Proceedings of the 2021 IISE Annual Conference
A. Ghate, K. Krishnaiyer, K. Paynabar, eds.

Wake Effect Calibration in Wind Power Systems with Adaptive Sampling based Optimization

January 2021

Abstract

The calibration of the wake effect in wind turbines is computationally expensive and high risk due to noise in the data. Wake represents the energy loss in downstream turbines and characterizing it is essential to design wind farm layout and control turbines for maximum power generation. With big data, calibrating the wake parameters is a derivative-free optimization that can be computationally expensive. But with stochastic optimization combined with variance reduction, we can reach robust solutions by harnessing the uncertainty through two sampling mechanisms: the sample size and the sample choices. We do the former by generating a varying number of samples and the latter using the variance-reduced sampling methods.

Keywords: wind turbines, trust-region, derivative-free optimization, variance reduction

1 Introduction

The presence of multiple turbines in the wind farm decreases downstream turbines' efficiency due to the wake effect. Limited land availability and high set-up and operating costs locate many turbines in the downstream direction. It is necessary to accurately model this wake effect to obtain the best performance out of the wind farm. Modeling the wake effect involves estimating the power deficit at the downstream turbines and deals with accuracy versus computational complexity. Jensen's wake model is a physics-based engineering model widely used in industry and academia because of its computational simplicity [**?**]. Engineering models like Jensen's wake model make certain assumptions that introduce unknown parameters, such as wake decay coefficient denoted by θ. It is often difficult to quantify these parameters experimentally, and performing full-scale computer simulations can be costly. Moreover, analysis of the uncertainty present in the available data, especially in wind farms, appears to be highly heterogeneous and challenging [**?**]. With the computational time for running the simulations being of the order of minutes, we have the flexibility of running more simulations, and data scarcity is not a concern as wind power generation data is widely available. The reasons above call for developing computationally efficient computer-based parameter calibration methods that are well-informed by estimated uncertainty. To limit the number of computer simulations, Bayesian approaches that use Gaussian processes to model uncertainty has been prevalent and extensively used [**?**]. However, Bayesian methods do not perform well when the computer-based model has some bias associated with it [**?**], and are highly sensitive to kernels and other hyper-parameters. These limitations have encouraged modeling the calibration problem with stochastic optimization algorithms, which are broadly classified into two categories: gradient-based algorithms and trust-region algorithms. While stochastic gradient-based algorithms have been used for parameter calibration [**?**], we use trust-regions in a derivative-free fashion since they exhibit versatility against stochastic noise. This study enhances the classic procedure with adaptive and stratified sampling techniques to reduce the cost of sufficiently accurate estimates in the optimization.

2 Preliminaries

This section provides the problem statement, notations, and an introduction of stratified sampling, trust-region optimization, and adaptive sampling tools used in the solution method to solve this problem.

2.1 Data-driven Stochastic Optimization Problem Statement

Following the effective wind speed in Jensen's wake model and the power curves, a deterministic computer model generates $y^c(\theta; x)$, a T-dimensional vector of power generated at each turbine in the wind farm for a given wind speed, x. The computer-generated response accuracy is then evaluated by comparison with y, the observed response at the same wind speed. Given the available dataset containing $\mathcal{D} = \{< x_j, y_j >\}_{j=1}^n$, $x_j \in \mathbb{R}$ being the input predictor and $y_j \in \mathbb{R}^T$ being the input response (power generated at turbines t, $t = 1, 2, \cdots, T$), one seeks a calibration parameter that minimizes $n^{-1}\sum_{j=1}^n \ell(y^c(\theta; x_j), y_j) := \|y^c(\theta; x_j), y_j\|_1$, where the loss function $\ell(y^c(\theta; x_j), y_j) : \mathbb{R}^T \to \mathbb{R}$ is the mean absolute error. The wind characteristic x_j can be the wind speed, the wind direction, or the turbulence intensity, which is the ratio of the 10-minute wind speed standard deviation to the 10-minute wind speed average. This problem is also known as *empirical risk minimization*. However, using the entirety of the available dataset may not be efficient due to its size. It may further not be effective in showing the performance of the calibrated outputs on other unobserved instances of data, or overfitting. Instead, our true objective function is

