On the importance of wind predictions in wake steering optimization - response to the reviewers (bis)

Elie KADOCHE[1,2], Pascal BIANCHI[2], Florence CARTON[1], Philippe CIBLAT[2], and Damien ERNST[2,3]

[1]TotalEnergies OneTech, 2 Place Jean Millier, 92400 Courbevoie, France
[2]Télécom Paris, 19 Place Marguerite Perey, 91120 Palaiseau, France
[3]Montefiore Institute, University of Liège, Belgium

We thank the reviewers for taking the time to read our first revised paper. In the second revised paper, the red and green text correspond to the differences between the first revision and the second revision (last and current version). In this document, the colored text in blue and in italic refers to comments of the reviewers. We address each comment individually below.

1 Anonymous Referee #1

I think that there is a misunderstanding about what dynamics are. If a signal (like the wind or yaw angles) is changing over time, it does not mean that dynamics are taken into account. Dynamics are taken into account when there is a model that takes as an input the time-varying signals and outputs another signal (like turbine power). This output signal is then changing due to the input signal, and its exact change depends on the model dynamics. Delays and inertia are typical phenomena that you model with a dynamical model and clearly present in the application at hand and not taken into account.

We thank the reviewer for insisting on that matter. Indeed, the term "dynamics" implies several phenomena that we are not taking into account in the simulations. Instead, we describe a temporal evolution of multiple states, interconnected via the rotational constraints of the machines. We corrected that in the revised paper.

The authors write: "Steady-state models are necessary for optimization. Higher fidelity models, taking into account the dynamics of the wind and the variations of the wake effects between time steps, exist. But these models are too computationally expensive to be used for optimization."

The statement that steady state model are necessary is questionable. It is a (over?) simplification of the reality and consequently, certain optimization problems can be solved. However, dynamical models can also be used in optimization problems, like high fidelity models with CPU time as a challenge. These things are known/researched, but where is the middle ground? A computational efficient enough dynamical model that can be used in an optimization. This seems to me the important and open question. Using a steady state model in some dynamical control framework remains questionable for me and the authors did not really convince me with their answers.

We agree that the application of steady-state models in wake steering optimization has important limitations. Finding a computational efficient enough dynamical wake model that can be used in optimization is the focus of active research in the community. But this subject goes outside the question raised in this paper. We focus on the optimization process itself, adhering to community standards by using widely accepted, open-source, low-fidelity simulators. We propose an improvement of a well known optimization problem, using state of the art steady state simulators.

We agree that the sentence "steady-state models are necessary for optimization" is badly formulated, we removed it in the revised paper. We better explained the necessity for steady-states models. An important challenge of wake steering optimization is to develop closed-loop controllers, able to perform continuous optimization, based on feedbacks of the environment (wind data, turbine orientations, power outputs, etc.). Conducting model-based, closed-loop optimization as the farm is operating, requires a simulator of the environment. In that context, high-fidelity simulators are not usable for large (offshore) wind farms because they require too much computational time. Instead, we use lower-fidelity simulators, which offer computation times fast enough to be used in a closed-loop controller.

At last, the authors state that the original optimization problem is not solvable. Why not perform a grid search? You can let the CPU do the work. If the search space is too large (which I doubt when taking a relatively small farm and relatively small prediction horizon), you can limit it around the optimal solution found by the simplified optimization problem.

We thank the reviewer for this comment. Indeed, the sentence "the optimization problem is not solvable" is badly formulated. To our knowledge, the optimization problem seems to be difficult to solve in a polynomial time. We corrected that in the revised paper.

Original problem In the original optimization problem, the number of turbines is 34 (this is a relatively small wind farm), the solution space for the yaw of one turbine in discretized in 120 values and the horizon is 10 data points. In this configuration, the total number of scenarios is equal to $120^{34^{11}} = 120^{374}$.

Simplified problem In a simplified but interesting enough problem, the number of turbines should still be 34 (this is already a small wind farm), the solution space for the yaw of one turbine could be discretized in 30 values and the horizon could be composed of 3 data points. In this configuration, the total number of scenarios is equal to $30^{34^4} = 30^{136}$.

On an Apple M2 Pro MacBook Pro with 32 GB of memory, the computation time for 1 scenario is about 0.24 seconds. Leveraging the vectorization capabilities of the simulator, we can reach 193.74 seconds for 10^4 scenarios. Given that exploring 10^{11} scenarios should take approximately 61 years, a grid-search approach does not seem practical.

2 Anonymous Referee #2

This is a much improved version of the paper. Well done considering and incorporating the previous feedback. I especially appreciate the added clarity in describing the future power heuristic and the relation back to applied problems in the conclusion.

My only feedback is that Figure 3 is still not clear. The description of the heuristic in Section 3.2.2 is good, but I'm not able to map this to the figure. Consider the following suggestions:

- *Change the black arrow that points to a label to a different type of arrow so that it does not look like the other arrows that indicate some distance.*
- *I'm not sure what the "rotation zone" represents - is it the possible cone for the wake centerline based on + / - 15 degrees yaw?*
- *In the description, it says "to get an average idea of how far the turbine will be from the predicted wind direction". It's difficult to understand what is meant by a distance from the wind direction.*
- *Is it possible to represent the turbine by a yawed line rather than a "x"?*

We thank the reviewer for the positive feedback. We updated the Figure 3 according to the proposed suggestions. The distance from the wind direction is actually the yaw of the machine, it is the angular distance between the turbine orientation and the wind direction. This is better explained in the revised paper. The rotation zone represents the cone of the future possible orientations of the turbine, centered around the current turbine orientation. This is detailed in the revised paper.