Wind Farm Optimization in PyWake/TOPFARM Using Gradients and Smart-Start Algorithms

Rafael Valotta Rodrigues, Pierre-Elouan Réthoré, Mads Mølgaard Pedersen, Katherine Dykes
DTU Wind Energy, Technical University of Denmark

[1]Correspondent author: ravaro@dtu.dk

Submission for Theme 1: Wind Resource, Wind Farms and Wakes

Abstract: The development of more cost-efficient wind farms depends on improvements in wind farm layout optimization techniques. In this context, TOPFARM is a computational tool that enables fast wind farm models evaluation [1]. TOPFARM uses OpenMDAO for optimization and PyWake for Annual Energy Production (AEP) computation with different wake models. OpenMDAO is a High-Performance Computing (HPC) tool for multidisciplinary analysis and optimization especially suited for performing gradient-based optimization [2]. Different gradient evaluation methods have been implemented for AEP optimization with respect to the wind turbine positions in TOPFARM/PyWake, including: automatic differentiation, Complex-Step (CS), and Finite Differences (FD). Automatic differentiation is performed using Autograd, a python package that computes exact derivatives (Jacobian matrix) of an objective function. The CS approach uses complex numbers to calculate directional derivatives. FD essentially perturbs the design variables to calculate all the partial derivatives, while the CS carries the perturbations on the imaginary part of the design variables. Autograd only requires one function evaluation to obtain all the gradients, while FD and CS require two times the number of design variables function evaluation. In this work, we compare these gradient estimation methods to run wind farm optimization through different wind farm scales. The simulations were performed using a smart-start algorithm to identify an initial layout that potentially maximizes AEP by selecting turbine positions that are less affected by wake effects. The same cases were also simulated not using the smart-start option aiming to compare the two strategies. Additionally, we compare these methods with other gradient-free optimization algorithms such as random search. A TOPFARM problem was defined with the position of the turbines as the design variables and spacing between turbines and boundaries as constraints. The site was defined as described in the IEA Wind task 37 case study 1 [3] site with the wind direction discretized into 16 bins, a constant wind speed of 9.8m/s, and a fixed CT curve. The turbines were placed considering a circular boundary constraint surrounding the farm, and different numbers of turbines were analyzed. For the analysis mentioned above, the ratio between the area of the circular boundary constraint and the rotors area was kept constant for the different number of turbines. Figures 1 and 2 show the results for the simulations across scales with and without the smart-start, respectively. In terms of AEP, the level of improvement consistently agrees with previous results from the existing literature [3]. The autograd achieved optimized results faster through all the scales, whereas the random-search performed similarly to the complex-step and the finite-differences for the 16 and 36 turbines case. For the 64 turbines case, the random-search converged faster than the complex-step and the finite-differences. Additionally, the autograd with smart-start converged to a higher AEP value compared to all the other options for the 64-turbines case. Moreover, the use of the smart-start allowed a faster convergence to optimized layouts. Further analysis will consider multiple smart-start

simulations (with multiple attempts of initial layout) to assess its impact on AEP and processing time.

Keywords: wind farm layout optimization, wind turbine wakes, wake aerodynamics, TOPFARM, PyWake, array interaction, wind farm arrays

