A method for reducing multi-modality in the wind farm layout optimization problem

IOP Conf. Series: Journal of Physics: Conf. Series **1037** (2018) 042012 doi:10.1088/1742-6596/1037/4/042012

A method for reducing multi-modality in the wind farm layout optimization problem

J J Thomas and A Ning

Department of Mechanical Engineering, Brigham Young University, Provo, Utah, USA

E-mail: jaredthomas@byu.net

Abstract. This paper presents a process using an approach related to continuation optimization methods for reducing multi-modality in the wind farm layout optimization problem, referred to as Wake Expansion Continuation (WEC). The reduction in multi-modality is achieved by starting with an increased wake spread, while maintaining normal velocity deficits at the center of the wakes, and then reducing the wake spread for each of a series of optimization runs until the standard wake spread is used. Two optimization cases were tested, one with 16 turbines and one with 38 turbines, by optimizing from 200 different starting positions with a gradient-based method, a gradient-free method, and a gradient-based method using WEC. Results using WEC show a 4% mean optimized annual energy production (AEP) improvement compared to gradient-based optimization without WEC for both test cases. A gradient-free algorithm had a mean optimized AEP that was 1% higher than WEC for the 16-turbine case, but WEC resulted in a 10% higher mean optimized AEP compared to a gradient-free optimization method for the 38-turbine problem. These results are specific to the test cases and may not be generally representative.

1. Introduction

The difficulty of solving the wind farm layout optimization (WFLO) problem is primarily due to the large number of variables and constraints required for realistic problems and the multi-modal nature of the problem's design space. Gradient-free optimization methods are the most common methods used to solve the WFLO problem. However, gradient-free methods have been shown to have reduced performance with high dimensional problems [1]. The WFLO scales quickly to high dimensions as the number of turbines is increased. Gradient-based optimization methods are well suited for high dimensional problems. Gradient-based methods are not widely used for WFLO problems, but are gaining interest due to their relatively low computational cost and their ability to handle many variables and constraints. However, gradient-based methods are highly susceptible to local optima [2]. Despite this weakness, they have been shown to find good solutions to WFLO problems [3, 4, 5].

Many techniques have been presented to make the WFLO problem more tractable, including discretization, multi-start, and hybrid approaches. Discretization techniques, used with gradient-free methods, attempt to simplify the problem by reducing the number of possible solutions [6, 7]. Through discretization, the number of possible turbine locations within a wind farm can be reduced from infinite to something on the order of hundreds of locations. However, discretization disregards any locations that are not pre-selected, and can thus preclude this approach from finding even a local optimum. It is also possible that constraints on variables

other than position may render the discretized optimization problem intractable. Multi-start approaches involve running many optimizations of one problem with different starting points [8]. This approach reduces the sensitivity of gradient-based optimization methods to local optima. Hybrid approaches combine gradient-based and gradient-free algorithms iteratively [9, 10, 11], and, depending on the problem size, can yield result quality comparable to multi-start approaches [9]. While each of these techniques yield improved results, there is still need for improvement because current methods have a wide spread in the quality of results, are highly dependent on starting locations, and/or artificially limit the design space. These limitations indicate that the optimizations are not converging to the global optimum.

Current methods seek to avoid local optima by searching the existing design space more broadly. This approach becomes intractable for large optimization problems with many variables and constraints. To address this issue, we propose a process for use with gradient-based optimization algorithms that temporarily reduces the number and magnitude of local optima using an approach related to continuation optimization methods such as the approach given in [12]. However, while continuation optimization uses a numerical approximation of the design space, our proposed process takes advantages of the physical properties of the design space directly. We will refer to the new process as Wake Expansion Continuation, or WEC.

2. Methods

2.1. Wake Expansion Continuation

The two primary characteristics that simple wake models seek to capture are wake spread and velocity deficit. These characteristics in turn largely determine the shape of the wind farm layout design space. The fluctuations of wind speed as turbines move in and out of the wakes during optimization are primarily responsible for the multi-modal nature of the WFLO problem.

