Wind Energ. Sci. Discuss.,
https://doi.org/10.5194/wes-2019-37-AC1, 2019

WIND
ENERGY
SCIENCE
DISCUSSIONS

Interactive comment on "Massive Simplification of the Wind Farm Layout Optimization Problem" *by* Andrew P. J. Stanley and Andrew Ning

Andrew P. J. Stanley and Andrew Ning

stanley_andrewpj@yahoo.com

Received and published: 2 October 2019

Response to Reviewer 1

Andrew PJ Stanley and Andrew Ning

October 2, 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 suboptimal 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, the 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.