$$\begin{aligned} \min_{\theta} \quad & \mathbb{E}_{X,Y}[\ell(y^c(\theta; X), Y)] \\ \text{subject to} \quad & \theta_{min} \le \theta \le \theta_{max}, \end{aligned} \tag{1}$$

where X and Y come from an unknown joint distribution function $f_{X,Y}(x, y)$. This problem can be reduced to a stochastic problem where the expected value in (1) is estimated with a sample average approximation, i.e., $F(\theta, \mathcal{S}(\theta)) := |\mathcal{S}(\theta)|^{-1}\sum_{<x_j,y_j>\in\mathcal{S}(\theta)} \ell(y^c(\theta; x_j), y_j)$ with a sampled subset of the data $\mathcal{S}(\theta) \subseteq \mathcal{D}$. Using a sample of the data will capture more general patterns while avoiding selection bias. While the calibration parameter can be unbounded, in this context, we consider a bounded case having θ_{min} and θ_{max}. The wake decay coefficient always has to be positive, and it is not theoretically possible for it to be greater than 1; hence, $\theta_{min} = 0, \theta_{max} = 1$. The following assumptions are stipulated for the remainder of this paper:

A1. As the size of $\mathcal{S}(\theta)$ increases, $F(\theta, \mathcal{S}(\theta))$ converges to $\mathbb{E}_{X,Y}[\ell(y^c(\theta; X), Y]$ almost surely.

A2. The function $\mathbb{E}_{X,Y}[\ell(y^c(\theta; X), Y]$ in θ is bounded below and its gradient Lipschitz continuous, that is there exists $L < \infty$ such that $||\nabla\mathbb{E}_{X,Y}[\ell(y^c(\theta_1; X), Y] - \nabla\mathbb{E}_{X,Y}[\ell(y^c(\theta_2; X), Y]|| \le L||\theta_1 - \theta_2||$.

A3. The wake decay coefficient θ is independent of the input predictor.

2.2 Stratified Sampling

Stratified sampling involves partitioning the data into I strata, i.e., $\mathcal{D} = \cup_i^I \mathcal{D}_i$ and $\mathcal{D}_i \cap_{i\neq i'} \mathcal{D}_{i'} = \emptyset$, which is a variance reduction technique that allocates samples to each $\mathcal{D}_i$ proportional to its density and impact on the output variability. We define $p_i = |\mathcal{D}_i|/|\mathcal{D}|$ as the ratio of total points in stratum i, and $\hat{\sigma}_{Y,i}$ as the estimated variability of the true power generated by all the wind turbines under consideration in that stratum. Given a θ and I strata, if the estimated objective in stratum i is $F(\theta, \mathcal{S}_i(\theta))$, where $\mathcal{S}_i(\theta) \subseteq \mathcal{D}_i$, with the estimated variance $\widehat{\text{Var}}(F(\theta, \mathcal{S}_i(\theta))) = \hat{\sigma}^2_{F,i}(\theta)/|\mathcal{S}_i(\theta)|$, then we have the estimated mean and its estimated variance as

$$F(\theta, \mathcal{S}(\theta)) = \sum_{\forall i\in I} p_i F(\theta, \mathcal{S}_i(\theta)) = \sum_{\substack{<\boldsymbol{x},\boldsymbol{y}>_i \\ \forall i\in I}} p_i \frac{\|\ell(y^c(\theta; \boldsymbol{x}), \boldsymbol{y})\|_1}{n_i(\theta)}, \tag{2}$$

$$\widehat{\text{Var}}(F(\theta, \mathcal{S}(\theta))) = \sum_{\forall i\in I} p_i^2 \widehat{\text{Var}}(F(\theta, \mathcal{S}_i(\theta))) = \sum_{\forall i\in I} \frac{p_i^2\hat{\sigma}^2_{F,i}(\theta)}{n_i(\theta)} = \sum_{\substack{<\boldsymbol{x},\boldsymbol{y}>_i \\ \forall i\in I}} \frac{p_i^2}{n_i(\theta)} \frac{\|\ell(y^c(\theta; \boldsymbol{x}), \boldsymbol{y}) - F(\theta, \mathcal{S}_i)\mathbf{1}_{n_i(\theta)}\|_2^2}{n_i(\theta) - 1}, \tag{3}$$