During optimization, the spaces between wakes translate to locally optimal locations for turbines. However, it is often the case that there are more optimal turbine locations that are not found by the optimizer because the turbines are effectively stuck on the peaks between wakes. The proposed WEC seeks to overcome this problem by temporarily reducing the multi-modality of the design space. The WEC is comprised of three basic steps:

1) Determine how the wake spread and wake deficit are controlled for the selected model(s).
2) If necessary, add a factor to the model such that the wake spread can be directly controlled without significantly altering the wake deficit in the center of the wake. (While we assume that it is important to avoid altering the velocity in the wake significantly, this assumption should be the subject of future study.)
3) Run a series of optimizations such that the wake spread is larger than normal for the first optimization, and then reduces with each subsequent optimization until the wake spread is no longer altered from the original model. The result of each optimization after the first should be started using the optimal layout found in the previous optimization.

By spreading the wakes, we can effectively fill in the gaps between wakes so that a gradient-based optimization algorithm can bypass local optima and proceed to a better solution. Another way of think about this is that by spreading the wakes, more turbines see the influence of each wake, which provides more information to the gradient-based optimization algorithm about the design space through the Jacobian.

Because of the general nature of the proposed method, it could theoretically be applied to a range of wake models, including the cosine version of the Jensen model [13], the FLORIS model [14, 15], and either version of the Bastankhah and Porté-Agel wake model [16, 17].

2.2. Wake and Wind Farm Models

In the Bastankhah and Porté-Agel wake model, the two primary characteristics of the wakes, wake deficit and wake spread, are particularly easy to differentiate. This, along with the smoothness and differentiability of the model, make it a good example for demonstrating WEC. We will use the Bastankhah and Porté-Agel wake model, as defined in eq. (1) [17], along with part of the Niayifar and Porté-Agel wind farm model [18] to demonstrate WEC.

$$\frac{\Delta\bar{u}}{\bar{u}_\infty} = \left(1 - \sqrt{1 - \frac{C_T \cos\gamma}{8\sigma_y\sigma_z/d^2}}\right) \exp\left(-0.5\left(\frac{y-\delta}{\sigma_y}\right)^2\right) \exp\left(-0.5\left(\frac{z-z_h}{\sigma_z}\right)^2\right) \tag{1}$$

Where $\Delta\bar{u}/\bar{u}_\infty$ is the wake velocity deficit, C_T is the thrust coefficient, γ is the upstream turbine's yaw angle with respect to the inflow direction, $y-\delta$ and $z-z_h$ are the distances of the point of interest from the wake center in the cross-stream horizontal and vertical directions respectively, and σ_y and σ_z are the standard deviations of the wake deficit in the cross-stream horizontal and vertical directions as defined in eqs. (2) and (3).

$$\sigma_y = k_y(x - x_0) + \frac{D_r \cos\gamma}{\sqrt{8}} \tag{2}$$

$$\sigma_z = k_z(x - x_0) + \frac{D_r}{\sqrt{8}} \tag{3}$$

In eqs. (2) and (3), x is the downstream distance from the turbine generating the wake to the point of interest, x_0 is the length of the wake potential core, D_r is the diameter of the turbine generating the wake, and k_y and k_z were determined as a function of turbulence intensity (I) as defined in eq. (4)[18].

$$k^* = 0.3837I + 0.003678 \tag{4}$$

While the Niayifar and Porté-Agel wind farm model calculates k^* based on local turbulence intensity at each turbine, the local turbulence intensity calculations introduce more local optima and discontinuities. For this reason, we chose to ignore local turbulence intensity in this study. While local turbulence intensity does impact the accuracy of the power predictions, it does not alter the general trends within the design space, and most simple wake models ignore local turbulence intensity. We are currently investigating possible solutions to the problems with the local turbulence intensity calculations.

The Gaussian shape of the Bastankhah and Porté-Agel wake model is well suited for gradient-based optimization because it is smooth, continuous, and has no flat regions. However, in the near wake, the model can either be flat, which can cause premature convergence, or be undefined, which can cause optimizations to fail. To resolve this issue, we used a simple solution for optimization purposes only: a linear interpolation of the velocity deficit, from the rotor hub to the length of the wake potential core, that maintains the Gaussian shape of the wake all the way to the rotor location. Because no turbines will be placed in this region of the wake in the final optimized layout, the accuracy of the model in the near wake is second in importance to wake shape and continuity.

We combined the wake deficits using a linear combination method as discussed in [18]. To save computation time, the inflow wind speed at each turbine was approximated using a single sample at the wind turbine hub location. Individual turbine inflow wind velocities, U_i, were solved consecutively from upstream to downstream. The power output of each turbine was then calculated based on eq. (5)

$$P_i = \frac{1}{2}\rho A_{r,i} C_P U_i^3 \tag{5}$$

where ρ is the air density, $A_{r,i}$ is the rotor-swept area of turbine i, and C_P is the power coefficient. The values of C_P and C_T were determined based on the calculated inflow velocity of each turbine and the power curve and thrust coefficient curve presented in [18].

