The Authors response to the reviewers' comments

The authors would like to thank the referees for carefully reading our manuscript and for giving such detailed comments which substantially helped to improve the quality of the paper. In the revised version of the manuscript, we have tried to address all the points that were raised. The points that relate to the text's language and grammar are not discussed here. In the following, the comments will be discussed one by one.

- **Anonymous Referee #1**

  a) Page 2 states "Mainly two different methods are used to set the inflow boundary conditions for ABL flow simulations". There are various other methods that are used within the community such as using white noise, Mann spectrum, (concurrent) precursor methods, etc. For a detailed discussion of turbulence inflow generation methods see Wu, Annu. Rev. Fluid Mech. 2017. 49:23–49 and Stevens and Meneveau, Annu. Rev. Fluid Mech. 2017. 49:311-39 for applications of such methods to wind farm applications.

  It was not the intention of the authors to say that the mentioned methods are the only available solutions to generate an inflow boundary condition for CFD solvers. However, it was not probably clear in the text that we are discussing in the context of the RANS steady simulations. The inflow boundary generation, which is now in Section 3.1.1, refers to the suggested scientific works and does not include the explanation of the analytical function approaches.

  b) Figure 1: The description of figure 1, and in particular the description of the strangely oriented circle, took me quite some time to understand. The description of this figure should be improved.
  Both the description in the text and the figure itself are improved.

  c) Can the method only be applied when the actual computational domain is cylindrical?

  The method can be applied to other computational domain which does not have cylindrical shapes. It is now mentioned in the text. Moreover, the added Ishihara test case proves this too.

  d) Figure 3: It is unclear what the green grid cell is. Please clarify.

  The figure is changed (see figure 2), and the forest part is clear now.

  e) Figure 5: The caption states the velocity field at 40 meters is given. What is meant? The velocity field at 40 meters from the ground level indicated in panel 5a?

  It is now the figure 9. It shows the velocity field on a plane which has a 40 [m] perpendicular distance from the terrain's surface points (basically a slice of the domain that follows the terrain).

  f) Figure 6: Can it be indicated in the figure which calculations are performed simultaneously (primal and adjoint solver?) or are the different blocks in the figure performed sequentially? If so, in what order.

  The flow chart is updated, and it is now mentioned that all steps are performed sequentially. The optimizer decides if the gradient or the cost function value is needed. Then the corresponding path and steps will be followed.

g) 1. The section starting with "As it was explained in" on page 13 is rather vague. It is unclear to me what the smoothing function does exactly. I think the authors should explain this in more detail, so readers would be able to implement this part of the solution method by themselves.

2. The fitted profile in figure 3b is neither a logarithmic nor a power law. How exactly is the 1D inflow generating domain adjusted to achieve this? How strong can the deviation from the logarithmic/power law be before the code becomes unstable? How would one run the simulation in such a case (one may end up in an infinite loop in the diagram outlined in figure 6).

The profile in figure 3b is the profile in the middle of the forest and above it. The velocity for the heights above the forest is used for gradient validation, not the inflow calibration. The logarithmic or power-law function discussion is about the inflow boundary, not the profile inside of the domain.

The smoothing is now explained in Section 4.2. The output of the optimizer is first filtered to avoid sharp spikes. Then it is checked if it is close enough to a logarithmic or power-law function. If either of these functions is fitted and its coefficient of determination is above 0.96 the smoothed inflow from optimizer (not the fitted profile!), is accepted for the CFD solver. Otherwise, the optimization takes the last fitted profile and asks for a new gradient evaluation. In this way, the inflow boundary is not necessarily a logarithmic or power-law profile, and, moreover, it is not so unrealistic to be problematic for the solver. As mentioned in the reviewer's comments and in the new manuscript, alternatively, constraints or penalization term can be added to the objective function. This will be explored in future works when for instance the inflow turbulence properties are also considered as design parameters.

h) Does the computational time required by the adjoint solved depend on the initial wind direction and speed that is selected?

Of course, the adjoint solver inherits some of the properties of the original solver. The inflow wind speed and direction affect the computational time of the primal solver. Subsequently, the run time of the adjoint would be some how dependent on these parameters. In other word, it is case dependent.

i) 1. Figure 8: It is not entirely clear to me what exactly is meant by the reference profile in figure 8a.

It is the target inflow boundary. The legend of the plot is updated (see figure 11).

2. Page 14 just below figure 8 states: "Indeed, the output of the optimizer could be the exact reference profile if the convergence criterion was stricter." I am not sure what is meant here. When I look at figure 8b it seems that the calibration is performed on the velocity profile over the hill and this seems to match quite closely. When the velocity profile is calibrated at a specific location it could mean that any deviations with respect to measurements, caused by the used simulation method, would accumulate at another location in the domain (for example at the inflow). Is something like this happening? It would be good to discuss how the solution reacts to this.