where $n_i(\theta) = |\mathcal{S}_i(\theta)|$ is the size of the stratified sample, $\ell(y^c(\theta; \boldsymbol{x}), \boldsymbol{y})$ is a vector of loss functions for $< \boldsymbol{x}, \boldsymbol{y} >_i = \{< x_j, y_j >\in \mathcal{S}_i(\theta)\}$, and $\mathbf{1}_m$ is an m-dimensional vector of ones. Then the optimal sample sizes for stratum i that minimize $\widehat{\text{Var}}(F(\theta, \mathcal{S}(\theta)))$ are $n_i(\theta) := \lceil w_i(\theta) \times n(\theta)\rceil$ where

$$w_i(\theta) = \frac{p_i\hat{\sigma}_{F,i}(\theta)}{\sum_{l=1}^I p_l\hat{\sigma}_{F,l}(\theta)}, \tag{4}$$

and $n(\theta) = \sum_{i=1}^{I} n_i(\theta)$. Importantly, (4) achieves the optimal allocation if $\hat{\sigma}_{F,i}(\theta)$ are consistent estimators of $\sigma_{F,i}(\theta)$, but it is known to malfunction otherwise. Aside from the sampling allocation, an effective way to identify the $\mathcal{D}_i$'s is to find partitions with small within variance and large between variance [**?**].

2.3 Trust-region Optimization

Trust-region optimization methods are iterative methods that involve approximating the objective function by creating a local model in the trust-region and optimizing it during each iteration [**?**]. A neighborhood of size Δ_k around the incumbent solution θ_k forms the trust-region, i.e., $B_k = \{\theta : \|\theta - \theta_k\|_2 \leq \Delta_k\}$. The next candidate solution is one that minimizes the local model within this neighborhood. In the derivative-free context where direct observations of the derivative that unbiasedly estimate $\nabla \mathbb{E}_{X,Y}[\ell(y^c(\theta_k; X), Y)]$, an additional use of the trust-region is that the objective function at several adjacent $\theta \in B_k$ is estimated for a response surface construction using, e.g., interpolation or regression. Suppose the candidate solution sufficiently reduces the objective function. In that case, we update the incumbent and expand the next iteration's trust-region around the new θ_{k+1}, i.e., $\Delta_{k+1} \geq \Delta_k$ as a vote of confidence in the model and to speed up the search. When the candidate solution fails despite minimizing the local model, the next iteration continues from the incumbent with $\Delta_{k+1} < \Delta_k$. The convergence of this optimization algorithm is guaranteed as long as $\Delta_k \to 0$ as $k \to \infty$.

2.4 Adaptive Sampling

In a stochastic optimization trajectory, as the solution approaches optimality, correctly detecting progress demands higher precision on the estimated function value at the solution. Adaptive sampling involves adaptively changing the sample size to reduce variance so that we can start with small sample size and increase the sample size only when higher precision is required. This ensures that less budget is utilized initially when we might possibly be farther away from the solution. The sample size is increased so that we can balance the stochastic error with the optimality error [**?** **?** **?**]. Adaptive sampling with stopping time involves picking the smallest sample size that satisfies certain criteria.

A recently developed adaptive sampling based trust-region called ASTRODF by Shashaani et. al [**?**], that globally converges almost surely and obtains near canonical rate efficiency, creates the local model via interpolation based on stochastic outputs and uses the stopping time sample size such that the standard error is bounded by a function of the optimality gap, namely, $\kappa\Delta_k^2/\sqrt{\lambda_k}$, where κ is a positive constant that controls the magnitude of the sample size and $\lambda_k = \mathcal{O}((\log k)^{1+\epsilon})$ slowly deflates the sampling error as the algorithm proceeds, and also lower bounds the sample size at each iteration deterministically. Note, the sample size is ultimately stochastic and determined with information from the optimality error and search trajectory, namely, Δ_k. This stopping time ensures that initially when we might be further away from the solution the sample size will be small and as we approach the optimal solution the sample size goes on increasing to ensure better accuracy.