2.3. Applying WEC

In this section we apply WEC, step by step, as presented in section 2.1 using the Bastankhah and Porté-Agel wake model.

2.3.1. WEC Step (1) The first parenthetical term of eq. (1) defines the magnitude of the velocity deficit. The exponential terms determine the wake spread. The wake spread and velocity deficit are coupled through σ_y and σ_z. However, because the spread and magnitude are expressed in separate terms, it is possible to adjust one without impacting the other.

2.3.2. WEC Step (2) We gained independent control of the wake spread from the wake velocity deficit by applying a factor, ξ, to σ_y and σ_z inside the exponential terms of eq. (1), as shown in eq. (6).

$$\frac{\Delta\bar{u}}{\bar{u}_\infty} = \left(1 - \sqrt{1 - \frac{C_T \cos\gamma}{8\sigma_y\sigma_z/d^2}}\right) \exp\left(-0.5\left(\frac{y-\delta}{\xi\sigma_y}\right)^2\right) \exp\left(-0.5\left(\frac{z-z_h}{\xi\sigma_z}\right)^2\right) \tag{6}$$

The addition of ξ allows manipulation of the shape of the design space. By increasing the value of ξ we can widen the wakes without changing the magnitude of the velocity deficit in the center of the wakes. As the wakes widen, they mix and smooth out the local optima, as shown in figs. 1 and 2.

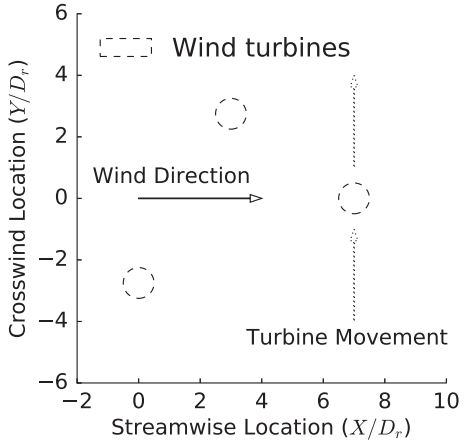

Figure 1. Simple design space used to demonstrate the effects of the relaxation factor, ξ, on the wind farm layout design space (see fig. 2).

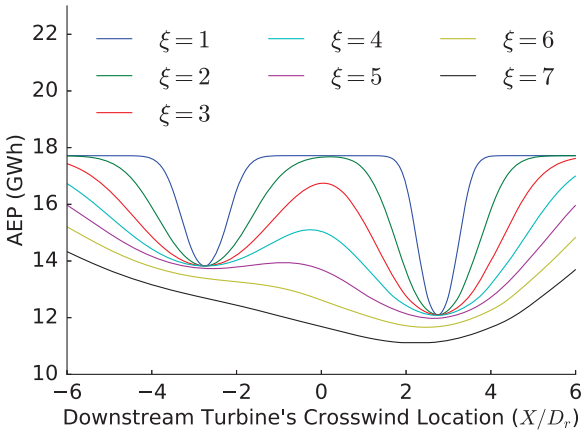

Figure 2. The impact of the relaxation factor, ξ, on a simple wind farm layout design space with three turbines and one wind direction. One turbine was moved across the wakes of two upstream turbines (see fig. 1).

Local optima scattered throughout the design space are reduced as the spaces between wakes are filled with mixed wakes. Larger values of ξ allow the smaller local optima to disappear completely. Smaller values of ξ allow for more accurate wake widths but with an increase in the number and magnitude of local optima. When ξ is set to 1, eq. (6) reduces to eq. (1).

2.3.3. WEC Step (3) To gain the benefits of WEC without losing the accuracy of the wake model, the optimization must be run multiple times, in a continuation approach, with values

of ξ decreasing to unity. The results from optimizing with each value of ξ can then be used to warm start the next optimization. In this way, the gradient-based optimization can intelligently explore the design space and then refine the model to find a final, more accurate, result. The interative optimizations are generally fairly fast due to starting from the previous optimized solution, so WEC does not necessarily take as long to run as an equal number of independent optimizations. For the optimization results presented in this paper, ξ took on the values [3, 2.75, 2.5, 2.25, 2.0, 1.75, 1.5, 1.25, 1.0]. These values were selected primarily through trial and error. The selection of these values should be investigated in more depth in future work.

