to follow the existing grid layout. Other prohibited zones could be handled by the BG parameterization, with no adjustments. This would be for cases where the prohibited zones are relatively small. For other cases, where the prohibited zones are larger and more restrictive, slight modifications would need to be made to the parameterization. The discrete variable of the inner grid would be initially defined such that the turbine location constraints are met. This would likely include some of the rows are not continuous, but have some gaps to accommodate the constraints. Likewise, the boundary turbines would be defined slightly differently, in that there would be some gaps to accommodate layout constraints."

How are turbines placed along the perimeter?

There is no "right answer" as to how to accomplish this, but we can briefly summarize how we accomplished this, then point to the code where we calculate the boundary turbine locations.
Preprocessing:

1. Calculate the perimeter of the wind farm boundary.

2. Divide the perimeter by number of turbines that are desired to have on the boundary, in this paper that was 45% of the total number of turbines. This gives the distance of the perimeter traversed between wind turbines.

3. If this spacing is greater than the minimum desired spacing times $\sqrt{2}$, the preprocessing is finished. If not, reduce the number of turbines until the perimeter traversed between wind turbines is greater than the minimum desired spacing times $\sqrt{2}$. The distance traversed around the boundary is simply the perimeter divided by the number of turbines placed on the boundary. The $\sqrt{2}$ is included to ensure that, except in extreme cases, the minimum turbine spacing is achieved for a convex wind farm boundary (i.e., the most extreme boundary angle would be 90 degrees).

Once the number of turbines and their spacings around the perimeter were determined, the location of each turbine around the perimeter was defined with a single variable, s.

1. First, an origin was defined. In our case, this was defined as the first point used to define the wind farm boundary.
2. Second, an "anchor turbine" was placed a distance s along the perimeter from the origin.
3. The remaining turbines were then placed such that all perimeter turbines are spaced equally traversing the wind farm perimeter.

The code is found here:

10.5281/zenodo.3261037

byuflowlab/stanley2019-variable-reduction/code/var_reduction_exact.py

The function name is `makeBoundary`

Is there consideration that two turbines near a corner could be closer than the minimum desired spacing?

Yes! The following text was added to section 2.2 to clarify this:

"When defining the number of turbines to be placed along the perimeter, the user must consider the most extreme boundary angles, such that minimum turbine spacing is preserved even at boundary corners."

What can be said of the results in Fig. 8 with respect to farm energy density? And in general, do the similar AEP results hold for all area densities?

The results in Figure 8 are intended to show that yes, we expect similar AEP results between the direct and parameterized optimizations regardless of the farm energy density. The following was added to the paper in Section 5.2:

"By showing the results for 3 different wind farm sizes, wind roses, and wind farm boundaries, we believe that our parameterization method can produce high AEP and optimize with reduced function calls for any scenario."

How would the authors deal in cases where all the internal and perimeter turbines have to align to an underlying base grid, for shipping and rescue operations?

Refer to our above discussion of prohibited zones within the parameterization.

What are the differences exactly between the 100 runs of the parameterised optimisation, the initial values of all variables? Different number of rows and columns? Or just the orientation angle theta?

The initialization of all of the design variables is randomized. The previously mentioned text we added to section 4 should clarify this:

"The random initialization was performed by fully randomizing the rotation variable θ and the boundary start location s, and defining the discrete and other design variables as defined in Sec. 2.2. The design variables dx, dy, and b are then randomly perturbed by plus or minus 10%. This random initialization method allows the number of rows and columns in the inner grid to differ between optimization runs."

Finally, is there future work aligned with this one? Are more/different variables interesting to look at for the design of wind farm layouts?

We do plan to implement the BG parameterization in future layout optimization studies, and perhaps make modifications based on the necessary constraints and design space. However, as for further development of the parameterization, there is no planned work directly associated with this one at the moment.

Technical correction: I suggest changing "verses" for "versus" in more than one place (e.g. line 25, fig 2).

This was corrected.

Response to Reviewer 2

Andrew PJ Stanley and Andrew Ning

October 28, 2019

Thank you for your thorough review of the manuscript and for your comments! We will address each of your comments and questions individually.

Question/Comments are in black.
The corresponding responses are immediately below in blue.

This paper proposed an interesting parameterization method for wind farm layout optimization, that has the potential of largely reducing the number of design variables. In general, the paper is well written, the new method is useful and results seems promising.
However, there are some major concerns the reviewer has on the current paper that he recommend this paper for a major revision. The major concerns are as follows:

1. Missing details in the proposed boundary-grid parameterization
As the central contribution of this study, the boundary-grid parameterization is not presented in a complete and clear manner. After reading Section 2, the reviewer can't figure out how exactly the 5 design variable can determine one and only one layout inside the specified boundary with a given number of turbines.

For example, if dx and dy is too big, the number of turbines you can put in the inner grid will be very few, then there might be too many turbines placed on the perimeter, that violate the minimal spacing constraints.

Thank you for bringing this up, our explanation in section 2.2 may have been lacking. We have added the following text to section 2.2 in order to clarify this:

"Note that the number of boundary turbines is determined before the number of turbines in the inner grid, to ensure that sufficient spacing in maintained between the boundary turbines."

Also the same set of dx, dy, theta and b can define a set of grid points that actually shift in the boundary, which will correspond to different layout. So the reviewer would argue that dx, dy, theta and b alone can't have a one-to-one map to a exact location of grid point.

The following was added to section 2.1 to clarify the parameterization:

"The inner grid is centered around the wind farm center, ensuring a one-to-one mapping from the design variables to the possible wind farm layouts."

The selection of discrete values also seems a little bit arbitrary.

The authors agree, there is some arbitrariness to the selection of discrete variables. We tested several different combinations of discrete variable selection, and included the rules that worked the best for us. Although for specific cases there may be a better method, in general the rules we provide worked well (see the first paragraph of section 2.2). We have added the following paragraph to Section 2.2 that addresses this concern:

"The process outlined to select the discrete variables used in the parameterization is recommended as a starting point, and when computational resources or time is limited. We tested many different methods of how to determine the discrete values, but found that the method shown above consistently produced wind farm layouts with high energy production. With sufficient resources, some scenarios may benefit from optimizing with a different ratio of boundary turbines, or different initializations of the boundary grid. However, the results discussed in this paper were produced with the method given in this section. Because these variables are discrete, they cannot be included as design variables when using a gradient-based optimization method, because the function space would be discontinuous. But, a gradient-free optimization may benefit from including some of these discrete variables as design variables in the optimizations."

It is stated in lines 87- 88, the discrete values remain fixed, but then again, you have the situation that there are too many grid points inside the boundary (when dx and dy are small), if you have to put 45% turbine around the boundary, you will have to remove some grid points, then which ones to remove according to what rule?

With very small wind farms (much less 4 rotor diameter average turbine spacing), our suggested discrete variable initialization would not be able to meet spacing constraints and boundary constraints. The optimizer should be able to handle this, and adjust dy and dx such that all the constraints are satisfied, however it would be helpful to start with a feasible layout. We have added the following to section 2.2 to clarify this:

"For extremely small wind farms, with an average turbine spacing much less than 4 rotor diameters, it may be impossible to initialize the turbine rows with dy equal to be four times dx and meet the minimum spacing constraints. In this case, the discrete row variable initialization would need to be adjusted."

For even more extreme cases, where you can't fit all of the turbines in the wind farm because the boundary is too small, you would just need to reduce the number of turbines desired in the wind farm and repeat our initialization process. This needs to be done in any layout optimization however, and is not unique to our study.

2. Some shortages in wind farm modelling.
First, in lines 117, it says "the turbulence intensity is equal to 0.0325", but shouldn't turbulence intensity change upon the wind speed?

Our revised paper will include results with the full 2016 Bastankhah Gaussian wake model rather than a simplified version. Details on this model will be included in the revised draft. This model also has a k value that is dependent on the freestream turbulence intensity, which we will clarify.

Second, according to Eqs.(3-4), you use the wake deficit at the rotor center to represent the average wake deficit on the whole rotor, since there is no integration over the rotor area in Eq. (4). This is problematic, as the profile of wake deficit is a Gaussian shape, and the one point deficit in the rotor center could be overestimating the mean deficit, if the two turbines are perfectly aligned.

Our revised paper will include results where several wind speeds are sampled across the wake and averaged to find the effective wind speed used in the power calculation. This dramatically increases computational expense, but reduces the possibility of overestimating the mean deficit from the Gaussian wake.

Third, there are only 5 wind speeds, and 23 wind direction sectors used in the wind resource modelling, according to Eq. (7). It has been shown in some studies that you need finer discretization, for example in (Feng and Shen 2015) in your references. This kind of coarse discretization could give you artificially optimistic AEP gains. You may also check the follow paper for recommended discretization:
Feng, Ju, and Wen Shen. "Modelling wind for wind farm layout optimization using joint distribution of wind speed and wind direction." Energies 8, no. 4 (2015): 3075-3092.

The revised draft will report the optimized AEP calculated with 360 wind direction bins and 50 wind speed bins. To avoid restrictive computation time, the optimizations are still run with fewer bins, but the final results will be reported with finer discretization.

3. The missing of comparison to gradient-free optimization technique.
I understand the focus of this study is on the proposed parameterization. But without direct comparison of the gradient based optimizer to some gradient free ones, e.g., GA or RS, it looks unfounded and somehow biased for a lot of claims that says the gradient free method will be infeasible, or perform worse.

Both gradient-based and gradient-free methods improve. We aren't claiming gradient-free is worse than gradient-based at the smaller dimension. The main motivation for this work is to make these kinds of problems tractable for gradient-free approaches. It is well documented that gradient-free methods don't scale well to large number of design variables. Here are just a few:

Lyu, Zhoujie, Zelu Xu, and J. R. R. A. Martins. "Benchmarking optimization algorithms for wing aerodynamic design optimization." Proceedings of the 8th International Conference on Computational Fluid Dynamics, Chengdu, Sichuan, China. Vol. 11. 2014.

Rios, L. M., and Sahinidis, N. V., "Derivative-Free Optimization: a Review of Algorithms and Comparison of Software Implementations," Journal of Global Optimization, Vol. 56, No. 3, 2013, pp. 1247–1293. doi:10.1007/s10898-012-9951-y, URL https://doi.org/10.1007/s10898-012-9951-y.

Zingg, David W., Marian Nemec, and Thomas H. Pulliam. "A comparative evaluation of genetic and gradient-based algorithms applied to aerodynamic optimization." European Journal of Computational Mechanics/Revue Européenne de Mécanique Numérique 17.1-2 (2008): 103-126.

Thomas, J. J., and Ning, A., "A Method for Reducing Multi-Modality in the Wind Farm Layout Optimization Problem," Journal of Physics: Conference Series, Vol. 1037, No. 042012, Milano, Italy, The Science of Making Torque from Wind, Jun. 2018. doi:10.1088/1742-6596/1037/4/042012

Ning, A., and Petch, D., "Integrated Design of Downwind Land-Based Wind Turbines Using Analytic Gradients," Wind Energy, Vol. 19, No. 12, pp. 2137–2152, Dec. 2016. doi:10.1002/we.1972

But with only 5 design variables both gradient-free and gradient-based methods should produce good results. We will add the above citations on the poor scaling of gradient-free optimization with few design varaibles.

Also do you have bounds on the design variables?

No. The only constraints were the boundary and spacing constraints mentioned in Section 4 of the paper. The following text has been added to the paper:

"No bound constraints, or additional constraints were used to define where the turbines must lie."

How are the constraints handled in the optimization process? Penalty function?