3 A Stratified Stochastic Trust-region Algorithm

In this study, we present ASTRODF-S to find the optimal wake decay coefficient. ASTRODF-S employs stratified sampling to ASTRODF [**?**]. In each iteration k, we evaluate the iterate θ_k by drawing samples with replacement from each stratum, denoted by $\mathcal{S}_i(\theta_k)$. To compute $\hat{\sigma}^2_{F,i}(\theta_k)$ at first, we set $n_i(\theta_k) = \lceil \lambda_k w_i \rceil$, where w_i is computed from (6), ensuring that $\sum_{i=1}^{I} n_i(\theta_k) = \lambda_k$. The total sample size is

$$n(\theta_k) = \min\left\{ n \geq \lambda_k : \frac{\hat{\sigma}^2_F(\theta_k)}{n} \leq \kappa \frac{\Delta_k^4}{\lambda_k} \right\}. \tag{5}$$

Figure 1 shows a general schematic for this study's adaptive sampling procedure. We first use the initial sample to estimate $\hat{\sigma}_F(\theta_k)$ and then if the condition in (5) is satisfied we stop, otherwise we add Δn points to the sample and recompute $\hat{\sigma}_F(\theta_k)$. We propose four methods for adaptive stratified sampling:

(a) Set $\Delta n = 1$, adding one point at a time till the condition in (5) is satisfied, and regarding the stratified

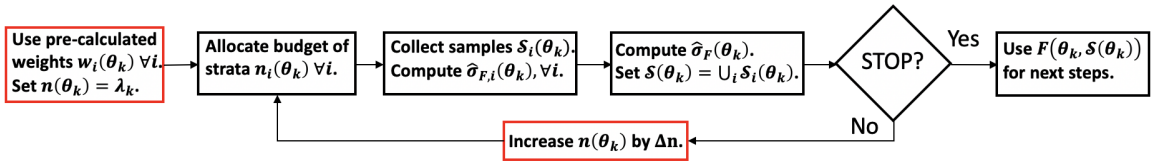

Figure 1: General adaptive sampling procedure, with the red boxes specifying variants in methods (a)-(d).

sampling weights use a proxy for (4) that allows for constant weights for all θ using $\hat{\sigma}_{Y,i}$, namely,

$$w_i(\theta_k) = w_i = \frac{p_i \hat{\sigma}_{Y,i}}{\sum_{\ell=1}^{I} p_\ell \hat{\sigma}_{Y,\ell}}. \tag{6}$$

Note that from (3) we use the fact that $\hat{\sigma}^2_{F,i}(\theta_k) \propto \widehat{\text{Var}}(\ell(y^c(\theta_k; X), Y)|X, Y \in \mathcal{S}_i(\theta_k))$ which is in part a function of the estimated variance of Y's and directly proportional to that. In other words, by minimizing $\widehat{\text{Var}}(Y)$ we confidently reduce $\widehat{\text{Var}}(F(\theta_k, \mathcal{S}(\theta_k)))$, even if not optimally, assuming $\widehat{\text{Var}}(Y|Y \in \mathcal{D}_i) = \hat{\sigma}_{Y,i}$ is homogeneous in $\mathcal{D}_i$. Note, $\mathcal{S}(\theta_k)$ is a stochastic object that affects $\text{Var}(F(\theta_k, \mathcal{S}(\theta_k)))$. The advantage of weights in (6) is that they are computed once but effective for any θ.

(b) Set $\Delta n = 1$ and use dynamic weights. Following [**?**] the dynamic weights are updated at the beginning of each iteration using previous iterates' information, as

$$w_i(\theta_k) = \frac{p_i \hat{\sigma}_{F,i}(\theta_{k-1})}{\sum_{\ell=1}^{I} p_\ell \hat{\sigma}_{F,\ell}(\theta_{k-1})}. \tag{7}$$

These dynamic weights use more recently obtained information and estimate $\sigma_{F,i}(\theta_k)$ with $\hat{\sigma}_{F,i}(\theta_{k-1})$.

(c) Use constant weights as in method (a) with $\Delta n = n_b$ which adds one batch of size n_b to the sample every time the condition in (5) is violated.