2.4. Test Cases and Optimization

To demonstrate the effectiveness of WEC, we present two wind farm optimization cases. For both cases, we used the Vestas V-80 2MW wind turbine. The cases were chosen to represent a range in the size, complexity, and difficulty of the optimization problem.

We created 199 pseudo-random starting wind farm layouts for each case. Each of the starting layouts had all the turbines inside the wind farm boundary constraint and did not have any turbine spacings less than one rotor diameter. These starting locations, along with the planned locations shown in figs. 3 and 5, served as the 200 starting locations for the optimizations.

To set up the optimizations, we used OpenMDAO: a computing platform for systems analysis and multidisciplinary optimization [19]. The optimizations were solved using SNOPT, a sparse non-linear optimization algorithm [20]; WEC with SNOPT as the optimization algorithm; and NSGA-II, a non-dominated sorting genetic algorithm [21]. Both of the optimization algorithms were used as implemented in PyOptSparse [22], except that a simple tolerance for convergence was added to NSGA-II. The added tolerance in NSGA-II required that the best solution not change more than some specified value for a minimum number of generations.

For each test case, the population size for NSGA-II was set to be ten times the number of design variables ($10 \times N_{turbines} \times 2$) and the convergence tolerance was set to be 1×10^{-4}. The simulation was considered converged if the objective function value did not change more than the convergence tolerance for 100 or 200 consecutive generations for case 1 and case 2 respectively. Total generations were mostly in the thousands for each optimization run.

2.4.1. Case 1: Small Grid Case 1 was carefully selected to provide a meaningful problem that would be tractable for all the optimization methods. We defined a wind farm with 16 wind turbines and a square boundary with enough space for four rows and columns of wind turbines with spacings of five rotor diameters (see fig. 3). We used a simple wind rose composed of a double Gaussian distribution binned into 20 directions and a constant wind speed of 10 m/s in all directions (see fig. 4).

For case 1, the optimization problem was formulated as shown in eq. (7)

$$\begin{aligned} \underset{x_i, y_i}{\text{maximize}} \quad & AEP(x_i, y_i,) \quad i = 1...16 \\ \text{subject to} \quad & S_{i,j} \geq 2D_r \quad i, j = 1...16, \quad i \neq j \\ & x_{min} \leq x_i \leq x_{max} \quad i = 1...16 \\ & y_{min} \leq y_i \leq y_{max} \quad i = 1...16 \end{aligned} \tag{7}$$

Where (x_i, y_i) is the position of each turbine i, $S_{i,j}$ represents the separation distance between each pair of turbines i and j, and $x_{max/min}$ and $y_{max/min}$ represent the boundaries of the wind farm. This case has a total of 32 variables and 120 constraints.

2.4.2. Case 2: Round Farm Case 2 was created to be significantly more challenging than the first case. We defined a wind farm with 38 wind turbines and a circular boundary (see fig. 5).

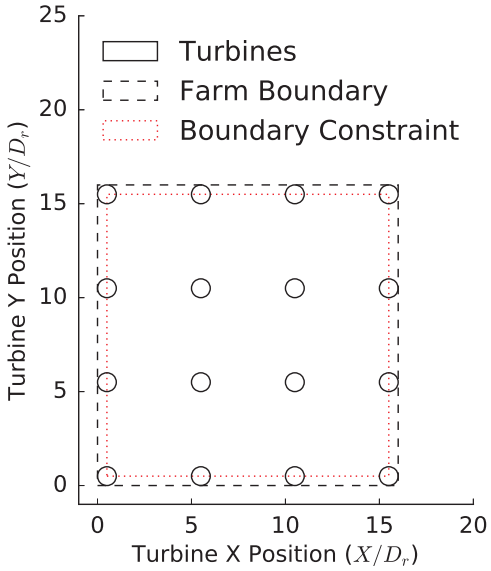

Figure 3. Baseline wind farm layout for case 1. The circles marking turbine locations are to scale, with diameters equal to the rotor diameter.