We used the optimizer SNOPT, which is an SQP algorithm. A sentence in Section 4 was modified to clarify this:

"We used the optimizer SNOPT, which is a gradient-based optimizer that uses sequential quadratic programming, and is well suited for large-scale nonlinear problems such as the wind farm layout optimization problem "

Below is a note referring to the documentation of SNOPT for further details:

https://web.stanford.edu/group/SOL/guides/sndoc7.pdf

4. The claim on the infeasibility of gradient-free technique for large wind farm is unfounded. AS stated in lines 9-10, Our presented method unlocks the ability to optimize and study large wind farms, something that has been mostly infeasible in the past. But I found this unfounded, you can check the following paper:
Wagner, Markus, Kalyan Veeramachaneni, Frank Neumann, and Una-May O'Reilly. "Optimizing the layout of 1000 wind turbines." European Wind Energy Association Annual Event 205209 (2011).

This was reworded to say:

"Our presented method facilitates the study and both gradient-free and gradient-based optimization of large wind farms, something that has traditionally been less scalable with increasing numbers of design variables."

Also engineering wake models are very fast to run, it shouldn't become too heavy or even infeasible for a gradient-free optimizer applied to a wind farm with 100 turbines, even if needs 10000 evaluations.

Excellent thought. We do make several claims throughout the paper about the infeasibility of wind farm layout optimization with increasing design variables, specifically in regards to gradient-free optimization. First let's look at the paper you mentioned above. In this paper, they optimize the layout of 1000 wind turbines, which is impressive. However, we see that they used 20 cores, and a single optimization still took 12 days. On a single core, they estimate that a single optimization would take about 140 days! Now, for most applications, we believe that 140 days is infeasible, or at the very least restrictive. Even 12 days limits the study of wind farms due to computation expense.

Now let's compare to our experience. Even with our fast engineering wake model, fewer turbines, and exact gradients, the direct optimizations for the first draft of our paper took 4-6 hours each to complete. With the updated wake model, (added ct curve, increased number of samples in the wake, finer bin samples to evaluate the final AEP values), the optimizations are taking at least 10 hours, some much longer. These additions really start to add up. This is with exact analytic gradients, so no additional function calls are happening to estimate gradients. Central-differenced gradients would take (at least) 3 times as long to optimize, and a gradient-free approach longer still. Additionally, we are using only one core in each optimization! Although a week or a month or longer to optimize a wind farm may not be restrictive if it is a one off occurrence, this is almost never the case. Usually the objective is to optimize the farm several times with different parameters and considerations, to see how the layout and performance is affected. Cases such as this benefit greatly fast optimization, which is provided by our presented parameterization.
Additionally, higher fidelity models are not very fast to run. In these cases, reducing the number of function calls required to optimize by several orders of magnitude or more is very important. As computation improves, these higher fidelity models will be used in wind farm layout optimization. In these cases, efficient optimization will play an important role.

5. Some very relevant references are missing.
Especially studies on grid-like layout optimization. The parameterization for the inner grid has been proposed in a similar way in some studies already. You may find the following two of interest:
González, Javier Serrano, Ángel Luis Trigo García, Manuel Burgos Payán, Jesús Riquelme Santos, and Ángel Gaspar González Rodríguez. "Optimal wind-turbine micro-siting of offshore wind farms: A grid-like layout approach." Applied energy 200 (2017): 28-38.
Neubert, A., A. Shah, and W. Schlez. "Maximum yield from symmetrical wind farm layouts." In Proceedings of DEWEK. 2010.

We added a citation for the paper by Neubert, Shah, and Schlez. The paper by González et al. was already cited on line 41.

Some minor issues:
1. It is stated in lines 20-25 that "Although these methods can be highly effective for small numbers of design variables, the computational expense required to converge scales poorly, approximately quadratically, with increasing numbers of variables. Because of this poor computational scaling, many companies and researchers have been limited in the size of wind farms they can optimize, as the number of variables typically increases with the number of turbines." But I doubt that's the case, since there are already large wind farms be designed and built in the world. Also optimization studies have been conducted for large wind farms, such as Horns Rev 1 with 80 turbines, as in one of your references (Feng and Shen, 2015).

Refer to our discussion to your statment: "Also engineering wake models are very fast to run, it shouldn't become too heavy or even infeasible for a gradient-free optimizer applied to a wind farm with 100 turbines, even if needs 10000 evaluations."

Yes it can and has been done. But it has been at great computational cost. Our presented parameterization makes these types of studies much more manageable.

2. Lines 31-32 “Power losses of 10–20% are typical from turbine interactions within a wind farm (Barthelmie et al., 2007, 2009; Briggs, 2013), and can be as high as 30–40% for farms with closely spaced wind turbines (Stanley et al., 2019).” This is somehow misleading, power losses of 30-40% are the worst wake case, which doesn’t happen that frequent in reality. So the actually AEP loss due to wake effects should be usually lower than 10-20%.

This was reworded for clarification:

“Power losses of 10–20% are typical from turbine interactions within a wind farm, and can be as high as 30–40% for farms with turbines spaced within 3 rotor diameters of each other.”

These values don’t refer to worst case, but are in fact the overall wake loss (refer to the cited paper for more details).

3. Rosenbrock function is used to demonstrate the convergence of gradient based optimizer scales better than gradient-free methods. First, you need to show what is Rosenbrock function, or at least provide a reference.

A reference was provided.

Second, this function is a function that we actually know where are the optimums, thus, we can easily see when it has converged to a local minimum. But in real life applications, we often can’t analytically prove that we have reached a local minimum, such as in layout optimization.

True, which is why the Rosenbrock function is a good test function for determining the efficiency of optimization algorithms. Figures 1 and 11 of the paper and the associated discussions demonstrate the multimodality and difficulty of the wind farm layout optimization problem.

Third, for such problem, converge faster (typically for gradient based methods) is just one aspect, the other aspect is the quality of the optimized results, i.e., whether the solution is close to the global optimum. Usually it is know that gradient free methods converges slower but has a higher probability to reach the global optimum, while gradient based methods converge faster but are also easier to be trapped in a local minimum.

Correct. However with large problems, convergence speed is a very important aspect. This simple example was used to highlight (and we feel that it is done so effectively) the huge importance of efficient computation, and the extreme effects that inefficient optimization can have on computation time.

4. Eq. (6), U_mean should be scale factor of the Weibull distribution. Note that the scale factor is not the same thing as the mean wind speed, instead the mean wind speed should be a function of scale factor and shape factor.

This has been corrected in the revised manuscript. Final results include this correction, and it is represented in Equation 6.

5. Line 275-276 states that "BG parameterization, cabling requirements can be clearly minimized by running cables across each of the rows, and around the boundary without the need for complex cabling algorithms." This is not true, as you still need to decide the location of sub-station, the exact topology of the cables and select cable types for different connections, thus, not necessarily easier than any random layout.

This claim was removed from the paper.

Massive Simplification of the Wind Farm Layout Optimization Problem

Andrew P. J. Stanley and Andrew Ning

Brigham Young University Department of Mechanical Engineering, 701 E University Pkwy, 350 EB, Provo, UT, 84602

Correspondence: Andrew P. J. Stanley (stanley_andrewpj@byu.net)

Abstract. The wind farm layout optimization problem is notoriously difficult to solve because of the large number of design variables and extreme multimodality of the design space. Because of the multimodality of the space and often discontinuous models used in wind farm modeling, the wind industry is heavily dependent on gradient-free techniques for wind farm layout optimization. Unfortunately, the computational expense required with these methods scales poorly with increasing numbers of variables. Thus, many companies and researchers have been limited in the size of wind farms they can optimize. To solve these issues, we present the boundary-grid parameterization. This parameterization uses only five variables to define the layout of a wind farm with any number of turbines. For a 100 turbine wind farm, we show that optimizing the five variables of the boundary-grid method produces wind farms that perform ~~within 0.5% of~~ just as well as farms where the location of each turbine is optimized individually, which requires 200 design variables. Our presented method ~~unlocks the ability to optimize and study~~ facilitates the study and both gradient-free and gradient-based optimization of large wind farms, something that has ~~been mostly infeasible in the past~~traditionally been less scalable with increasing numbers of design variables.

1 Introduction

In 2018, wind energy produced 6.6% percent of the electricity use in the United States[1]. With current market trends and technology, the U.S. Energy Information Administration projects that this number will rise by 1% in both 2019 and 2020 (U.S. Energy Information Administration, 2019a), and the installed capacity will increase by 4% every year through 2050 (U.S. Energy Information Administration, 2019b). In order for the U.S. and the rest of the world to meet and exceed these projections, it is necessary to be able to create efficient turbine layouts for large wind farms. The wind farm layout optimization problem is notoriously difficult to solve because of the large number of design variables, computationally expensive models for high fidelity simulations, and extreme multimodality of the design space (see Fig. 1).

Because of the multimodality of the space and often discontinuous models used in wind farm modeling, the wind industry is heavily dependent on gradient-free techniques for wind farm layout optimization (Herbert-Acero et al., 2014). Although these methods can be highly effective for small numbers of design variables, the computational expense required to converge scales poorly, approximately quadratically, with increasing numbers of variables (Singg and Pulliam, 2008; Rios and Sahinidis, 2013; Lyu and Ma
. Because of this poor computational scaling, many companies and researchers have been limited in the size of wind farms they

[1]https://www.eia.gov/tools/faqs/faq.php?id=427&t=3

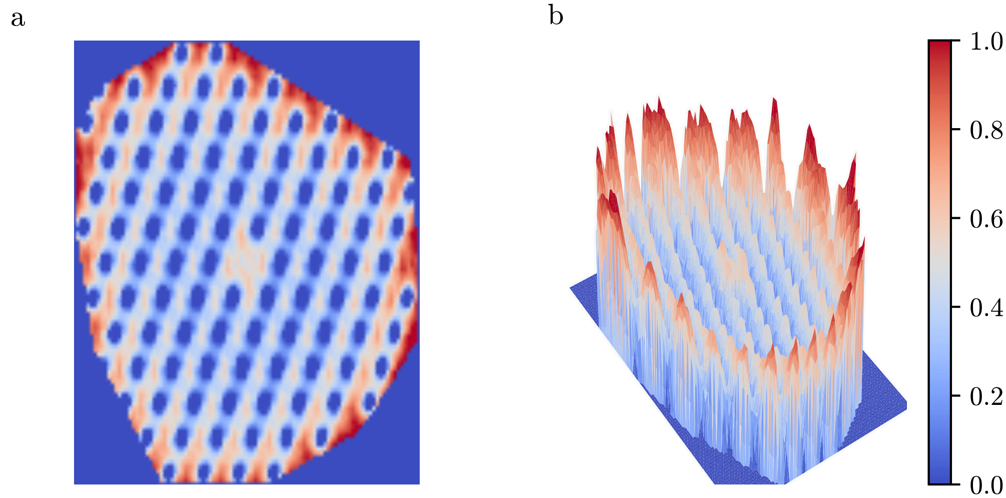

Figure 1. The complexity and multimodality of wind farm layout design space. Shown is the normalized annual energy production of a 100 turbine wind farm as a function of the location of one turbine. 99 turbines remain fixed, while one is moved throughout the wind farm. (a) A 2-dimensional view of the design space. (b) A 3-dimensional surface which highlights the extreme variation of the peaks and valleys. This figure shows only the multimodality from two dimensions, where the true design space has 200 design variables

can optimize, as the number of variables typically increases with the number of turbines. Figure 2 demonstrates this principle. This figure shows the number of function evaluations required to optimize the multi-dimensional Rosenbrock function ~~verses~~ versus the number of variables (Rosenbrock, 1960). To give a sense of what these numbers mean, if this problem with 64 variables and exact-analytic gradients takes one hour to optimize, using finite difference gradients would take almost four days, while a gradient-free method would take over 20 years! The trends, not the exact numbers, shown in this figure are general for other optimization problems, such as wind farm layout. As the size of the problem increases, the computational expense with certain optimization methods can become unmanageable.