(d) Use dynamic weights as in method (b) with $\Delta n = n_b$.

Next, we seek effective stratification methods to find $\mathcal{D}_i$'s using turbulence intensity, following [**?**]. We create strata according to three methods (M1-M3), keeping the total number of strata equal I:

M1 Divide the data according to the quartiles of the input variable, i.e., $\mathcal{D}_i \subseteq \{< x_{[j]}, y_\cdot >\}_{j=\lceil (i-1)n/I \rceil}^{\lceil in/I \rceil}$, where $x_{[j]}$ is the order statistic of x's and $y_\cdot$ is the corresponding input response.

For the next two methods we use classification and regression trees (CART) to partition the available data. We randomly divide the data into $\mathcal{D}_v^{(train)}$ and $\mathcal{D}_v^{(test)} = \mathcal{D} \backslash \mathcal{D}_v^{(train)}$ for $v = 1, 2, \cdots, V$. Then we generate a tree using $\mathcal{D}_v^{(train)}$ to predict the responses from the predictors, as proposed by [**?**]. The same set of trees are used in M2 and M3 but for each tree we compute a different risk function G_v on $\mathcal{D}_v^{(test)}$, and then stratify the data via branches in $\mathcal{D}_{v^*}$ where $v^* = \arg\min_{v=1,2,\ldots,V} G_v$.

M2 Use the mean squared prediction error as the risk function $G_v := \|\ell(\hat{m}(x_j), m(x_j))\|_2^2 / |\mathcal{D}_v^{(test)}|$, where $< x_j, y_j > \in \mathcal{D}_v^{(test)}$, $m(x_j) = T^{-1} \sum_{t=1}^{T} y_{j,t}$, $y_{j,t}$ is the input response for the t-th turbine, and $\hat{m}(x_j)$ is the predicted response for x_j by CART.

M3 Minimizing the total variance of the response inputs decreases the variance within and increases the variance between, which is the goal of effective stratification. Total variance of the response inputs stratified into I strata on v-th tree is

$$(\hat{\sigma}_Y)_v^2 = \sum_{i=1}^{I} \frac{p_{i,v}^2 (\hat{\sigma}_{Y,i})_v^2}{n_{i,v}}, \tag{8}$$

where $p_{i,v}$ is the ratio of total points in stratum i created by $\mathcal{D}_v^{(test)}$, $(\hat{\sigma}_{Y,i})_v^2$ is the variance of the

response inputs, and $n_{i,v}$ is the size of stratum i of $\mathcal{D}_v^{(test)}$. Thus for the third method, we choose the risk function as $G_v = (\hat{\sigma}_Y)_v^2$.

4 Implementation and Results

For each of the cases corresponding to the combination of the stratification method and adaptive sampling method, the solver is run for ten macro-replications, each starting from a random initial solution θ_0 and terminating at a final solution upon exhausting of 5,000 function evaluations. The intermediate solutions reported from ASTRODF-S at various budget points are then re-evaluated by 100 post-replications independent of the seeds used to sample data for optimization to avoid the resulting optimization bias [**?**]. Common random numbers (CRN) are used both in the optimization and post-processing of the solutions for a reduced variance of the results when comparing each case. The average post-replicated values from each macro-replication are then used to construct the mean and 95% confidence interval (CI), which provides data to analyze each case's performance in accuracy, reliability, and speed of convergence. From the resulting mean and 95% CI convergence plots, we extract the following three performance metrics: $b' = \min\{b : \widehat{\mathbb{E}}_{X,Y}[\ell(y^c(\theta_{k(b)}; X), Y)] \leq 4,000\}$ or the first crossing of 10% optimality (with estimated optimal value from a separate larger dataset) of the mean convergence curve representing the accuracy, $h(b')$ or the half-width of the convergence curves at b' representing the variability/reliability, and τ being the total time in seconds until budget exhaustion representing the computational efficiency.

In all the experiments we let $I = 4$ and the batch size $\Delta n = 100$. For initial stratification using CART, we generate $V = 500$ regression trees, each of which has the ratio $|\mathcal{D}_v^{(train)}| : |\mathcal{D}_v^{(test)}| = 80 : 20$. For ASTRODF-S, the constant $\kappa = 10^8$ and $\lambda_k = \max\{40, 10(\log k)^{1.5}\}$ from pre-tuning the hyper-parameters.

