Answer to reviews: Integer programming for optimal yaw control of wind farms

We would like to thank both reviewers for their detailed and constructive remarks. We will revise the manuscript carefully to accommodate as many suggestions and requests as possible, while keeping an eye on the page count. Please see below for our detailed replies to the review reports.

Answers to Review 1

- "In Sec. 2.3.3 it is mentioned that this new formulation of the yaw optimization problem based on this "covering approach" is equivalent..."—We will rephrase the sentence to make clear that a marginal difference is possible.
- "Moreover, it would be nice to highlight qualitatively or quantitatively the advantages..."—There is an analogous recommendation in Review 2. Thus, we will incorporate the following comparison computations: First, we will use our covering approach but with the software FLORIS as simulation software, second, we will use FLORIS to compute the direct problem (i.e., full enumeration) in (at least) one case, and third, we use the serial-refine method in FLORIS.
- "Other minor comments:"—We will include the suggestions (vector for wind direction; summarizing table in Sect. 4.4; use "cross-reference" in the text; briefly mention (the most important) mild assumptions in Abstract and Conclusion)

Answers to Major Comments of Review 2

- 1. a. Reproducibility:

    i. "There is insufficient description of the overall process, and the paper would benefit from a flow-chart or pseudo-code of the entire process."—We will improve the description and try (space allowing) to include a flow-chart or pseudo-code.

    ii. "How is the constraint of having consistent yaw offset angles..."—In practice, it is done in two steps as described in the manuscript. First, we compute the valid yaw configurations $\tilde{L}_{k+1,\ell_k}$ relative to the upstream section S_k and the chosen yaw configuration ℓ_k (cf. line 414 ff.); this is done sequentially. Second, the compatibility in the wind farm is finally ensured in Eq. (9); this is not sequentially.

    To describe this in more detail and to answer the question regarding the order of the sections, we use example Fig. 5 (a): $\tilde{L}_{3,\ell_2}$ only represents the compatibility of upstream section S_3 and S_2 if yaw configuration ℓ_2 was chosen in S_2; analogously, $\tilde{L}_{2,\ell_1}$ represents the compatibility of S_2 and S_1 for a specific ℓ_1. Considered individually, this allows a choice of ℓ_1 and ℓ_3 such that the yaw offset of WT 2 is not compatible, i.e., $\gamma_2(\ell_1) \neq \gamma_2(\ell_3)$. In lines 412 ff. we refer to "global consistency" by "resorting to valid yaw configurations [...] as by construction, for any WT $i \in S_{\hat{k}} \cap S_k$ with $\hat{k} \geq k+2$, necessarily also WT $i \in S_{k+1}$."; this refers to Eq. (9) (and we shall refer to it there), which makes sure that the choice of ℓ_2 is compatible with the choices of ℓ_1 and ℓ_3. In detail, we choose them via decision variables y_{k,ℓ_k} and ensure compatibility in the integer linear program by $0 \leq \sum y_{2,\ell_2} - y_{1,\ell_1} \leq 1$ (from $k = 1$) and $0 \leq \sum y_{3,\ell_3} - y_{2,\ell_2} \leq 1$ (from $k = 2$); the solution of the integer program ensures that $\gamma_2(\ell_1) = \gamma_2(\ell_2)$ and $\gamma_2(\ell_2) = \gamma_2(\ell_3)$, which means that $\gamma_2(\ell_1) = \gamma_2(\ell_3)$, i.e., the choices ℓ_1 and ℓ_3 are compatible. This works if covering sections are "directly adjacent pairs" (see line 410).

    So, it is possible to construct an example that shows that an arbitrary order which violates this condition (of directly adjacent pairs) leads to a problem. Again, we use example Fig. 5 (a) but extend the wind farm orthogonal to the wind direction such that we have 16 turbines and 8 covering sections. We choose the following order of the upstream sections: 1, 4, 7, 2, 5, 8, 3, 6. In this order, the intersection of S_k and S_{k+1} has no wind turbine; so, $\tilde{L}_{k+1} = L_{k+1}$ (independent of a chosen ℓ_k); finally, Eq. (9) does not ensure compatibility. In principle, any order of the covering sections would be possible if we adapted the method: first, we need $\tilde{L}_{\hat{k},\ell_k}$ for each(!) $\hat{k} \neq k$ and second, Eq. (9) has to ensure the compatibility of ℓ_k with each(!) $\ell_{\hat{k}}$ by $0 \leq \sum y_{\hat{k},\ell_{\hat{k}}} - y_{k,\ell_k} \leq 1$.