Despite its difficulty, layout optimization is an essential step in wind farm development in order to maximize power production. Power losses of 10–20% are typical from turbine interactions within a wind farm (Barthelmie et al., 2007, 2009; Briggs, 2013), and can be as high as 30–40% for farms with ~~closely spaced wind turbines~~ turbines spaced within 3 rotor diameters of each other (Stanley et al., 2019). However, because the difficulties in finding optimal turbine placement increase with the number of turbines, layout optimization can quickly become infeasible for large wind farms (Ning and Petch, 2016). Even so, accelerated research and understanding of the principles governing wind energy, as well as public demand for renewable energy sources are encouraging developers and communities to install farms with more wind turbines than have been typical in the past. Current turbine layout definitions and optimization methods are woefully inadequate for these increasingly large farms.

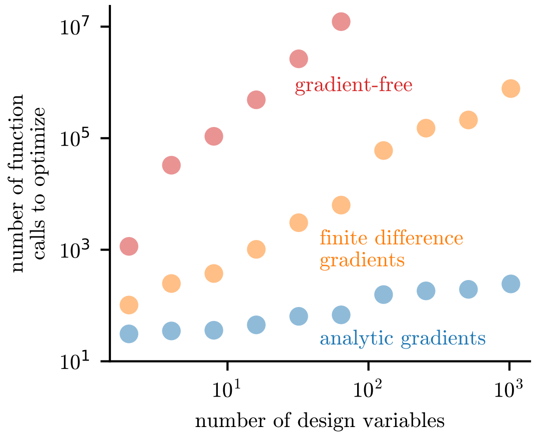

Figure 2. The number of function calls required to optimize the multi-dimensional Rosenbrock function verses versus the number of variables. The computational expense of gradient-free and finite difference gradients scale poorly with the number of variables.

The most common current wind farm layout definitions include defining the location of each turbine directly (Feng and Shen, 2015; Guirguis et al., 2016; Gebraad et al., 2017), preassigning some locations in a wind farm as suitable turbine locations to limit size of the design space (Emami and Noghreh, 2010; Parada et al., 2017; Ju and Liu, 2019), and parameterizing the turbines as a grid (González et al., 2017; Perez-Moreno et al., 2018)(Neubert et al., 2010; González et al., 2017; Perez-Moreno et al., 2018) . Defining the location of every wind turbine directly allows the most freedom, but also requires two variables for each turbine. In addition, the design space is the most multimodal. If one limits the design space by predetermining acceptable turbine locations or parameterizing the turbine locations with a simple grid, they are able to optimize larger wind farms. However, these methods produce simplistic wind farm designs, which underperform for most realistic scenarios.

In this paper we present the boundary-grid (BG) layout parameterization, a new wind farm layout parameterization. This new method solves the challenges that have previously made wind farm layout optimization so difficult. BG parameterization uses only five variables, and can produce layouts that perform within 0.5% of just as well as or better than the layouts achieved by directly optimizing the location of each wind turbine. With some of the most advanced wind farm optimization methods that have previously been available, we can directly optimize the location of every turbine in a 100 turbine wind farm in 4–5 hours. More common methods take on the order of days or longer. With BG parameterization, we can optimize a 100 turbine wind farm in 3 minutes. Additionally, this new parameterization dramatically reduces the multimodality of the design space compared to direct layout optimization (compare Figs. 1 and 13b). Finally, BG parameterization has additional benefits, including a regular, aesthetically pleasing layout , and naturally defined roads or shipping lanes, and easily defined cabling between turbines. This technique can immediately be applied to wind farm design to obtain excellent wind farm layouts with limited computational resources.

2 Boundary-Grid Parameterization

When the locations of wind turbines in a farm are optimized directly, the final layout often follows two general rules. First, a large fraction of turbines are grouped on or near the wind farm boundary. Second, the turbines that are not positioned on the boundary are loosely arranged in rows throughout the farm (Fig. 3a). By observing these patterns in optimal wind farm layouts, we defined our new layout parameterization such that it would create wind farms that filled these requirements.

2.1 New Layout Variables

In BG parameterization, the turbines are divided into two groups: the boundary and the inner grid (Fig. 3b). The boundary turbines are spaced around the circumference of the wind farm and are defined with one design variable. The rest of the turbines in the farm make up the inner grid, which is defined with four design variables for a total of five variables to describe the location of every turbine in the farm. The boundary turbines are placed on the wind farm boundary, spaced equally traversing the perimeter. These are defined by one variable, s, which is the distance along the perimeter where the first turbine, or start turbine, is placed. This in turn defines the position of every turbine around the boundary (Fig. 3c). During development of our parameterization method, we tested various strategies of spacing the turbines around the boundary. However, we found that equally spacing the turbines around the perimeter consistently provided the best results. The inner grid turbines are defined by four design variables: dx, dy, b, and θ. The grid spacing, dx and dy, are the distance between columns and rows in the grid, b is the offset distance, which defines how far consecutive rows are offset, θ is the grid rotation angle, which rotates the entire grid (Fig. 3d). The grid offset could also be defined as an angle, however we have used a distance as the gradients are more conducive to optimization. The inner grid is centered around the wind farm center, ensuring a one-to-one mapping from the design variables to the possible wind farm layouts.

2.2 Selection of Discrete Values

There are some discrete values which are important in our formulation, namely the number of turbines which are placed along the boundary and how many are in the grid, how many rows and columns are in the grid, and how the rows and columns are organized. We present some rules that we have found effective in determining these discrete values for all wind roses, wind farm boundaries, and wake models that we tested. Each individual case may benefit slightly from a more specialized selection of these values but our method works well across all cases tested.

The number of turbines placed on the boundary is determined by the wind farm perimeter and turbine rotor diameter. If the perimeter is large enough, 45% of the wind turbines are placed on the boundary. In some cases, the wind farm perimeter is small, and would result in turbines that are too closely spaced if 45% were placed around the boundary. In this case, the number of boundary turbines is reduced until the minimum desired turbine spacing in the wind farm is preserved. When defining the number of turbines to be placed along the perimeter, the user must consider the most extreme boundary angles, such that minimum turbine spacing is preserved even at boundary corners. No matter how many turbines are placed around the boundary, they are always spaced equally traversing the perimeter, and all of the remaining turbines are placed in the inner

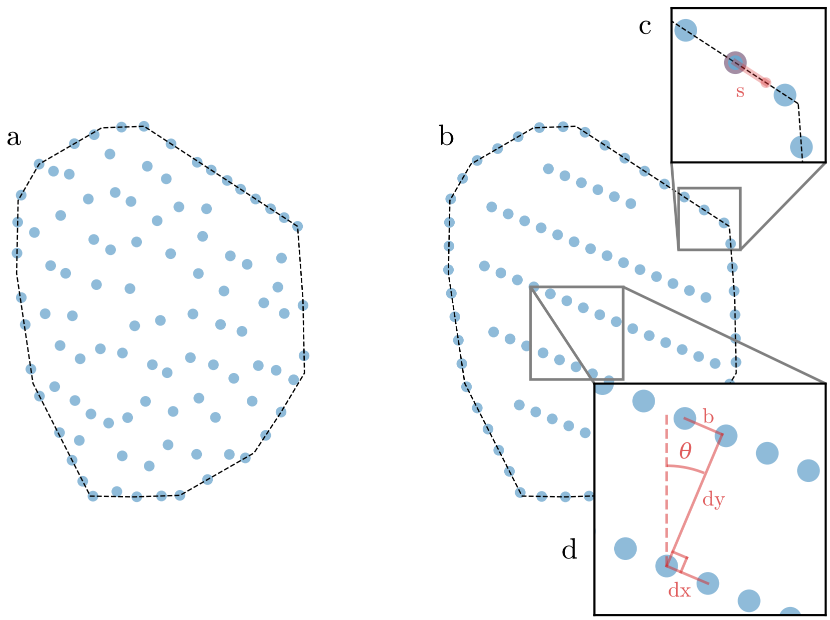

Figure 3. Example 100 turbine wind farm layouts, and parameterized wind turbine layout definition. Each dot is to scale, representing the wind turbine diameter. (a) Wind farm layout when the position of each turbine has been optimized directly. This optimization required 200 design variables, the x and y location of each turbine. (b) Wind farm layout optimized with boundary-grid parameterization. This optimization required five design variables, shown in (c) and (d). (c) The start location design variable, s. (d) The four variables defining the inner grid: the grid spacing, dx and dy, the grid offset b, and the rotation, θ.

grid. Note that the number of boundary turbines is determined before the number of turbines in the inner grid, to ensure that sufficient spacing in maintained between the boundary turbines.

The number of rows, columns, and their organization in the grid is determined with the following procedure. First, dy is set to be four times dx, b is set such that turbines are offset twenty degrees from those in adjacent rows, and θ is initialized randomly. Then, dx is varied with θ remaining constant, and dy and b changing to fulfill the requirements prescribed in the initialization definition, until the correct number of turbines are within the wind farm boundary. During optimization, each of the grid variables can change individually, however the discrete values remain fixed. For extremely small wind farms, with an average turbine spacing much less than 4 rotor diameters, it may be impossible to initialize the turbine rows with dy equal to be four times dx and meet the minimum spacing constraints. In this case, the discrete row variable initialization would need to be adjusted.

The process outlined to select the discrete variables used in the parameterization is recommended as a starting point, and when computational resources or time is limited. We tested many different methods of how to determine the discrete values, but found that the method shown above consistently produced wind farm layouts with high energy production. With sufficient resources, some scenarios may benefit from optimizing with a different ratio of boundary turbines, or different initializations of

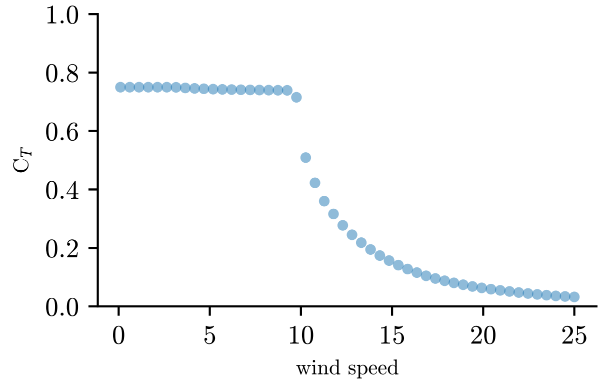

Figure 4. The thrust coefficient curve for the 3.35-MW turbine used in this paper.

the boundary grid. However, the results discussed in this paper were produced with the method given in this section. Because these variables are discrete, they cannot be included as design variables when using a gradient-based optimization method, because the function space would be discontinuous. But, a gradient-free optimization may benefit from including some of these discrete variables as design variables in the optimizations.

3 Wind Farm Modeling

3.1 Wind Turbine Parameters

In the testing of BG wind farm layout parameterization method, we modeled the turbine parameters after the IEA 3.35-MW reference turbine [2] ~~.~~ (Bortolotti et al., 2019). The relevant parameters are a rotor diameter of 130 meters, hub height of 110 meters, ~~and a rated electrical power of 3.35 MW~~a rated aerodynamic power of 3.6 MW, and a generator efficiency of 93%. The thrust coefficient ~~was assumed idealized and constant, $C_T = 8/9$.~~ curve for this turbine is shown in Fig. 4, and was generated using CCBlade, a blade element momentum code (Ning, 2013). The power curve was defined as a piecewise equation in Eq. 1.

$$P_i(V) = \begin{cases} 0 & V < V_{\text{cut-in}} \\ P_{\text{rated}}\left(\frac{V}{V_{\text{rated}}}\right)^3 & V_{\text{cut-in}} \leq V < V_{\text{rated}} \\ P_{\text{rated}} & V_{\text{rated}} \leq V < V_{\text{cut-out}} \\ 0 & V \geq V_{\text{cut-out}} \end{cases} \quad (1)$$

In this power curve definition, P_i is the aerodynamic power produced by an individual wind turbine, V is the hub velocity at that turbine (Lackner and Elkinton, 2007; Chen et al., 2015; Park and Law, 2015), P_{rated} is ~~3.35~~ 3.6 MW, V_{rated} is 10 m/s, $V_{\text{cut-in}}$

[2] ~~https://www.nrel.gov/wind/assets/pdfs/se17-9-iea-wind-task-37-systems-engineering.pdf~~

is 3 m/s, and $V_{\text{cut-out}}$ is 25 m/s. The aerodynamic power is then multiplied by the generator efficiency to calculate the electric power.