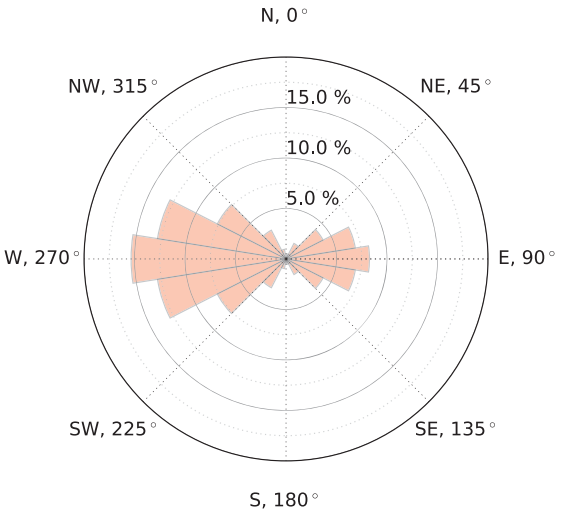

Figure 4. Direction probability wind rose for case 1. This wind rose is composed of a double Gaussian distribution binned into 20 directions.

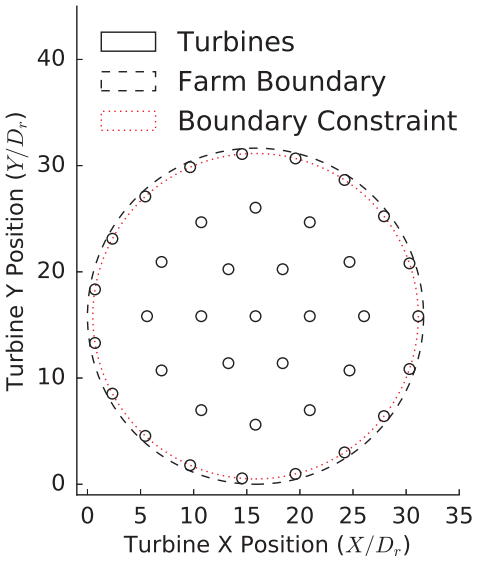

Figure 5. Baseline wind farm layout for case 2. The circles marking turbine locations are to scale, with diameters equal to the rotor diameter.

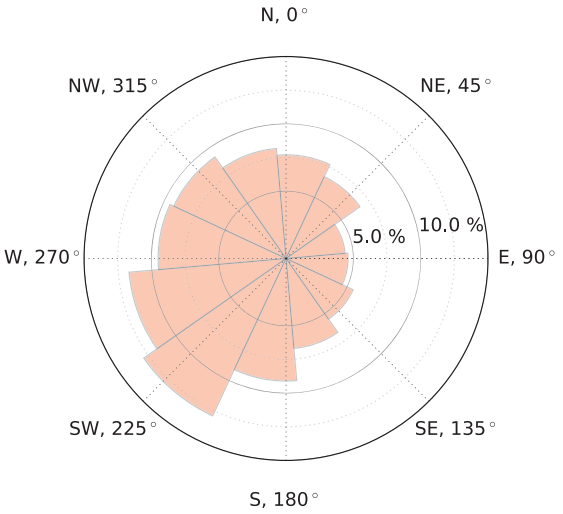

Figure 6. Directional probability wind rose for case 2. This is the Nantucket wind rose binned into 12 directions [23].

The size of the boundary allowed for at least a five-diameter spacing between turbines. We used a simplified version of the Nantucket wind rose binned into 12 directions with a constant wind

speed of 10 m/s in all directions (see fig. 6).

The optimization problem for case 2 was formulated as

$$\begin{aligned} \underset{x_i,y_i}{\text{maximize}} \quad & AEP(x_i, y_i,) \quad i = 1...38 \\ \text{subject to} \quad & S_{i,j} \geq 2D_r \quad i,j = 1...38 \quad i \neq j \\ & [x_c - x_i]^2 + [y_c - y_i]^2 \leq R_b^2 \quad i = 1...38 \end{aligned} \tag{8}$$

where (x_c, y_c) is the location of the center of the wind farm, and R_b is the radius of the wind farm boundary. This case has a total of 76 variables and 741 constraints.

3. Results

We compared optimized AEP, as well as the number of function calls required, for each of the cases discussed previously. We used function calls as a surrogate for time. We did not report wall time because we did not maintain enough consistency in the computational resources used for each optimization run (cores, processor types, computational isolation, etc). Results for each test case will be presented and discussed in the following sections.

