Response to RC1

Andrew P. J. Stanley, Christopher Bay, and Paul Fleming

May 2023

Before we proceed with our response to your comments, we want to express sincere gratitude for your time and energy spent acting as a reviewer for this paper. We recognize that you have taken time out of a busy schedule to help our work to be more clear and impactful, and we appreciate your contribution.

We have incorporated your feedback into our revised paper. Below is a summary of your points in blue, and our response and changes in the paper immediately following shown in black.

1 General comments

Summary: Layout design and wake steering (achieved through yaw control) is combined in a co-design process that does not exceed the computational cost of a traditional layout optimization by determining yaw angles from the layout. The work is well-placed in context to related literature and clearly motivated.

The reported 0.8% increase in AEP is indeed a very interesting result – but how generally applicable is it? In this paper it is based on one comparison. It would be interesting to have further comments on this and the limitations of their work from the authors.

Thank you for this feedback, it is something we fully agree with and think further discussion on this matter will be a significant improvement on the preprint version. In the revised paper, we will add discussion about our example layout optimization results. Specifically, we will discuss the potential sensitivity of the results to the various problem parameters and in what situations we expect this method to perform particularly well, and where it might be less impactful.

All in all a useful, very well-written and interesting study.

Thank you! We are very excited about this work, and the implications it may have to other aspects of wind farm codesign.

2 Specific comments

2.1 Section 1. Introduction

p. 3. Fig. 1: As a reader, it would be helpful with additional information in order to better relate to the stated computation time for an optimization. What are the computational resources used (e.g., amount and type of CPUs)? How well are these optimization problems converged (in SNOPT that would be the Major Optimality Tolerance)? How are the gradients computed (further down on p. 9 finite difference is mentioned but that is for the SciPy setup)? The information cannot fit into a single figure caption so one idea could also be to fit them into the Section 2 on the Geometric Yaw Relationship where SNOPT is mentioned. Still, without more information it is difficult for a reader to relate to the reported computation time.

We agree this is important information to include. In the revised manuscript, we will add discussion about the computational resources used for the results in the paper, as well as information about the problem setup.

p. 3. Fig. 1: Why are there three dots on the curve?

The three dots represent the three fully coupled layout and yaw optimizations that we ran to understand this computational expense relationship. We will clarify this in the revised paper.

---

p. 3: 'Revolutionize' is as a word brought up three times in this paper. Abstract, here, and finally in the Conclusions (where it says 'can revolutionize' and not will revolutionize). Indeed, the method is clearly an improvement – but the 0.8% presented further down is based on one comparison. One could perhaps modify the wording slightly here – or to build confidence of the significance for the findings presented in this study - somewhere include comments of how generally applicable these results are. What about some of the factors not taking into considerations (comparing orders of magnitude in effect)?

This is a great point. We were bold with this language, perhaps without providing sufficient evidence to justify it. In the revised draft we will incorporate your feedback, to modify wording or better build confidence of the significance of these findings.

2.2 Section 2. Geometric Yaw Relationship

p. 4: See comment on Fig 1 above. It would be interesting to have more information on the optimizations.

Agreed, in the revision we will provide more detailed information.

---

p. 4 On the geometric yaw relationship: It is based on the (very illustrative) plot in Fig. 2C. Still, the resulting yaw relationship in Fig 2.D seems 'manually' constructed (i.e., "defined through observation of the pattern"). Why not simply interpolate using e.g., a piecewise linear model or perhaps a curvature-minimizing interpolant in 2D? Could the authors' reasoning be, that these potential improvements would only lead to minimal extra gain? This could be further elaborated.

We agree, this needs to be more clear. You are correct that the relationship used for the results in this paper was manually constructed. This was intentional to illustrate that significant improvements can be achieved with such a simple relationship. This will be clarified in the revision. Additionally we will add discussion that this is a specific area for improvement as we develop this method further.

2.3 Section 3: Results

p. 6: suggestion: SciPy instead of scipy.

This will be corrected in the revised manuscript.

---

p. 6: Spelling typo: SLSQP gradient-based optim[i]zer

This will be corrected in the revised manuscript.

---

p. 6: What was the reason for changing optimization setup from SNOPT to SciPy?

This change in optimizer was purely a logistical one. The lead author who ran the optimizations changed organizations shortly before running them, and did not have access to SNOPT in the new organization. We don't think this is necessary information to include in the revision, although we are happy to add some explanation if you think it is important.

---

p. 6: Was it necessary to add the final continuous yaw optimization in the second example where the formulation leveraging geometric yaw relationship is tested? It would seem, that the ideal incorporation of the geometric yaw in the optimization procedure would not need a final yaw optimization at the very end. How much improvement is gained from this extra optimization? Could this extra step be avoided somehow (e.g., through a more accurate geometric yaw relationship based on interpolation)? More comments on this would be interesting.

This is a great comment. In short, yes the final continuous yaw optimization is necessary. The geometric yaw relationship underperforms the continuous yaw optimization fairly significantly in the current form. This information in itself is quite interesting however. The geometric yaw relationship does not need to perform very closely to a fully continuous optimization! It just needs to perform well enough that when used in a codesign framework, wake steering is sufficiently accounted for during the layout optimization and a better solution is achieved. In the revision, we will add discussion on this matter, and also clarify that an improved geometric yaw relationship may enable us to find even better solutions, and remove the requirement of a final continuous yaw optimization step.

---

p. 9: Spelling typo: 0.8% higher AEP [that] to 0.8% higher AEP [than].

This will be corrected in the revised manuscript.

---

p. 9: When comparing the final layout result for the two optimization strategies (seen in Fig. 4C-D) it would be interesting to also see the starting position of the two layouts. This is relevant since the final part of the Results section speaks to the disparate final layouts obtained with the two methods. This is indeed a clear tendency also seen from the illustrative 1-D example. Still, a contributing factor not mentioned could also be, that the two strategies started from different baseline layouts (each method could start from any of the 50 candidate layouts). This could be further clarified in the text.

Thank you for this comment, we agree this is important to clarify and discuss better. The 50 starting layouts were generated in a preprocessing step. Then, each of the 50 baseline layouts was optimized with both methods (with and without the geometric yaw relationship). In short, each method had the same set of starting conditions. Regardless, this is critical to explain better, and we will clarify this in the revision.

---

p. 9: Further information on the two final optimizations being compared would be of interest. How many function calls for each optimization? How well were the optimization problems converged? Comments on the computational resources, computation time used, etc.

Thank you for this comment, we will add discussion about the computational resources, computational expense, convergence, and other information about the optimizations in the revision.

2.4 Section 4: Conclusions

p. 9: New information concerning number of function calls is introduced for the first time in this section. Ideally, this point should be introduced and clarified further already in the Results section before being mentioned in the conclusion. Other than that it is a nice, succinct conclusion.

Thank you! And yes, as mentioned in the previous comment we will add discussion about computational expense in the results section.