3.2 Wind Farm Details

The major benefit of wind turbine layout parameterization comes for large wind farms. For farms with just a few turbines, the layout can be optimized directly with a small amount of design variables. In such cases with few design variables, there is little to no benefit gained from intelligently parameterizing the design space. In this study, each wind farm layout that we optimized had 100 wind turbines, to demonstrate the benefits of BG parameterization for large wind farms.

We tested the performance of our parameterization method on wind farms with different average turbine spacing: four, six, and eight rotor diameters shown in Fig. 10. In addition to testing wind farms with different turbine spacing, we modeled and optimized several different wind farm boundaries in this study: the boundary of the Princess Amalia wind farm, a real farm in the North Sea (Van Dam et al., 2012; Gebraad and Van Wingerden, 2015; Kanev et al., 2018), a circle, and a square to demonstrate the sharp angles that can occur in wind farm boundaries. These boundaries are shown in Fig. 12.

3.3 Wake Model

We calculated the wind speed deficit in the wake behind a wind turbine with a simple

Wind speed deficits in this paper were predicted from turbine wakes with a modified version of the 2016 Bastankhah Gaussian wake model , modified from that originally developed by Bastankhah and Porté-Agel (Bastankhah and Porté-Agel, 2016; Baker et al., 2 . The (Bastankhah and Porté-Agel, 2016). The original formulation of the model does not define the wake deficit in the near wake region, creating undefined regions which make optimization difficult. To mitigate this issue, Thomas and Ning added a linear interpolation of the wake loss from the turbine up to where it is defined by the wake model, which is the version used in this paper (Thomas and Ning, 2018). The most important equation for this Gaussian wake model is defined with Eqs.?? and ??. shown in Eq. 2:

$$\frac{\Delta \bar{u}}{\bar{u}_\infty} = \left(1 - \sqrt{1 - \frac{C_T \cos\gamma}{8\sigma_y \sigma_z / d^2}}\right) \exp\left(-0.5\left(\frac{y-\delta}{\sigma_y}\right)^2\right) \exp\left(-0.5\left(\frac{z-z_h}{\sigma_z}\right)^2\right) \tag{2}$$

$$\sigma = kx + \frac{D}{\sqrt{8}}$$

$$l_i = \left(1 - \sqrt{1 - \frac{C_T}{8\sigma^2/D^2}}\right) \exp\left[\frac{-1}{2}\left(\frac{y}{\sigma}\right)^2\right]$$

In these equations, x is the distance between turbines for which the wakeloss is being calculated in line with the wind direction, y is the cross stream distance between turbines, D is where $\Delta\bar{u}/\bar{u}_\infty$ is the velocity deficit in the wake; C_T is the rotor diameter of the wind turbines, k is a parameter calculated from the turbulence intensity and is equal to 0.0325, C_T is the thrust coefficient, thrust coefficient; γ is the yaw angle, which is assumed to be zero throughout this paper; $y-\delta$ and l_i is the wake loss contribution from a single turbine. This model is a simplification of the model proposed by Bastankah and Porté-Agel by calculating all wake losses in the horizontal plane, $z-z_h$ are the distances from the wake center and the point of interest in the cross-stream horizontal and vertical directions, respectively; and σ_y and by assuming a potential core length of zeroσ_z are the standard deviations of the wake deficit, again in the cross-stream horizontal and vertical directions, respectively. These standard deviations are defined in Eqs. 3 and 4. The total loss at any point in the wind farm was given by the L2-norm of the individual loss contributions from each turbine shown in Eq. ??.

$$\sigma_y = k_y(x-x_0) + \frac{D\cos\gamma}{\sqrt{8}} \tag{3}$$

$$\sigma_z = k_z(x-x_0) + \frac{D}{\sqrt{8}} \tag{4}$$

where D is the diameter of the wind turbine creating the wake, $x-x_0$ is the distance downstream from the turbine to the point of interest, and k_y and k_z are unitless, and are functions of the freestream turbulence intensity:

$$L k_y, k_z = \sqrt{\left(\sum_{i=1}^{\text{nTurbines}} l_i^2\right)} 0.3837\ \text{TI} + 0.003678 \tag{5}$$

Finally, the Because $\gamma = 0$ throughout this paper, $\cos(\gamma) = 1$ meaning that $\sigma_y = \sigma_z$. Wakes were combined with a linear combination method, about which more details can be found in the cited literature (Bastankhah and Porté-Agel, 2016; Thomas and Ning, 20

.

To find the effective wind speed across the entire wind turbine to be used in turbine power calculation, we averaged the the velocities sampled at several points across the rotor. During optimization, we sampled at 4 points over the swept area of the rotor, shown in Fig. 5a. We have found that using just these four sampling locations gives almost identical effective velocity compared to using more sampling points. For the final evaluation, we sampled the wind speed at any point is expressed in Eq. ??. 100 points equally spread across the rotor swept area, shown in Fig. 5b.

$$V = U(1-L)$$

In this equation, V is the wind speed affected by wake losses, U is the free-stream wind speed , and L is the total wake loss.

3.4 Wind Resource

As the goal of this paper is to demonstrate the performance of our layout parameterization method in wind farm optimization for any scenario, we chose three different wind roses from cities in California, USA: North Island, Ukiah, and Victorville[2]. We During optimization, we divided the wind roses into 23 24 equal bins for each wind rose, with an associated directionally

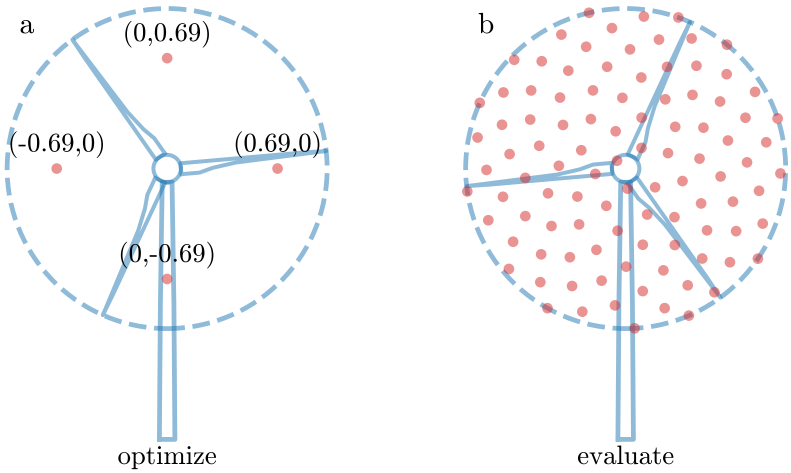

Figure 5. **The sampling points across the swept rotor area to calculate the effective wind speed at the turbine. Wind speeds are sampled at each point, and then averaged.** (a) The sparse sampling locations used during optimization. The coordinates shown are normalized by the rotor radius. (b) The 100 sample points used for final evaluation.

averaged wind speed, shown in Fig. 6. We have assumed that the wind speed distribution from each wind direction can be approximated with a Weibull distribution defined with the directionally averaged wind speeds (Fig. 7 and Eq. 6). Weibull distributions have been shown to be good representations of real wind speed data (Justus et al., 1978; Rehman et al., 1994; Seguro and Lambert, 2000).

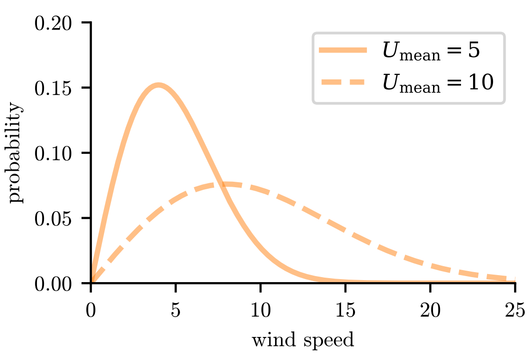

Figure 7. Example Weibull distributions for two different average wind speeds. Each wind direction is associated with an average wind speed (shown in Fig. 6), which is used for the value U_{mean}.

$$f(U,U_{\text{mean}}) = \frac{k}{U_{\text{mean}}}\frac{U}{U_{\text{mean}}}^{k-1} e^{-(U/U_{\text{mean}})^k}$$

$$f(U,\lambda,k) = \frac{k}{U_{\text{mean}}}\left(\frac{U}{U_{\text{mean}}}\right)$$

$$\lambda(U_{\text{mean}},k) = \frac{U_{\text{mean}}}{\Gamma(1+1/k)} \qquad (6)$$

In Eq. 6, f is the probability of wind for a given wind speed, U is any wind speed ~~, and~~ (non-negative), U_{mean} is the directionally averaged wind speed for the direction bin of interest, and Γ is the gamma function. The shape parameter, k, is assumed to be

[2]https://mesonet.agron.iastate.edu/sites/windrose.phtml?station=AAT&network=CA_ASOS

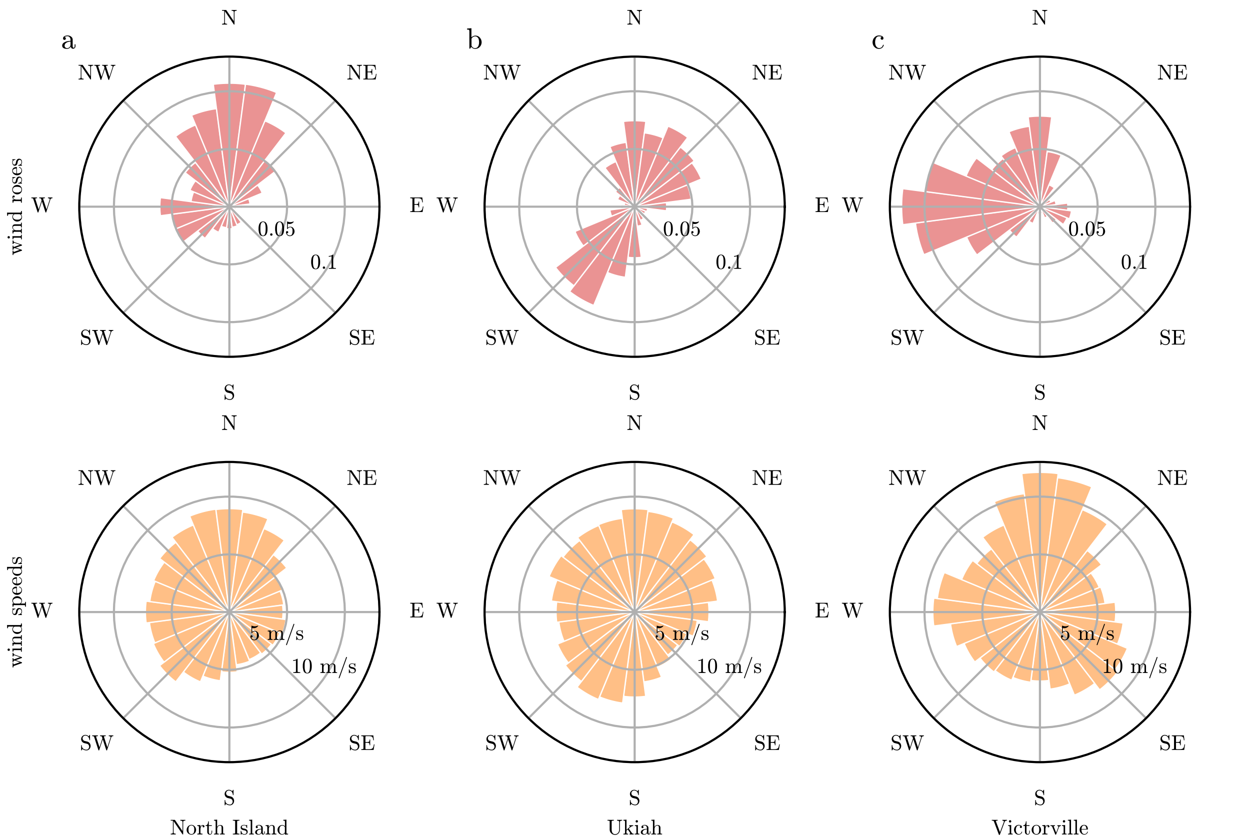

Figure 6. The three wind roses and associated average wind speeds used in this study. The wind resources are from (a) North Island, California, (b) Ukiah, California, and (c) Victorville, California.

equal to 2.0 for every wind direction, which is a realistic value for the Weibull distributions that represent real wind speed probability data (Rehman et al., 1994; Seguro and Lambert, 2000). For each wind direction, we have sampled the Weibull distribution at five equally spaced points during optimization. Five wind speed samples and ~~23~~ 24 wind direction samples are chosen as the sampling amount required to converge to the true wind farm production for a given wind farm (Stanley and Ning, 2019). Although the wind farms are optimized with the more coarse sampling of 24 wind directions and 5 wind speeds, the final wind farm layouts are evaluated with a more fine sampling of 360 wind directions and 50 wind speeds, to avoid the possibility of artificially inflated energy production due to coarse wind resource sampling.