3.1. Case 1 Results

Results for case 1 are presented in figs. 7 and 8 and table 1. The SNOPT results had the lowest mean AEP (167.39 GWh) and the highest standard deviation (2.74 GWh). The results using WEC with SNOPT had a mean AEP and standard deviation of 174.13 GWh and 2.26 GWh respectively. The results using NSGA-II had the highest mean AEP (175.98 GWh) and lowest standard deviation (0.65 GWh) of the optimization methods tested. The maximum AEP values for SNOPT, SNOPT with WEC, and NSGA-II were 174.23 GWh, 177.25 GWh, and 177.69 GWh respectively (see table 1). The AEP results for all cases were fairly normally distributed, but the results with SNOPT and WEC had several low outliers (see fig. 7).

The median number of function calls required by SNOPT, SNOPT with WEC, and NSGA-II were 246, 1,026, and 368,003 respectively. The high values for function calls required by SNOPT, SNOPT with WEC, and NSGA-II were 10,461, 11,216, and 1,342,723 respectively. However, the function call distributions for both SNOPT and SNOPT with WEC are highly skewed, with more runs grouped at the low end of the distributions. The function call distribution for NSGA-II is more normally distributed, but is still slightly skewed (see fig. 8).

3.2. Case 2 Results

Results for case 2 are presented in figs. 9 and 10 and table 2. The NSGA-II results had the lowest mean AEP for case 2 (331.98 GWh) and the second highest standard deviation (3.74 GWh). The results using WEC with SNOPT had the highest mean AEP and lowest standard deviation, 379.94 GWh and 2.24 GWh respectively. The results using SNOPT by itself had the second highest mean AEP (364.93 GWh), but highest standard deviation (5.88 GWh) of the optimization methods tested. The maximum AEP values for SNOPT, SNOPT with WEC, and NSGA-II were 380.71 GWh, 384.55 GWh, and 342.56 GWh respectively (see table 2). The AEP results for all cases were fairly normally distributed, but the results with SNOPT alone had several low outliers (see fig. 7).

The median number of function calls required by SNOPT, SNOPT with WEC, and NSGA-II were 242.5, 1,482, and 551,003 respectively. The high values for function calls required by SNOPT, SNOPT with WEC, and NSGA-II were 893, 6,748, and 1,368,003 respectively. However, the function call distributions for SNOPT with WEC was again highly skewed, with more runs grouped at the low end of the distribution with quite a few outliers at the higher end.

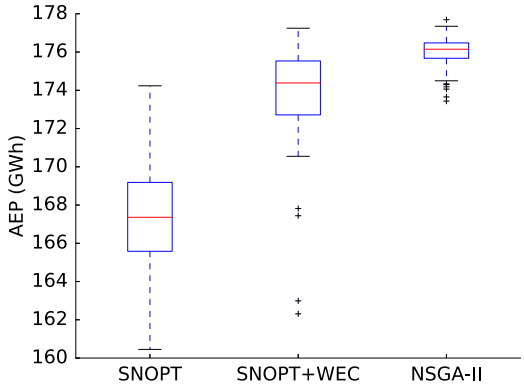

Figure 7. Box plot comparing the AEP results across each optimization approach for case 1 (16 turbines).

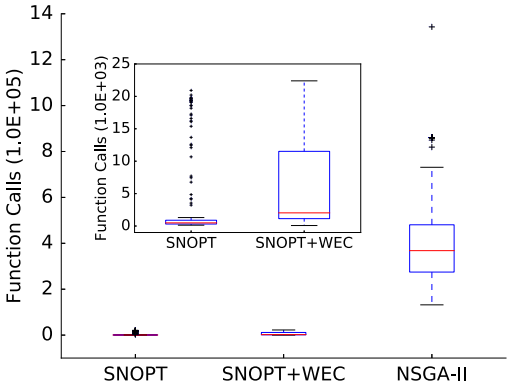

Figure 8. Box plot comparing the number of function calls used by each optimization approach for case 1 (16 turbines).

The function call distributions for NSGA-II and SNOPT and are more normally distributed, but are also slightly skewed (see fig. 8).

It may be noted that the function calls required for each method to solve case 2 are comparable to the function calls required for case 1. This is likely due to a combination of design variable and constraint scaling difference, the objective being scaled down more for case 2 (1×10^{-5}) than for case 1 (1×10^{-3}), and the SNOPT objective tolerance being loosened from case 1 (1×10^{-5}) to case 2 (1×10^{-4}) because the higher tolerance is difficult to reach for larger problems.