    Your questions show that it is important to make clear which part is sequentially and which not. In the revision, we will, in particular, refer to Eq. (9) earlier and point out that the condition "directly adjacent pairs" excludes an arbitrary order of covering sections. Nonetheless, we prefer to omit the very detailed answer given above in order to not digress and distract from the main thread.

- 1. b. Solver: "...descriptions of both the SCIP and Gurobi solvers..."—In line 526f., we state that SCIP is an open-source academic software and Gurobi is proprietary. Both solvers implement sophisticated branch-and-cut solution techniques for mixed-integer programs, which has been the standard approach for decades and is therefore not mentioned explicitly in the paper. We also point out that a detailed comparative description of the two solution frameworks is not possible since one cannot access the source code of the commercial Gurobi software.

- 1. c. Trapezoidal Sections:

  i. and ii. "sensitivity studies on the trapezoid slope" and "multiple yaw configurations"—In general, the choice of the section shape is no main aspect in the manuscript as it is interchangeable. We have looked for a plausible section shape that is not too large. Therefore, we have carefully formulated (using the words "based on" and "simply") that "the concrete chosen area is a trapezoid [...] based on this threshold: For this, we simply use the wind speeds at the so-called observation points..." (lines 280 f). It is a good idea to mention that this does not guarantee the threshold.

  In more detail, if we use a degree of 8.5 (line 283) for the trapezoid slope then the outer observation points often are located on it. The slope is increased if an observation point outside the trapezoid exceeds the threshold. We check this with multiple yaw configurations (0, the smallest, e.g. -15 deg, and the biggest, e.g., +15 deg) at one turbine, where a turbulence intensity of 0 is used in each case. The turbulence intensity of 0 has two advantages: first, the wind speed at the observation points is independent of the time, and second, it overestimates the size of the trapezoid section, see the "correlation between the increase of turbulence intensity and faster wake recovery" (line 161 f.) from Talavera and Shu (2017). It may be criticized that the wind speed directly next to the observation point (and outside the trapezoid section) does not obey the threshold. However, in the presented setup turbines always have a minimum distance of 3D = 378 m; in case of 11 m/s, the observation points after 389 m meet the threshold, and in case of 6 m/s, after 394 m; so, the threshold is not met directly after a distance of 3D, but in the case of 3D the turbines are located directly behind each other, i.e., the turbine in question is inside the section anyway. The last question about the "sections anchored at both upstream and downstream turbines" is presumably aimed at the fact that a turbine with a different (lower) wind speed could need a different (larger) section, e.g., a wind turbine in the second row in case of a wind farm with three rows; we will check the influence of this aspect. In general, we have limited ourselves to the evaluation of the observation points as only those are available directly in the used software WinFaST but checked the plausibility of the choices visually by figures like Fig. 1.

  To verify the aspects discussed, we try to work with the software FLORIS. In addition, we have already validated the covering approach (and thus, also the choice of sections) in one case by comparative computations, see lines 546 ff.

  iii. "choosing to use the "upstream" sections"—As described in the answer to i. and ii., in practice, we determine the downstream section by simulations with one turbine; then, the upstream section is determined by point mirroring of this downstream section. To avoid confusion, we will only focus on the used "upstream sections" in the revised manuscript. Defined this way, they only change the perspective ("influenced by" or "influence"). If turbine A influences turbine B, then B is influenced by A. So, it makes sense to use either the downstream or the upstream section.

  iv. "... how well ... not laid on a grid pattern"—As mentioned in line 137 it is possible to "choose the grid resolution as fine as needed to allow representing any layout". So, in terms of accuracy it will work very well, but may of course be detrimental to the run time. However, as the precomputations are the (run time) bottleneck (cf. line 634), we are confident that it will work well with a faster simulation software like FLORIS. However, we have not implemented this adaption.

- 1. d. "Combinations"—Yes, this was a mistake in the formula (first occurring in line 310); we will correct it.

- 1. e. "Loading Components"—Yes, the weightings of the tower and pitch activity penalties are included in the theoretical part (and in the implementation), but not used in the computational results (due to space limitations). Nevertheless, we see benefit in showing the potential, but will clarify it.

- 2. a. "Farm-Wide Optimisation"—Probably, line 545 is misleading: you ask for a farm-wide optimization, i.e., a full enumeration. What we did is a farm simulation with baseline and optimized yaw configurations as inputs but not a full enumeration as this is impractiable with the simulation software WinFaST (even with a 6 × 3 farm because of the run time). As mentioned in the answer to review 1, we will alternatively compare with full enumerations with the software FLORIS in the table.