4 Optimization

In this paper we compare how optimizing with BG wind farm layout parameterization compares to two common currently used parameterization methods. We have optimized wind farms using a simple grid parameterization (referred to as "grid

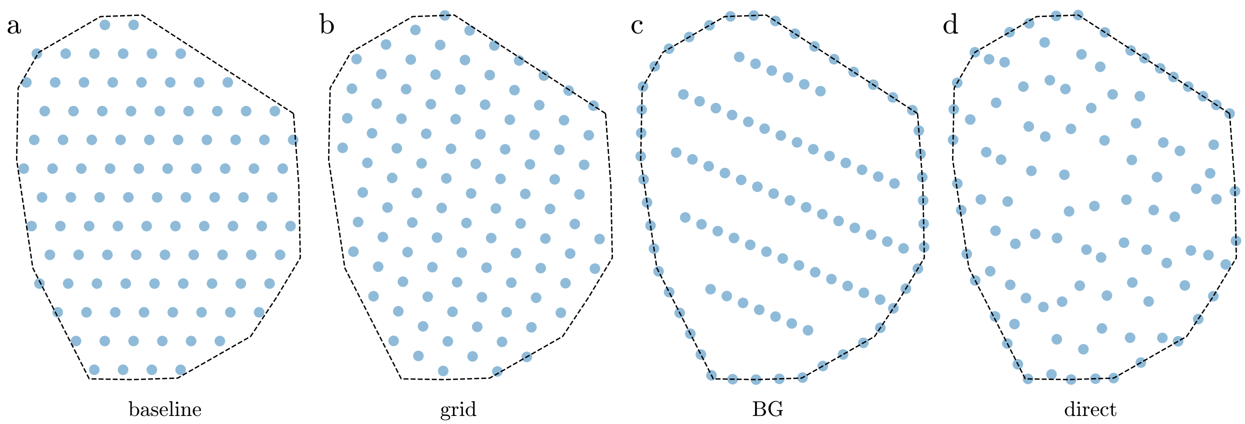

Figure 8. Example optimal layouts achieved with each parameterization method. These are 100 turbine layouts, with an average turbine spacing of four rotor diameters and the Princess Amalia wind farm boundary. They were optimized with the wind rose from North Island, California. (a) The baseline grid to which other methods were compared in Sect. 5.1. (b) An example optimized grid layout. (c) An example optimized boundary-grid layout. (d) An example layout that was optimized directly.

optimization"), BG parameterization ("BG"), and by directly optimizing the location of each turbine independently ("direct optimization"). Examples of these layouts, along with the baseline layout that was used to compare results in Sect. 5.1, are shown in Fig. 8.

In each case, the objective function of the optimization was to maximize the annual energy production (AEP) of the wind farm, shown in Eq. 7.

$$\mathrm{AEP} = 8760 \sum_{i=1}^{23} \sum_{j=1}^{5} P(\phi_i, U(\phi_i)_j) f_i f_j \tag{7}$$

In this equation, 8760 is the number of hours in a year, P is the wind farm power production, ϕ is the wind direction, V is the free-stream wind speed, f_i is the wind direction probability, and f_j is the wind speed probability. The design variables were determined by the optimization method that was used. For the grid optimization, the design variables were the grid spacing in the x and y directions, dx and dy, the grid offset b, and the grid rotation θ for a total of four variables. The discrete variables in the grid were determined with the same method described above to find the discrete variables in the grid portion of the BG parameterization, except ~~dx and dy were equal~~ $dy = dx$ or $dy = 2dx$ while determining the grid format. We experimented with different values of dy during grid initialization and found that ~~setting them equal~~ the the 1:1 or 1:2 ratios provided the best results. We ran every grid optimization with each initialization ratio, and chose the best results. The design variables for the BG optimization were the same as the grid optimization for the inner grid turbines, and an additional variable s defining the start location of the boundary turbines for a total of five design variables. For the direct optimization methods, the design variables were the x and y locations of each turbine in the wind farm for a total of 200 design variables. In each optimization, we applied turbine spacing constraints and boundary constraints. The turbine hub locations were constrained to not be within two rotor diameters of any other turbine hub. Additionally, the turbine hubs were constrained to be within the defined wind farm boundary. No bound constraints, or additional constraints were used to define where the turbines must lie. A link for the code used in this project is included at the end of this paper. Please refer to the code for specific details about how these constraints were enforced. This optimization is expressed in Eq. 8.

$$\begin{aligned}
&\text{maximize} && \text{AEP} \\
&\text{w.r.t.} && dx,\, dy,\, b,\, \theta && \text{(grid)} \\
& && dx,\, dy,\, b,\, \theta,\, s && \text{(BG)} \\
& && x_i,\, y_i\; (i=1,\ldots,100) && \text{(direct)} \\
&\text{subject to} && \text{boundary constraints} \\
& && \text{spacing constraints}
\end{aligned} \tag{8}$$

We used the optimizer SNOPT, which is a gradient-based optimizer that uses sequential quadratic programming, and is well suited for large-scale nonlinear problems such as the wind farm layout optimization problem (Gill et al., 2005). A challenge of gradient-based optimization is the tendency to converge to local solutions. In order to better search design space, we optimized the problem to convergence 100 times with randomly initialized design variables. The random initialization was performed by fully randomizing the rotation variable θ and the boundary start location s, and defining the discrete and other design variables as defined in Sec. 2.2. The design variables dx, dy, and b are then randomly perturbed by plus or minus 10%. This random initialization method allows the number of rows and columns in the inner grid to differ between optimization runs. This was done for each parameterization method, lending confidence that the best solution after optimizing the 100 random starts is near the global optimum. From the random starting points, we were also able to determine the spread of solutions obtained with each layout parameterization.

We used exact-analytic gradients in each optimization. The gradients for each portion of the model were obtained with an automatic differentiation source code transformation tool, Tapenade (Hascoet and Pascual, 2013). To combine the gradients to get the total derivative of the objective with respect to each of the design variables, we used the open-source optimization framework, OpenMDAO, which propagates the partial derivatives of each small section of the model and calculates the gradients of the entire system (Gray et al., 2010).

Using exact, rather than finite-difference, gradients is important in this study because the computational expense required for optimization problems with increasing design variables scales better with exact gradients (see Fig. 2). For the parameterized optimizations, the exact gradients were not as vital in terms of computational expense, but they were very important for the direct optimizations which had 200 design variables. In addition to reducing the function calls required to reach convergence, the exact gradients helped the optimizer converge to a better solution, avoiding many of the numerical difficulties that often plague the optimization process when using finite-difference gradients.

For this paper we have used only a gradient-based optimization method. The purpose of this research is to explore a novel wind turbine layout parameterization, and how it compares to other more commonly used layout parameterizations. We do not explore how different optimization methods compare when applied to the wind farm layout problem. As mentioned in the introduction, the relationship of how optimization method performance scales with increasing numbers of design variables is well documented. Additionally, our past work suggests that the large number of random starts allow for a reasonably thorough search of the design space.

5 Results and Discussion

In this section we demonstrate how the optimal wind farms using BG parameterization compared to wind farms that have been optimized directly, or with a common grid parameterization. We will discuss the best results, the computation expense required to optimize, and the multimodality of the design space with each parameterization method.

5.1 Best Results

Figure 9 shows the best results of the 100 random starts for each parameterization method, compared to a simple baseline grid (Fig. 8a). In Fig. 9, subfigures a, b, and c show results for varied turbine spacing, wind roses, and boundary shapes, respectively.

For each wind farm BG layout parameterization performs slightly better than the direct layout optimization, although all BG results are within 0.4% of the corresponding direct results. This does not mean that directly optimizing the layout of each turbine cannot perform as well as the BG parameterization. Clearly, with complete freedom of where to place each wind turbine, the optimizer could find the exact same layout as the BG layout. However, the complete freedom of the direct optimization means that the optimizer is free to explore many sub-optimal layouts as well, and will often converge in those areas. With BG parameterization, we have forced the turbines to only explore desirable turbine locations. For the scenarios that we explored, 100 BG optimizations produced a better result than 100 direct optimizations.

Figure 9a shows the optimal results for wind farms with varied average turbine spacing, with the North Island wind rose and Princess Amalia wind farm boundary. For ~~each wind farm, BG layout parameterization performs very well compared to the direct layout optimization results. The worst case, for the smallestwind farm with an average turbine spacing of four rotor diameters, is a mere 0.39% difference between the improvement from the direct optimization and the improvement from our parameterization. For the largest turbine spacing, an average spacing of eight rotor diameters, BG optimization actually performs slightly better than the direct optimization. For the smallest,~~ the smallest, most tightly packed wind farm, the optimized grid performs ~~significantly~~ better than the baseline, but underperforms by about ~~2~~2.3% compared to the other parameterization methods. Even at an average turbine spacing of six rotor diameters, the direct and parameterized optimizations perform about ~~0.7~~1% better than the grid optimization, which may or may not be significant depending on the uncertainty of the models used. For the largest wind farm, the optimal grid performs ~~almost identically to~~ within 0.4% of the other parameterization

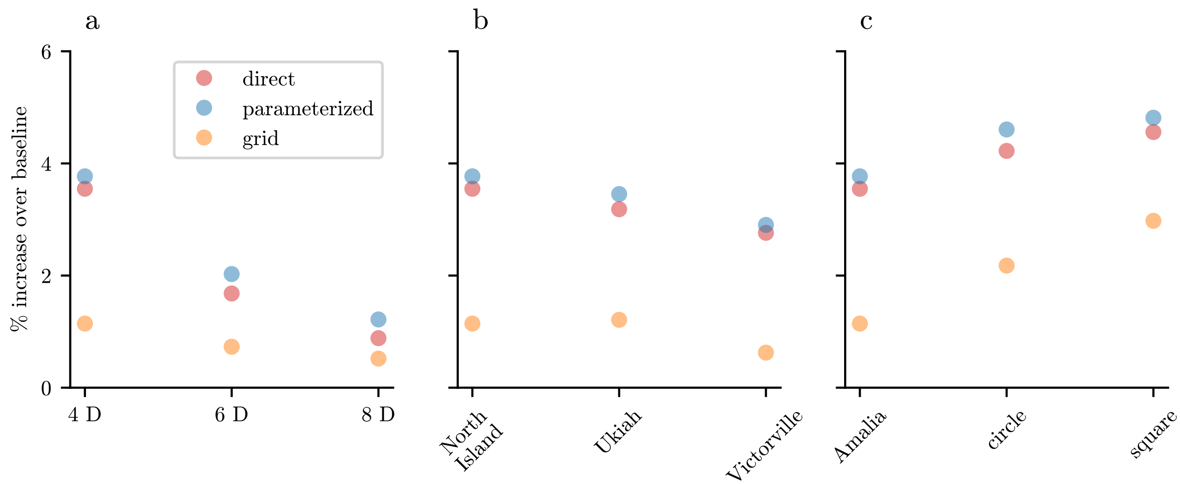

Figure 9. The best annual energy production achieved with 100 randomly initialized optimizations. Shown are the best results from the grid turbine parameterization (four design variables), our new boundary-grid parameterization method (five design variables), and by directly optimizing the location of each turbine (200 design variables). Results are shown as a percent increase over a baseline grid layout. (a) Varied average turbine spacing in the wind farm. (b) Varied wind rose. (c) Varied boundary shape.

methods. For large wind farms where the turbines are spaced very far apart, wakes are mostly recovered by the time they reach other turbines in the wind farm. In these cases, even an optimized grid performs almost as well as the direct or BG optimization.