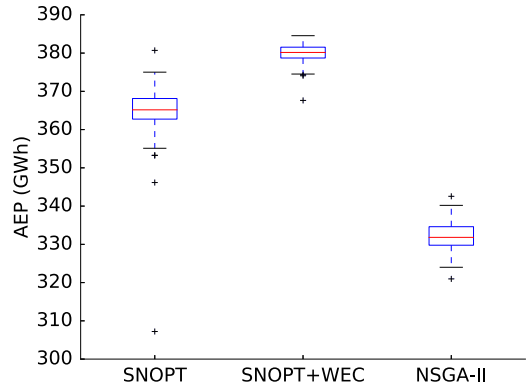

Figure 9. Box plot comparing the AEP results across each optimization approach for case 2 (38 turbines).

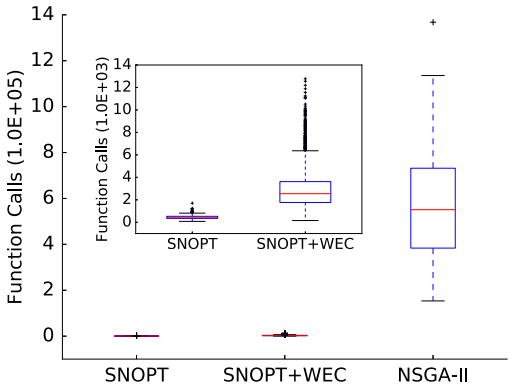

Figure 10. Box plot comparing the number of function calls used by each optimization approach for case 2 (38 turbines).

4. Discussion

In the WFLO problem, we are looking for high AEP values, a tight distribution, and relatively few function calls. The benefit of high AEP means that the resulting wind farm will be more efficient, likely leading to a reduced cost of energy. A tight distribution means that the method is consistent, that the optimized solution is independent of the starting layout. The tighter the

Table 1. Optimization Results for Case 1

	Function Calls			Annual Energy Production (GWh)			
Method	Median	Low	High	Mean	SD	Low	High
SNOPT	246	67	10,461	167.39	2.74	160.45	174.23
SNOPT+WEC	1,026	40	11,216	174.13	2.26	162.31	177.25
NSGA-II	368,003	131,843	1,342,723	175.98	0.65	173.44	177.69

Note: AEP for the layout in fig. 3 was 160.18 GWh

Table 2. Optimization Results for Case 2

	Function Calls			Annual Energy Production (GWh)			
Method	Med.	Low	High	Mean	SD	Low	High
SNOPT	242.5	82	893	364.93	5.88	307.21	380.71
SNOPT+WEC	1481	119	6748	379.94	2.24	367.63	384.55
NSGA-II	551,003	153,523	1,368,003	331.98	3.74	320.96	342.56

Note: AEP for the layout in fig. 5 was 352.02 GWh

distribution the fewer optimizations need to be run before we are confident that we have one of the best wind farm designs we can get. A low number of function calls means that wind farm design can be done more cheaply and that more variables could potentially be tested, which could again lead to a lower cost of energy. Each of the methods tested demonstrated at least one of these characteristics.

The relatively high AEP results from NSGA-II on the smaller case, compared to the relatively low AEP results of NSGA-II on larger case illustrates an inherent weakness of gradient-free algorithms. Namely, that increased dimensionality generally leads to a decrease in the performance of gradient-free algorithms [1]. As shown by the high number of function calls for NSGA-II, gradient-free algorithms also tend to take a relatively long time to run. These weakness are the main drivers for increasing interest in gradient-based algorithms.

The relatively wide spread and moderate results from SNOPT on both cases demonstrate one of the weaknesses of gradient-based algorithms. While gradient-based algorithms tend to need fewer function calls, and thus typically less time, they are highly susceptible to local optima. That WEC AEP results were relatively much higher and less spread for both test cases, as compared with SNOPT, seems to indicate that the process is at least partially remedying the problem of local optima.

The high values of AEP and lower standard deviation of AEP compared to SNOPT with WEC, indicates that SNOPT with WEC is more accurate and reliable. So much so that, ignoring outliers, the probability distributions for SNOPT with and without WEC from case 2 just overlap only slightly (see fig. 9). This reliability does come at the cost of more function calls than SNOPT without WEC. The increase in function calls when using WEC is expected because the relaxation optimization method ran nine optimizations to convergence (one for each value of ξ) for every one optimization by SNOPT alone. The number of function calls needed for WEC was still only a fraction of what was need for NSGA-II, so the added cost in function calls seems reasonable from that perspective. However, because WEC has a smaller probability distribution than SNOPT by itself, fewer runs would be needed to gain the same level of confidence in the results. The reduction in overall runs could mean an overall reduction in the number of function evaluations for WEC as compared to SNOPT alone.