- 2. b. "Timings"—We will estimate the run times of full enumerations using the number of combinations to address this question.

- 3. a. ""Deeper" Wind Farms"—Indeed, deeper wind farms (also in wind direction) are possible by the following idea: as the precomputations in the trapezoid section are the computational (run-time) bottleneck, we have to compute the yaw configurations in rows (or layers) (in this trapezoid section). So, we set the yaw offsets of the first row, run the simulation and save the resulting wind field; next, we run several yaw configurations of the second row with this wind field as input and so on. Finally, we have all yaw configurations within the trapzoid section, i.e., our precomputations. For wind directions other than 0, it is more sophisticated to define the layers. We have not discussed this due to space limitations, and have also not yet implemented this.

- 3. b. Steady-State Models:
  - i. As in the answer to review 1, we will use the software FLORIS for comparison with steady-state.
  - ii. "sufficient time for changes to propagate through the farm"—We will revise our description, also because there is another parameter in the used software WinFaST for the initial simulation time that we had not mentioned. Of course, changes need more time to propagate through the wind farm than our description suggests (observation time starts 2 mins after begin of simulation). The plausible default setting for a wind farm with three rows (with 5D distance) was 800 s for the inital simulation and 660 s for the following simulation. We adapted them to 680 s initial simulation and 540 s following simulation observing that we have to wait up to 120 s (in the following simulation) as the transient phase has not finished. So, finally yaw changes have 800 s (again) to propagate through the wind farm before observation starts. We will reformulate this and determine the propagation times (also to answer the 17th minor comment asking about the 15 min time scale to change the yaws).
- 4. "Paper Length"—We appreciate the proposed suggestions to shorten the manuscript. While addressing some of the reviewers' comments will require some space, we will do our best to be concise and to shorten the paper wherever possible without compromising clarity.
- 5. "Contributions"—Yes, we will highlight the splitting method in Abstract and Conclusions.
- 6. a. "Table 2"—We agree that a diagram would be a good idea and will try to include it in the revised manuscript, space permitting.
- 6. b. "Figure 6"—We understand this remark (to visualize wakes instead of values) for the revision also with regard to other illustrations and will work on improving the presentation.

Answers to Minor Comments of Review 2

- 1.–9. Thanks, we will realize the suggestions.
- 10. "concept explained instead"—We will try to accommodate this in the available space.
- 11. "accuracy of 1%"—The comparisons in the paper Dar et al. (2017) are described with the so-called "windfarm efficency" (referring to Adaramola and Krogstad (2011)), which is computed "from the maximum total power produced in the presence of wake effects and the maximum total power produced without considering any wake effects". We will clarify this in the revised manuscript.
  (Adaramola and Krogstad (2011): M. Adaramola and P. Krogstad, "Experimental investigation of wake effects on wind turbine performance," Renewable Energy, vol. 36, no. 8, pp. 2078–2086, 2011)
- 12.–16. Thanks, we will do this.
- 17. "15 min"—It is an example to make the order of magnitude clear. Yes, changes should at least have the time to propagate through the farm, see also the answer to major comment 3. b. ii.
- 18. "curse of dimensionality"—We will try to shorten Section 2.1.2 and merge Sections 2.1.1 and 2.1.2.
- 19. "Line 234: Surely standard IP solvers can handle black-box problems?"—Standard IP solvers require linear or some simple nonlinear objective or constraint functions to be specified *explicitly*. A black-box function is characterized by being only accessible via an "oracle" (such as simulation software output) and having no known analytical explicit formulation, which is a well-known conundrum for many practical optimization tasks. Hence, we do point out in the paper that black-box problems "cannot simply be handled" by state-of-the-art IP solvers (like SCIP and Gurobi).
- 20.–23. Thanks, we will do this.
- 24. "different time scales"—Yes, we will rework this, see also the answer to major comment 3. b. ii.
- 25. Thanks, we will add a reference number.

- 26. “turbines individually weighted”—We do not use individual weights in the computational results, but we implemented them. For clarity of notation, individual weights are not shown in Eq. (9). We will reformulate it to make this clear.
- 27.–29. Thanks, we will include the remarks.
- 30. “...steady-state model instead cut down the expected pre-computation times?”—We will do comparison computations as outlined in the answer to review 1 to show this.