Figure 9b shows results for optimized wind farms with different wind resources, with an average turbine spacing of four rotor diameters and the Princess Amalia wind farm boundary. The wind roses and the associated directionally averaged wind speeds are shown in Fig. 6. As with the varied turbine spacing results, the BG results are ~~almost identical to~~ slightly better than the direct optimizations~~. The largest difference occurred for the Ukiah wind rose, for which direct optimization performed 0.43%~~ , and much better than the ~~BG optimization. Additionally, for all wind resources, the BG optimizations perform significantly better than the simple gridoptimizations. For the North Island and Victorville roses, the BG layout optimization performs about 1.5% better than the grid parameterization, where for the Ukiah rose the parameterized is 3.5% better~~simple grid. For each wind rose, the grid achieves a slight improvement over the baseline, but underperforms by 2-2.3% compared to the direct and BG parameterizations.

Figure 9c shows the results for a varied wind farm boundary. The farms in this subfigure have an average turbine spacing of four rotor diameters and the North Island wind rose. Consistent with the previous results, the parameterized optimization performs superbly~~. The worst case difference is 0.39% compared to the direct optimization, which occurs for the most complex Amalia wind farm boundary. The parameterized optimization performs very slightly~~ , ~~0.02%, better than the direct optimization for the square wind farm boundary. For the circle and square boundaries, the optimized grid parameterization performs very well compared to the parameterized~~ always slightly outperforming the direct optimizations. In addition, we can see that the BG and direct optimizations ~~. The grid optimization for the circle boundary is only 0.43% worse than the~~

~~parameterized optimization and only 0.62% worse than the direct optimization. At least for the North Island wind rose, the grid parameterization may be sufficient for the simple , symmetric boundaries~~perform better than the more simple grid optimizations, by 1.5-2.3%.

In terms of the best achievable wind farms with each parameterization method, our new BG method performs almost identically to optimizing the location of each wind turbine directly. ~~Of all the cases~~ In all cases that we tested, the ~~largest difference between the BG and direct results was 0.43%, and in some cases the BG results were slightly better than the direct optimization~~BG optimizations we able to find solutions that slightly outperformed the direct optimizations, although they were almost identical. With only five design variables, we can create wind farms that perform the same as or better than farms that have been designed with 200 variables. While the grid parameterization is able to achieve good results for some wind farms, it often performs much worse than our parameterization. One additional variable is a small price to pay for significant improvement in optimal wind farm design.

5.2 Computational Expense

The utility of any wind farm layout parameterization is not only measured by the ability to create high energy producing wind farms, but by the ability to do so quickly and reliably. Figures 10, 11, and 12 are histograms showing optimal results and the computational expense required for each of the 100 optimizations run for each wind farm and parameterization method. In each figure, Subfigures a-c show the normalized optimal AEP for each of the 100 runs, and Subfigures d-f show the number of ~~function~~ wake model function calls required to converge to a solution. The AEP results have each been normalized by the maximum AEP achieved by the direct optimizations for the associated wind farm. Also note that the number of function calls are shown with a log scale.

In general, the grid and the BG optimal AEP results have a similar spread, with the BG results shifted up higher. Compared to the direct optimizations, the grid and BG optimizations have a larger spread in optimal solutions. This is a consequence of the discrete variables that are initialized at the start of each optimization run. The number of rows and columns, as well as their organization in the grid are determined by the randomly initialized rotation design variable, θ. Some of these grid formations are more desirable than others, leading to higher AEP values. This spread in optimal solutions is not a significant issue, because the number of functions calls required for the grid and BG optimizations are an order of magnitude lower than that required by the direct optimization. This allows for many randomly initiated runs in a short amount of time. If it did become an issue, the spread could be reduced by predefining the discrete grid variables, or including them as design variables in a gradient-free formulation. By showing the results for 3 different wind farm sizes, wind roses, and wind farm boundaries, we believe that our parameterization method can produce high AEP and optimize with reduced function calls for many scenarios.

With regards to the function calls required to converge, the grid optimizations required about one third of the function calls to converge compared to the BG optimizations, while the direct optimizations required about an order of magnitude more. The only exception was the circular wind farm, for which the direct optimizations converged quickly, on the same order as the BG optimizations. Function calls are an important measure of computational expense, as they are correlated with time and processing power required to optimize. Here it is important to remember that our results were obtained with exact-analytic gradients, meaning that one function call was required to obtain the wind farm AEP, as well as the gradients with respect to each of the design variables. The same is true of the constraints, one function call gave both the constraint values and the gradients. Without exact gradients, a finite-difference method would need to be used to calculate the gradients. At every optimization step, finite-difference gradients require one (forward or backward difference) or two (central difference) additional function calls for every design variable to approximate the gradients. Thus, if forward-difference gradients were used rather than exact, the grid optimizations would need about four times as many function calls to reach a solution, the BG optimization would need about five times as many function calls, and the direct optimization would need 200 times as many function calls to converge. This is the best case scenario, as optimizations with finite-difference gradients often have trouble converging. Compared to gradient-free optimization, the exact analytic gradients are vital. The direct optimization with a gradient-free technique would be near impossible because of the massive required computational expense (Ning and Petch, 2016; Thomas and Ning, 2018).

5.3 Multimodality

One of the major difficulties of the wind farm layout optimization problem is the extreme multimodality of the design space (Fig. 1). There can be thousands or even millions of local solutions, often varying drastically in their quality. Figure 13 shows one dimensional sweeps across the design variables, for each of the three different parameterization methods discussed in this paper. Because of the number of variables in this problem, it is difficult to fully represent the full design space graphically, however this figure is a good indicator of the multimodality of the different design spaces. Figures 13a, 13b, and 13c show the multimodality of the grid, BG, and direct layout parameterizations, respectively.

Parameterizing the design space with a grid and with the BG method (Figs. 13a and 13b) does not completely remove the multimodality of the wind farm layout problem. However, it does result in a smoother response and fewer local minima compared to the design space when each of the turbines are optimized directly. These function spaces can be explored easily with a few random starting locations, or with a gradient-free optimization method. The design space when varying the location of individual turbines (Figs. 1 and 13c) is much more noisy, filled with comparatively larger peaks and valleys in the design space. These figures only show the design space with respect to the location of one turbine, which is defined with two variables. The full space consists of the location of all 100 turbines, or 200 variables, for which the multimodality and overall noisiness of the design space is exacerbated. Figures 13a and 13b do not show the function space with respect to the discrete grid variables. Even so, considering each combination of the feasible grid variables is more desirable than the difficulty involved with the 200-dimensional function space of the direct layout definition.

Notice that the ranges of the design variable sweeps is different for the BG and grid parameterizations compared to the direct sweep. This is because the more simple parameterizations are more limited in the feasible design values. The range through which the design variables can sweep is relatively limited, without violating the minimum spacing or the boundary constraints.

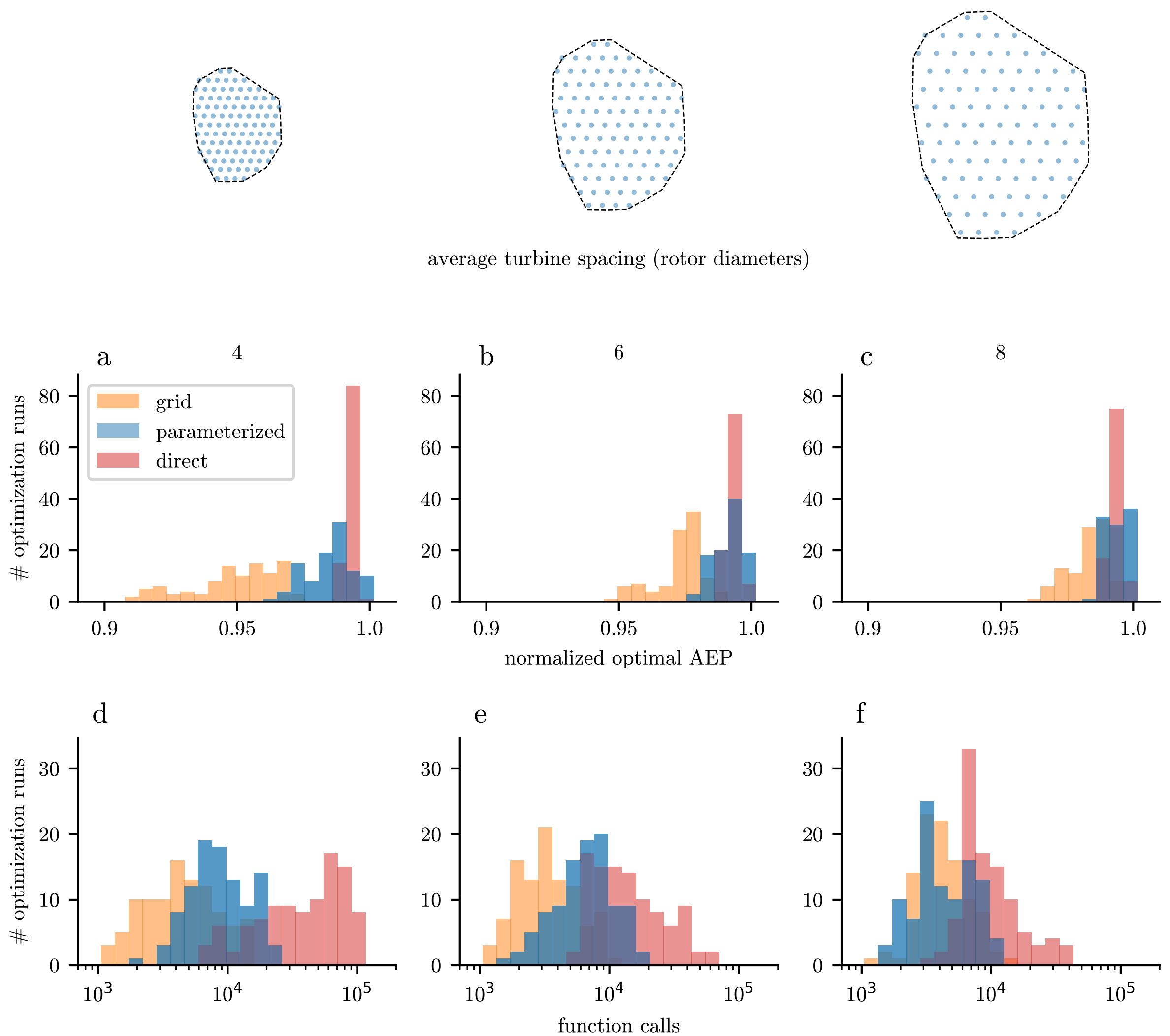

Figure 10. Results from 100 randomly initialized optimizations for wind farms with varied average turbine spacing and 100 wind turbines. The farm optimized had the Princess Amalia boundary, and the wind rose from North Island, California. Shown are results using the grid turbine parameterization, our new boundary-grid parameterization, and by direct optimization. The optimal annual energy production distribution achieved for each of the optimization runs, in wind farms with varied turbine spacing of 4, 6, and 8 rotor diameters for subfigures (a), (b), and (c), respectively. The number of function calls required to converge for each of the optimization runs, in wind farms with varied turbine spacing of 4, 6, and 8 rotor diameters for subfigures (d), (e), and (f), respectively.