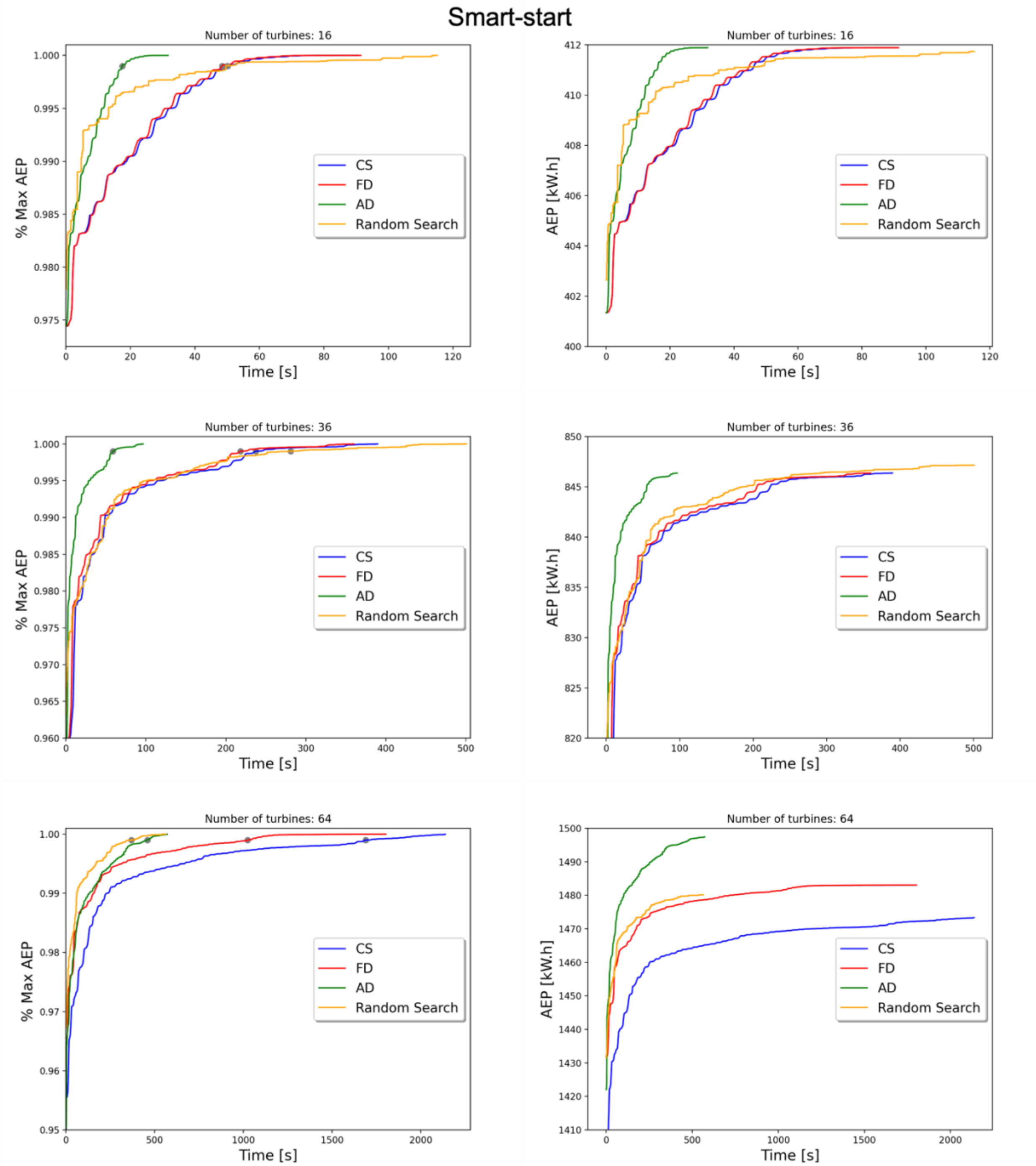

Figure 1 – Smart-start AEP optimization using autograd, complex-step, finite-differences, and random search.