Table 1: Comparing various adaptive sampling and stratification combinations' performance metrics (b': accuracy, $h(b')$: reliability, τ: computational efficiency) from 10 macro-replications on $\mathcal{D}$ - best values in each category in bold.

	Adaptive Sampling Method											
	(a)			(b)			(c)			(d)		
Stratification Method	b'	$h(b')$	τ	b'	$h(b')$	τ	b'	$h(b')$	τ	b'	$h(b')$	τ
M1	948	218	64	-	-	86	854	289	**55**	871	372	66
M2	**522**	120	60	-	-	120	1524	**116**	59	1090	254	135
M3	589	191	57	647	237	100	1138	255	56	-	-	63

Table 1 summarizes each case's performance concerning the three performance metrics introduced earlier. Some cases do not reach the 10% optimality mark within the set budget limit of $5,000$. Methods (b) and (d) use dynamic weights. Still, even though they directly target minimizing the objective function, they fail to cross the 10% optimality mark for several cases, implying that they do not perform better than the methods with static weights using an indirect proxy to minimizing the objective function. The same observation can be made from Figure 2 that plots the mean and 95% confidence interval (CI) of the convergence curves against the budget. Dynamic weights use the variance at the previous iterate to estimate the variance at the current iterate. In a stochastic setting for initial iterations, the variance at two consecutive iterates can be very different. In contrast, the number of iterations increases, the variance of iterates may behave more similarly. Computing weights at the beginning of each iteration also increases the computational time for the methods with dynamic weights. The reliability of the results is weaker with methods (b) and (d), suggesting sensitive estimates employed in the stratified sampling.

Even though batch increases in methods (c) and (d) cross the 10% optimality mark for more number of cases as compared to one-by-one increments in methods (a) and (b), they tend to exhibit slower convergence. Adding a batch instead of a single point when the condition in (5) is not satisfied implies more samples at the iterate and, hence, better performance. However, more samples also mean a higher budget utilization and slower convergence. Therefore, a trade-off exists between the accuracy and convergence rate, too apparent in Figure 2 where method (b) outperforms method (d) for M3 and vice versa for M2 and M3. The performance of adaptive sampling methods changes more notably as we change the initial stratification technique.

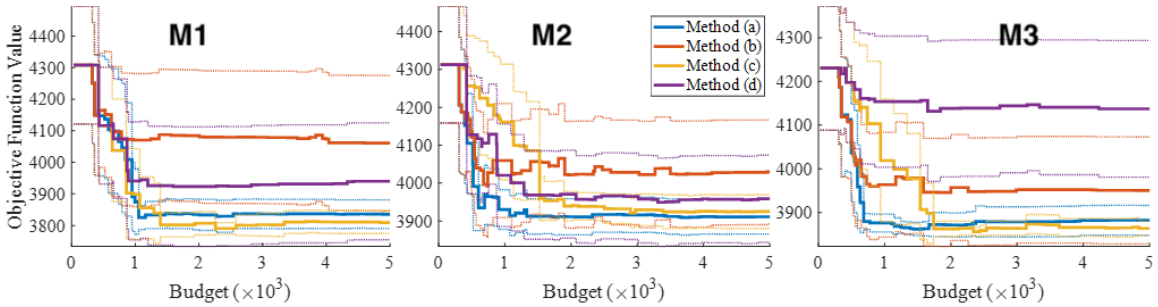

Figure 2: Mean-95% CI plots for all the cases considered in this study.

5 Conclusion

This study proposes methods to implement adaptive stratified sampling to calibrate the wake decay coefficient and compares their performance with various stratification methods. We show consistent variance estimators, even if leading to constant stratified sampling weights, significantly affect the derivative-free trust-region optimization. We plan to apply the proposed methods for stochastic gradient-based algorithms and integrate dynamic stratification with varying strata numbers at each iterate. The extension of this work for higher dimensions, e.g., calibrating turbines' thrust coefficient simultaneously, remains a future study.