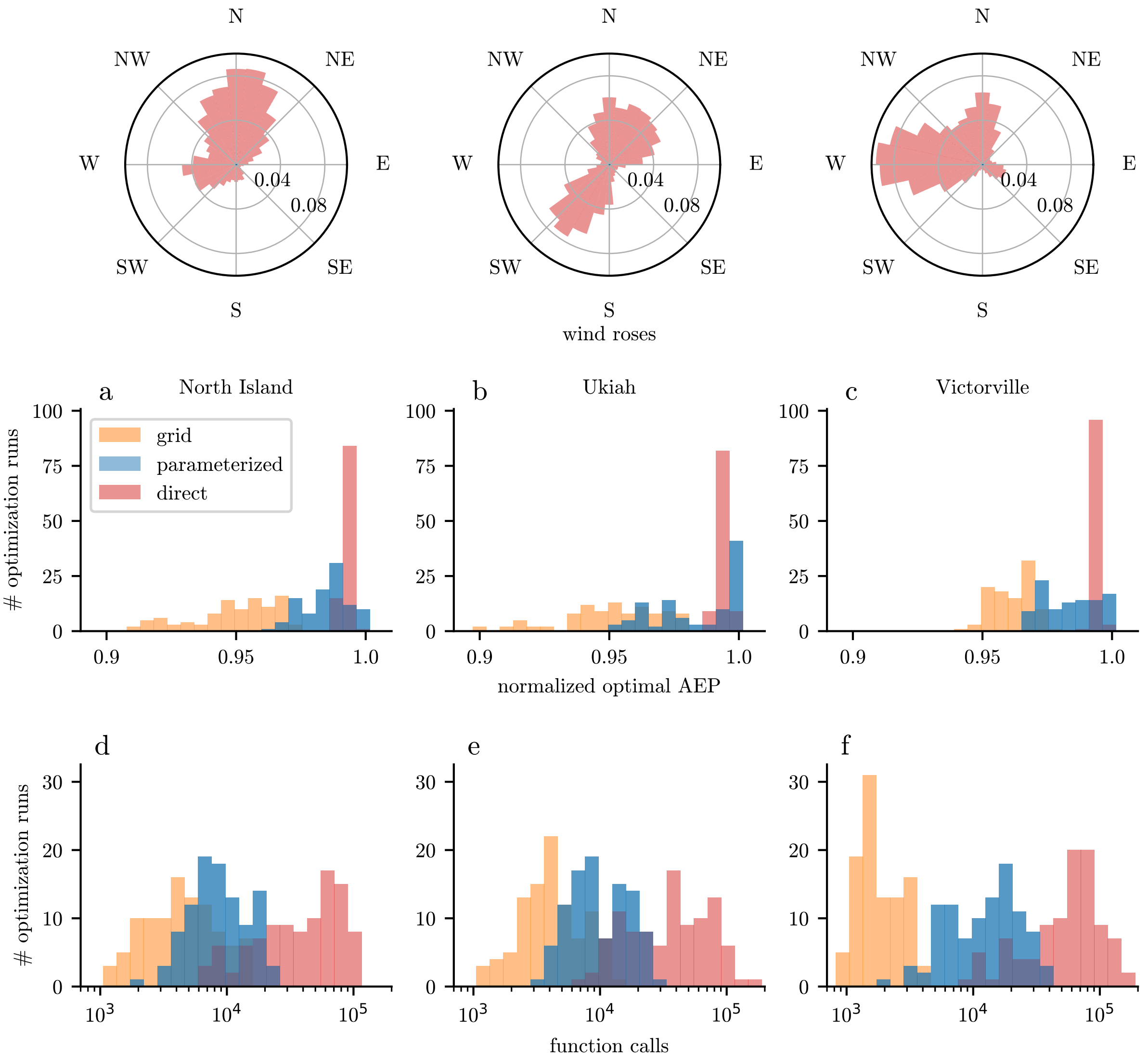

Figure 11. Results from 100 randomly initialize optimizations for wind farms with varied wind roses and 100 wind turbines. The farm optimized had the Princess Amalia boundary, and the average turbine spacing was four rotor diameters. Shown are results using the grid turbine parameterization, our new boundary-grid parameterization, and by direct optimization. The optimal annual energy production distribution achieved for each of the optimization runs, in wind farms with varied wind roses. Wind rose from (a) North Island, California, (b) Ukiah, California and (c) Victorville, California. The number of function calls required to converge for each of the optimization runs, in wind farms with varied wind roses. Wind Rose from (d) North Island, California, (e) Ukiah, California and (f) Victorville, California.

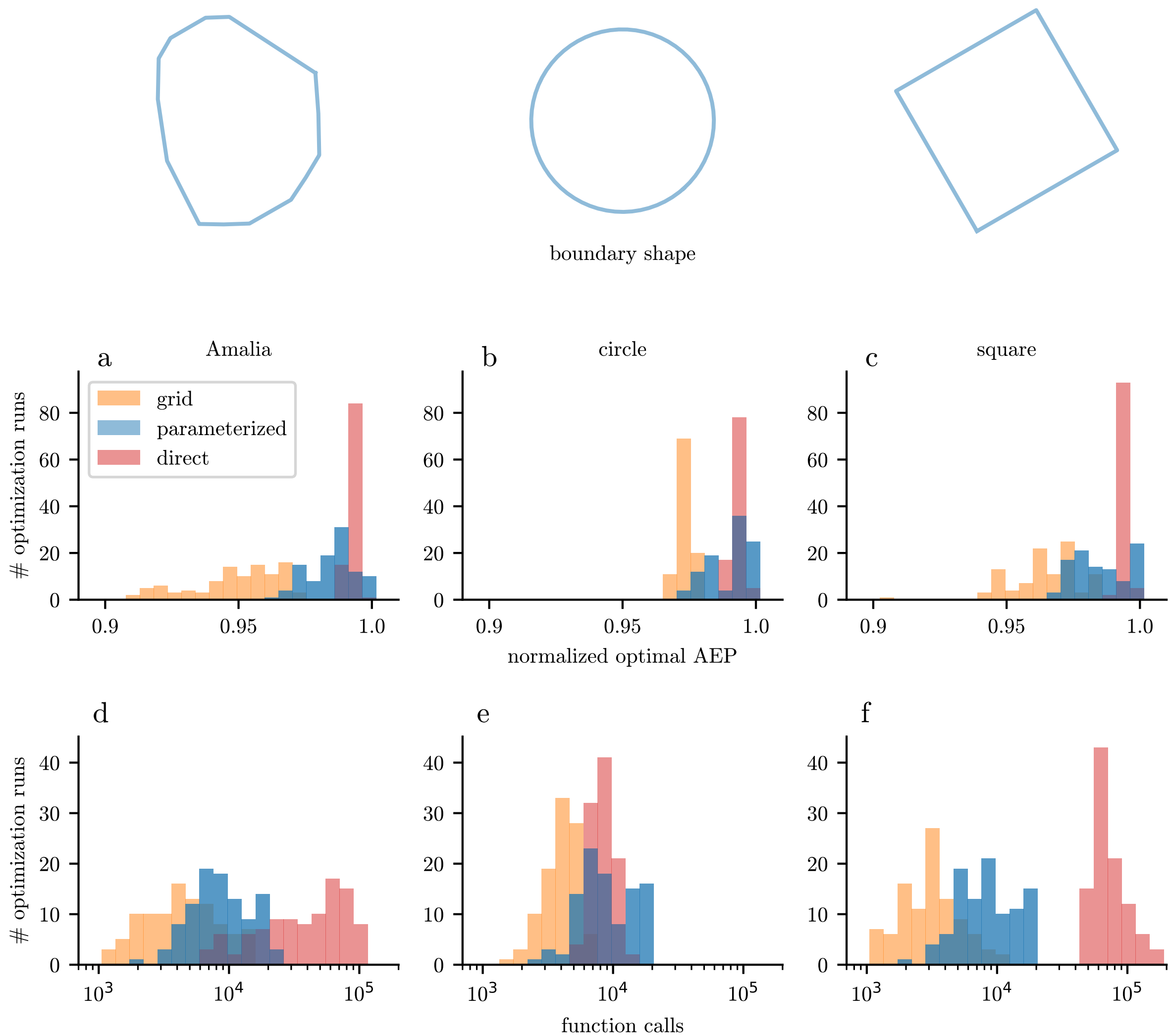

Figure 12. Results from 100 randomly initialize optimizations for wind farms with varied wind farm boundaries and 100 wind turbines. The average turbine spacing was four rotor diameters, and the wind rose was from North Island, California. Shown are results using the grid turbine parameterization, our new boundary-grid parameterization, and by direct optimization. The optimal annual energy production distribution achieved for each of the optimization runs, in wind farms with varied boundary shapes. (a) Princess Amalia wind farm boundary. (b) Circular wind farm. (c) Square wind farm. The number of function calls required to converge for each of the optimization runs, in wind farms with varied boundary shapes. (d) Princess Amalia wind farm boundary. (e) Circular wind farm. (f) Square wind farm.

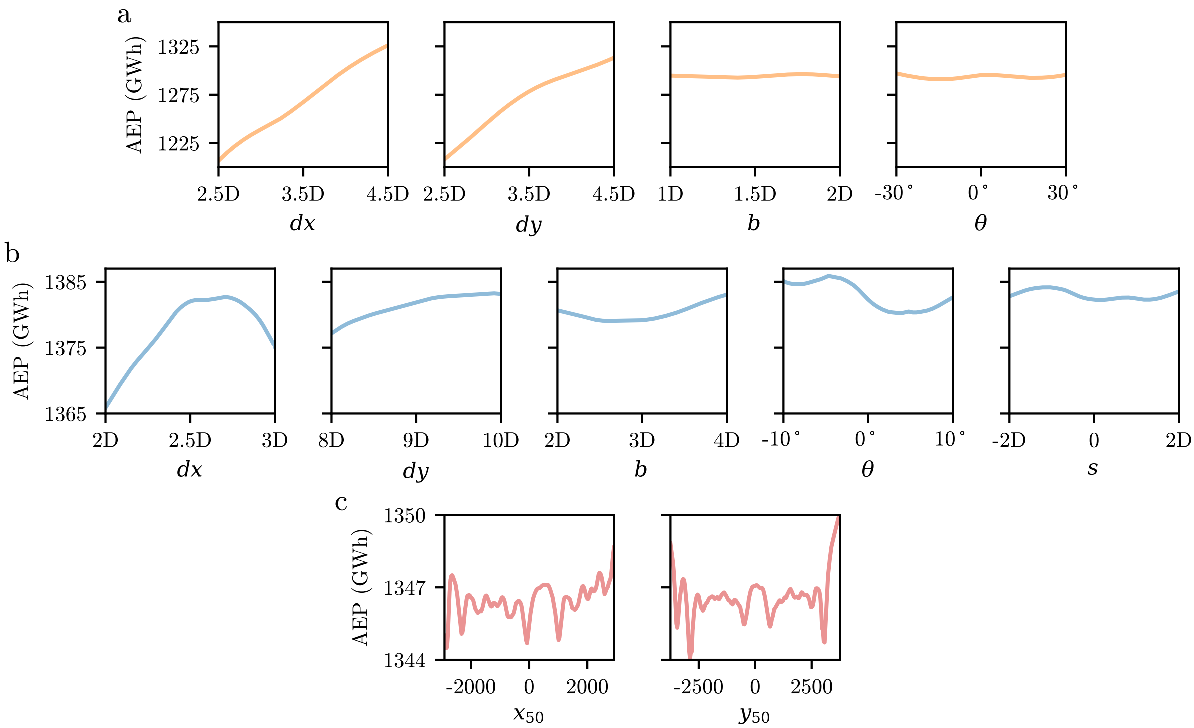

Figure 13. One dimensional sweeps across the design space of each parameterization method discussed in this paper. These figures show the multimodality of each of the design spaces. (a) The simple grid parameterization. (b) Our newly presented boundary-grid parameterization. (c) Moving the location of one wind turbine across the wind farm in x and y (refer to Fig. 1). With the direct turbine layout definition there are actually 200 variables. This figure shows the multimodality in just 2 of these variables, where the whole design space is much more complex.