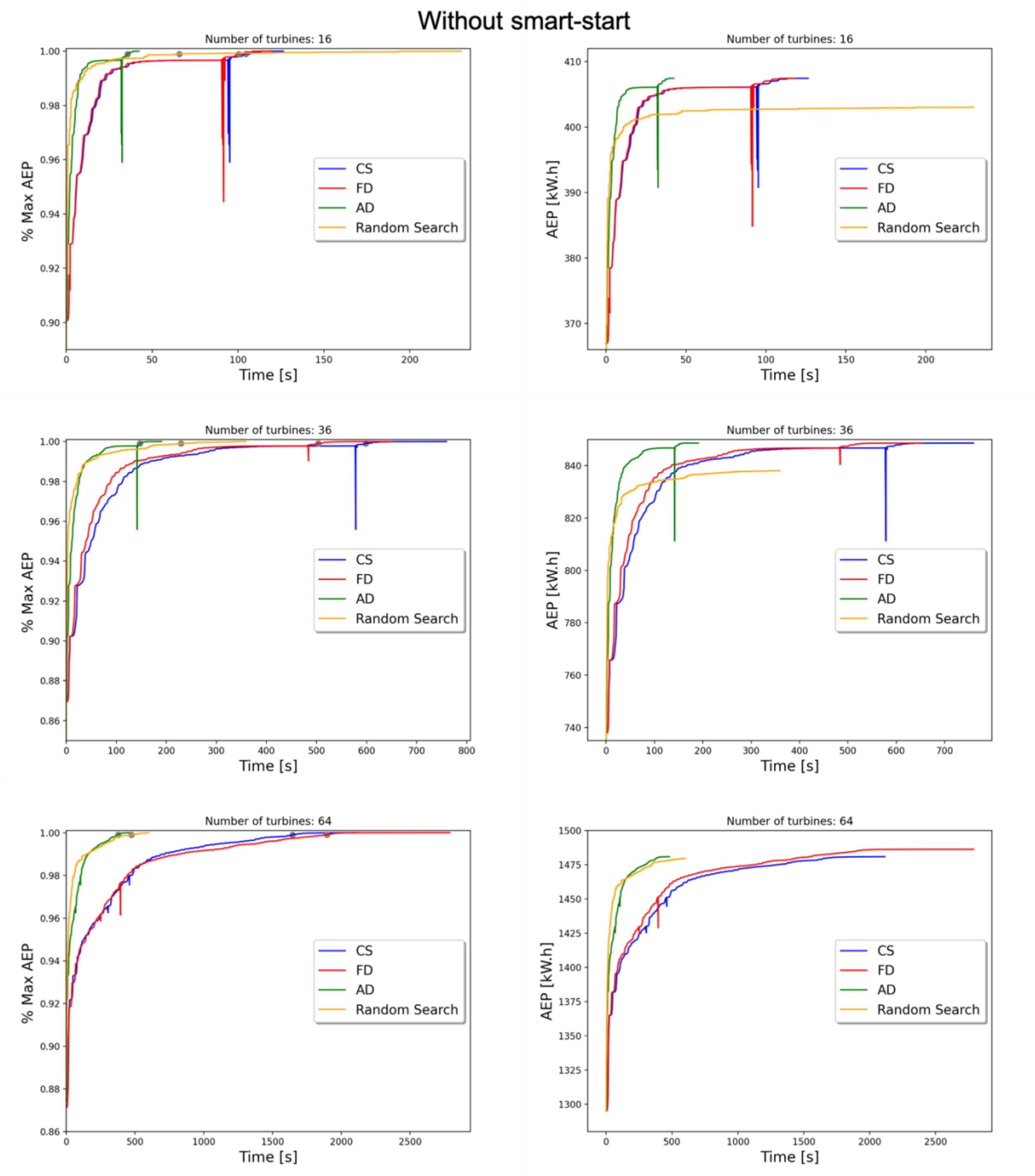

Figure 2 –AEP optimization without using the smart-start, considering autograd, complex-step, finite-differences, and random search.

References

[1] Mads M. Pedersen, Paul van der Laan, Mikkel Friis-Møller, Jennifer Rinker and Pierre-Elouan Réthoré. (2019, February 12). DTUWindEnergy/PyWake: PyWake (Version v1.0.10). Zenodo. http://doi.org/10.5281/zenodo.2562662

[2] J. S. Gray, J. T. Hwang, J. R. R. A. Martins, K. T. Moore, and B. A. Naylor, "OpenMDAO: An Open-Source Framework for Multidisciplinary Design, Analysis, and Optimization," Structural and Multidisciplinary Optimization, 2019.

[3] Baker, N. F., Stanley, A. P., Thomas, J. J., Ning, A., & Dykes, K. (2019). Best practices for wake model and optimization algorithm selection in wind farm layout optimization. In *AIAA Scitech 2019 Forum* (p. 0540).

Information about the authors

Rafael Valotta Rodrigues, Ph.D.: Dr. Rodrigues is a researcher at the Systems Engineering and Optimization Section at the Technical University of Denmark, Wind Energy Department. His research focuses on wind farm layout optimization using surrogates, wind farm flow physics, adjoint optimization, and HPC for wind energy.

Pierre-Elouan Réthoré, Ph.D.: Dr. Réthoré is a senior researcher at the Systems Engineering and Optimization Section at the Technical University of Denmark, Wind Energy Department. His research focuses on wind farm optimization and wind flow modeling.

Mads Mølgaard Pedersen, Ph.D.: Dr. Pedersen is a development software engineer at the Loads and Control Section at the Technical University of Denmark, Wind Energy Department. He is currently responsible for most of the latest developments and new releases in TOPFARM/PyWake.

Katherine Dykes, Ph.D.: Dr. Dykes is the Head-Section of the Systems Engineering and Optimization Section at the Technical University of Denmark, Wind Energy Department. Her research focuses on wind farm optimization and control.