5. Conclusion

The WEC proposed in this paper uses characteristics of typical wake models to reduce the multi-modal nature of the design space. We tested WEC on two WFLO problems, one with 16 turbines and one with 38 turbines. Results using WEC show a 4% mean optimized AEP improvement compared to gradient-based optimization without WEC for both test cases. A gradient-free algorithm had a mean optimized AEP that was 1% higher than WEC for the 16-turbine case, but WEC resulted in a 10% higher mean optimized AEP compared to a gradient-free optimization method for the 38-turbine problem. Future work should investigate potential improvements and best practices for WEC, apply WEC to other wake models, provide a more complete comparison to gradient-free wind farm layout optimization including discrete parameterization, and validate the results obtained through WEC using a higher-order modeling approach such as Large Eddy Simulation (LES).

Acknowledgments

This work was supported by the National Science Foundation under Grant No. 1539384. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] Rios L M and Sahinidis N V 2013 *Journal of Global Optimization* **56** 1247–1293 ISSN 1573-2916 URL https://doi.org/10.1007/s10898-012-9951-y
[2] Herbert-Acero J, Probst O, Rethore P, Larsen G and Castillo-Villar K 2014 *Energies* **7**
[3] Fleming P, Ning A, Gebraad P and Dykes K 2015 *Wind Energy*
[4] Guirguis D, Romero D A and Amon C H 2016 *Applied Energy* **179** 110 – 123 ISSN 0306-2619 URL http://www.sciencedirect.com/science/article/pii/S0306261916308765
[5] Gebraad P, Thomas J J, Ning A, Fleming P and Dykes K 2017 *Wind Energy* **20** 97–107
[6] Mosetti G, Poloni C and Diviacco B 1994 *Journal of Wind Engineering and Industrial Aerodynamics* **51** 105 – 116 ISSN 0167-6105 URL http://www.sciencedirect.com/science/article/pii/0167610594900809
[7] Grady S, Hussaini M and Abdullah M 2005 *Renewable Energy* **30** 259 – 270 ISSN 0960-1481 URL http://www.sciencedirect.com/science/article/pii/S0960148104001867
[8] Martínez J E G and Zaayer M 2014 *Layout optimisation of offshore wind farms with realistic constraints and options* Master's thesis Delft University of Technology
[9] Réthoré P E, Fuglsang P, Larsen G C, Buhl T, Larsen T J and Aagaard Madsen H 2014 *Wind Energy* **17** 1797–1816
[10] Graf P, Dykes K, Scott G, Fields J, Lunacek M, Quick J and Rethore P E 2016 *Journal of Physics: Conference Series* **753** 062004 URL http://stacks.iop.org/1742-6596/753/i=6/a=062004
[11] Mittal P and Mitra K 2017 *IFAC-PapersOnLine* **50** 159 – 164 ISSN 2405-8963 20th IFAC World Congress URL http://www.sciencedirect.com/science/article/pii/S2405896317300393
[12] Mobahi H and III J F 2015 URL https://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/9299
[13] Jensen N O 1983 A note on wind generator interaction Tech. rep. Risø National Laboratory DK-4000 Roskilde, Denmark
[14] Gebraad P M O, Teeuwisse F W, van Wingerden J W, Fleming P A, Ruben S D, Marden J R and Pao L Y 2014 *Wind Energy*
[15] Thomas J, Gebraad P and Ning A 2017 *Wind Engineering*
[16] Bastankhah M and Porté-Agel F 2014 *Renewable Energy* **70** 116–123
[17] Bastankhah M and Porté-Agel F 2016 *Journal of Fluid Mechanics* **806** 506–541
[18] Niayifar A and Porté-Agel F 2016 *Energies* **9** 1–13
[19] Gray J, Moore K T and Naylor B A 2010 *Proceedings of the 13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference*
[20] Gill P, Murray W and Saudners M 2005 *SIAM Review* **47** 99–131
[21] Deb K, Pratap A, Agarwal S and Meyarivan T 2002 *IEEE Transactions on Evolutionary Computation* **6** 182–197 ISSN 1089-778X
[22] Perez R E, Jansen P W and Martins J R R A 2012 *Structures and Multidisciplinary Optimization* **45** 101–118
[23] Western Regional Climate Center 2012 Station wind rose: Nantucket URL https://wrcc.dri.edu/cgi-bin/wea_windrose.pl?laKACK