6 Additional Details on BG Parameterization

BG parameterization requires few variables, produces wind farm layouts that perform similarly to ones that have been optimized directly with much lower computational expense, and reduces the multimodality of the design space. In addition, there are some innate design characteristics that are useful in wind farm design. First, the layouts produced are regular, aesthetically pleasing patterns. To the untrained eye, BG parameterization looks well designed compared to the seemingly random layouts that are often produced when every turbine location is optimized individually. This can play an important role in the public perception of large scale wind energy. Second, BG parameterization has clear roads or shipping lanes naturally built into the design. Roads and shipping lanes are requirements in wind farm design that are often neglected in research studies. ~~Finally, the layouts produced have a clear cabling pattern. Wind farm cabling is an expensive and complex part of wind farmdesign. With~~

~~BG parameterization, cabling requirements can be clearly minimized by running cables across each of the rows, and around the boundary without the need for complex cabling algorithms~~

Often, there are prohibited areas within a wind farm. This could be for many reasons, such as natural geography, roads or shipping lanes, or a variety of other reasons. Although beyond the scope of this paper, and not addressed in the results shown in Sect. 5, we have a few ideas on how this would be handled with BG parameterization. Many prohibited zones, such as shipping lanes, roads, or cable lines, are easily managed with a grid turbine layout, as these could easily be designed to follow the existing grid layout. Other prohibited zones could be handled by the BG parameterization, with no adjustments. This would be for cases where the prohibited zones are relatively small. For other cases, where the prohibited zones are larger and more restrictive, slight modifications would need to be made to the parameterization. The discrete variable of the inner grid would be initially defined such that the turbine location constraints are met. This would likely include some of the rows are not continuous, but have some gaps to accommodate the constraints. Likewise, the boundary turbines would be defined slightly differently, in that there would be some gaps to accommodate layout constraints.

7 Conclusions

In this paper, we have presented the new boundary-grid wind farm layout parameterization method. This method uses only five design variables, regardless of the number of wind turbines, but is capable of producing turbine layouts that perform just as well ~~to~~ as or better than layouts where the location of each wind turbine has been optimized directly. ~~Of all the cases that we tested, the largest difference in AEP improvement over a common baseline between BG layout optimization and a direct layout optimization was 0.43%, and in a few cases BG parameterizationeven performed better. BG parameterization also performed better than a simple grid parameterization in all cases tested~~We optimized the layout of 7 different wind farms with 3 different parameterization methods: a simple grid, directly optimizing the location of each turbine, and our new Boundary-Grid parameterization. For each wind farm and parameterization method, we ran 100 optimizations with randomly initialized design variables. In every case, the the best layout achieved with the BG parameterization perform slightly better than the best layout achieved with the direct optimizations.

In addition to being able to match the optimal energy production of wind farms that were directly optimized, BG parameterization requires an order of magnitude fewer function calls to reach a solution. This is with exact-analytic gradients, which means if finite-difference gradients or a gradient-free optimization method were used instead, our parameterization method would require at least two to three orders of magnitude fewer function calls to optimize. BG parameterization also reduces the multimodality of the design space, simplifying the optimization process and making it easier to find a good solution.

The BG layout definition places a portion of the wind turbines around the boundary, spaced equally traversing the wind farm perimeter. The rest of the turbines are placed in a grid inside the farm boundaries. The wind farm layouts created have a regular, aesthetically pleasing pattern, naturally defined roads and shipping lanes, and an easily defined cabling pattern. BG parameterizations solves many of the problems that typically accompany wind farm layout optimization. It is a simple, easily implemented technique that can immediately be applied by researchers and wind farm developers, playing and important role in continued growth of wind energy.

Code and data availability

The code written for this paper is included here: DOI: 10.5281/zenodo~~.3261037~~.3523383. All dependencies, with the exception of the optimizer SNOPT are open source.

Author Contributions

APJS performed every part of this research and is the primary author. AN helped develop ideas and direction, provided feedback, and provided editing for the paper.

Competing Interests

The authors declare no competing interests.

References

Baker, N. F., Stanley, A. P. J., Thomas, J. J., Ning, A., and Dykes, K.: Best Practices for Wake Model and Optimization Algorithm Selection in Wind Farm Layout Optimization, in: AIAA Scitech 2019 Forum, San Diego, CA, https://doi.org/10.2514/6.2019-0540, 2019.

Barthelmie, R. J., Frandsen, S. T., Nielsen, M., Pryor, S., Rethore, P.-E., and Jørgensen, H. E.: Modelling and measurements of power losses and turbulence intensity in wind turbine wakes at Middelgrunden offshore wind farm, Wind Energy, 10, 517–528, 2007.

Barthelmie, R. J., Hansen, K., Frandsen, S. T., Rathmann, O., Schepers, J., Schlez, W., Phillips, J., Rados, K., Zervos, A., Politis, E., et al.: Modelling and measuring flow and wind turbine wakes in large wind farms offshore, Wind Energy, 12, 431–444, 2009.

Bastankhah, M. and Porté-Agel, F.: Experimental and theoretical study of wind turbine wakes in yawed conditions, Journal of Fluid Mechanics, 806, 506–541, 2016.

Bortolotti, P., Tarres, H. C., Dykes, K. L., Merz, K., Sethuraman, L., Verelst, D., and Zahle, F.: IEA Wind TCP Task 37: Systems Engineering in Wind Energy - WP2.1 Reference Wind Turbines, https://doi.org/10.2172/1529216, 2019.

Briggs, K.: Navigating the complexities of wake losses, Norik American Windpower, 10, 2013.

Chen, Y., Li, H., He, B., Wang, P., and Jin, K.: Multi-objective genetic algorithm based innovative wind farm layout optimization method, Energy Conversion and Management, 105, 1318–1327, 2015.

Emami, A. and Noghreh, P.: New approach on optimization in placement of wind turbines within wind farm by genetic algorithms, Renewable Energy, 35, 1559–1564, 2010.

Feng, J. and Shen, W. Z.: Solving the wind farm layout optimization problem using random search algorithm, Renewable Energy, 78, 182–192, 2015.

Gebraad, P. and Van Wingerden, J.: Maximum power-point tracking control for wind farms, Wind Energy, 18, 429–447, 2015.

Gebraad, P., Thomas, J. J., Ning, A., Fleming, P., and Dykes, K.: Maximization of the Annual Energy Production of Wind Power Plants by Optimization of Layout and Yaw-Based Wake Control, Wind Energy, 20, 97–107, https://doi.org/10.1002/we.1993, 2017.

Gill, P. E., Murray, W., and Saunders, M. A.: SNOPT: An SQP algorithm for large-scale constrained optimization, SIAM review, 47, 99–131, 2005.

González, J. S., García, Á. L. T., Payán, M. B., Santos, J. R., and Rodríguez, Á. G. G.: Optimal wind-turbine micro-siting of offshore wind farms: A grid-like layout approach, Applied energy, 200, 28–38, 2017.

Gray, J., Moore, K., and Naylor, B.: OpenMDAO: An open source framework for multidisciplinary analysis and optimization, in: 13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference, p. 9101, 2010.

Guirguis, D., Romero, D. A., and Amon, C. H.: Toward efficient optimization of wind farm layouts: Utilizing exact gradient information, Applied energy, 179, 110–123, 2016.

Hascoet, L. and Pascual, V.: The Tapenade automatic differentiation tool: Principles, model, and specification, ACM Transactions on Mathematical Software (TOMS), 39, 20, 2013.

Herbert-Acero, J., Probst, O., Réthoré, P.-E., Larsen, G., and Castillo-Villar, K.: A review of methodological approaches for the design and optimization of wind farms, Energies, 7, 6930–7016, 2014.

Ju, X. and Liu, F.: Wind farm layout optimization using self-informed genetic algorithm with information guided exploitation, Applied Energy, 248, 429–445, 2019.

Justus, C., Hargraves, W., Mikhail, A., and Graber, D.: Methods for estimating wind speed frequency distributions, Journal of applied meteorology, 17, 350–353, 1978.

Kanev, S., Savenije, F., and Engels, W.: Active wake control: An approach to optimize the lifetime operation of wind farms, Wind Energy, 21, 488–501, 2018.

Lackner, M. A. and Elkinton, C. N.: An analytical framework for offshore wind farm layout optimization, Wind Engineering, 31, 17–31, 2007.

Lyu, Zhoujie, X. Z. and Martins, J. R. R. A.: Benchmarking optimization algorithms for wing aerodynamic design optimization, Proceedings of the 8th International Conference on Computational Fluid Dynamics, 11, 2014.

Neubert, A., Shah, A., and Schlez, W.: Maximum yield from symmetrical wind farm layouts, in: Proceedings of DEWEK, 2010.

Ning, A. and Petch, D.: Integrated Design of Downwind Land-Based Wind Turbines Using Analytic Gradients, Wind Energy, 19, 2137–2152, https://doi.org/10.1002/we.1972, 2016.

Ning, S.: CCBlade documentation, National Renewable Energy Laboratory, 2013.

Parada, L., Herrera, C., Flores, P., and Parada, V.: Wind farm layout optimization using a Gaussian-based wake model, Renewable energy, 107, 531–541, 2017.

Park, J. and Law, K. H.: Layout optimization for maximizing wind farm power production using sequential convex programming, Applied Energy, 151, 320–334, 2015.

Perez-Moreno, S. S., Dykes, K., Merz, K. O., and Zaaijer, M. B.: Multidisciplinary design analysis and optimisation of a reference offshore wind plant, in: Journal of Physics: Conference Series, vol. 1037, p. 042004, IOP Publishing, 2018.

Rehman, S., Halawani, T., and Husain, T.: Weibull parameters for wind speed distribution in Saudi Arabia, Solar Energy, 53, 473–479, 1994.

Rios, L. M. and Sahinidis, N. V.: Derivative-Free Optimization: a Review of Algorithms and Comparison of Software Implementations, Journal of Global Optimization, 56, https://doi.org/doi:10.1007/s10898-012-9951-y, 2013.

Rosenbrock, H.: An automatic method for finding the greatest or least value of a function, The Computer Journal, 3, 175–184, 1960.

Seguro, J. and Lambert, T.: Modern estimation of the parameters of the Weibull wind speed distribution for wind energy analysis, Journal of wind engineering and industrial aerodynamics, 85, 75–84, 2000.

Singg, David W., N. M. and Pulliam, T. H.: A comparative evaluation of genetic and gradient-based algorithms applied to aerodynamic optimization, European Journal of Computational Mechanics, 17, 103–126, 2008.

Stanley, A. P. and Ning, A.: Coupled Wind Turbine Design and Layout Optimization with Non-Homogeneous Wind Turbines, Wind Energy Science, 4, 99–114, https://doi.org/10.5194/wes-4-99-2019, 2019.

Stanley, A. P. J., Ning, A., and Dykes, K.: Optimization of Turbine Design in Wind Farms with Multiple Hub Heights, Using Exact Analytic Gradients and Structural Constraints, Wind Energy, 22, 605–619, https://doi.org/10.1002/we.2310, 2019.

Thomas, J. J. and Ning, A.: A Method for Reducing Multi-Modality in the Wind Farm Layout Optimization Problem, in: Journal of Physics: Conference Series, vol. 1037, The Science of Making Torque from Wind, Milano, Italy, https://doi.org/10.1088/1742-6596/1037/4/042012, 2018.

U.S. Energy Information Administration: Short-term energy outlook, Department of Energy, pp. 1–49, 2019a.

U.S. Energy Information Administration: Annual energy outlook 2019 with projections to 2050, Department of Energy, pp. 1–83, 2019b.

Van Dam, F., Gebraad, P., and van Wingerden, J.-W.: A maximum power point tracking approach for wind farm control, Proceedings of The Science of Making Torque from Wind, 2012.