Although a tighter error criterion would improve the accuracy of the optimized profile, it also increases the calibration iterations and subsequently the number of CFD solver calls. The sensitivity of wind speed at a certain point in the domain to a very small change in the inflow boundary is dependent on many parameters such as terrain complexity, wind direction, CFD model, etc., and cannot be easily generalized.

j) At the end of the manuscript the authors mention various extensions of the method. I am unsure whether various of these effects could be represented using this methods. Due to the use of the 1D domain to generate the vertical profile means that there is no information on the three-dimensional structure of the flow. It is known that for various properties of the atmospheric boundary layer capturing this three dimensional structure is crucial. I do not see how that can easily be incorporated in this method. Can the authors discuss in more detail what the effect of missing some of the three dimensional flow statistics, spatial flow correlations in the inflow are lost, is?

In general, adding the differentiation of the turbulence model with thermal stratification and Coriolis force to the current adjoint model would definitely improve the accuracy of the calibration. Moreover, the usage of the 1D inflow does not reflect the 3D structure of the flow. In order to overcome this limitation, this method could be extended to optimize the inlet BC as a spatially 2D field with three velocity components. However, a more sophisticated smoothing or penalization is needed to avoid having an unrealistic inlet field.

– **Anonymous Referee #2**

1. In general, I found the notation confusing and difficult to understand. In some cases I was unable to follow the derivations because of the confusing notation. Below are a few specific suggestions to clarify or reword the notation:
   a) It would be clearer if vectors and tensors were clearly identified in the text. Using Gibbs notation, either denote vectors as boldface italic and tensors in boldface or use vector symbols above. Alternatively, use index notation. For example, it's confusing to differentiate between the vector V and (scalar?) V (z).
   b) What is meant by the subscripts x, y, z in equation 6?
   c) V(z) is never defined. I assume it is the magnitude of the planar-averaged velocity vector.
   d) Equation 12: I'm not sure what is meant by (U, q)R. I would guess that U and R are vectors and that term can be expanded as UxR1 + UyR2 + UzR3 + qR4.
   e) In equations (14)-(16) What do δV and δp mean? What are J_Γ and J_Ω? Without knowing what this means, I was unable to follow the adjoint equation derivation.
   f) Equation 8: Why not use an equality instead of the right arrow? (equality is now used in Eq. (1))

The suggestions are considered in the new manuscript:

- The italic bold letter is a vector (e.g. $\boldsymbol{U}$, $\overline{\boldsymbol{U}}$) and the normal bold letter stands for a tensor ($\mathbf{D}$).

- The terms with δ removed and all the equations are represented and explained by derivative symbol, ∂.

- V(z) is not anymore in the text.

- Γ and Ω represent the boundary and the volume of the computational domain respectively. As a result, J_Γ and J_Ω are the part cost function which is dependent on flow state values on the boundary and the interior of the domain. This has been clarified in the manuscript.

2. I would significantly rework the structure of the paper to better integrate related ideas. As written, I'm not sure that someone could replicate this algorithm from the details of the paper.

   a) I would combine Section 3-5. Section 3 and 4 are closely related since the adjoint equations are needed to calculate the gradient. Section 5 is part of the adjoint equation derivation and should be included in Section 4. As it stands, it's hard to follow which sections are part of the adjoint-based optimization method description.

   b) Pg. 12 line 16 through pg. 13, Figure 6, and parts of pg. 9 lines 14-19 are related to the flow solver. I would put these details at the top of Section 2: "Flow Model" or with the details of the adjoint equations and gradient-based solver. It would be particularly helpful to have the k-ε turbulence model mentioned in Section 2.

   The suggestion that the structure of the paper could be improved was common among the reviewers. The new manuscript is restructured as follows: The gradient-based optimization and the theory of adjoint method are briefly presented in Section 2. The derivation of the adjoint equations and its BCs from the primal flow model is explained in Section 3. Finally, the numerical results and the conclusions are presented in Sections 4 and 5.

3. This method appears to be related to existing methods in meteorological applications (3DVar) or in existing wind energy papers (see the 4DVar implementation in Bauweraerts and Meyers, BLM, 2019). This is touched on in the introduction, but a more explicit discussion of how your method is related to existing approaches would be helpful.

   The existing 4DVar implementation by Bauweraerts and Meyers is one of the recent studies related to this subject. Instead of calibration of the inflow boundary, they optimize the whole initial field of the domain. They have developed an adjoint solver for a LES-based primal flow model and calibrate the initial flow field with the LIDAR measurements data from the whole of the domain. This is now mentioned in the introduction.

4. Is Section 2.1 used directly in the paper? My understanding is that these inflow conditions are calculated from your algorithm. This section may not really be necessary if that is the case. Also, how are equations (3) and (4) used simultaneously? What is z_{ref} ? How do you get n?

   This has been clarified in Section 4.2.

5. pg. 4, line 27: What do you mean by "error-prone"? Finite difference is simply too expensive to really be used in a gradient-based algorithm. I think that is sufficient justification for using an adjoint-based derivation.

   To avoid confusion, and based on some other comments, this has been deleted. The intention was to mention some of the disadvantages of the FD method. As said, the FD is too expensive for CFD gradient-based optimization. However, its usage should not completely be ruled out. For instance, there have been some studies on the application of FD to some selective terms in the discrete adjoint differentiation of CFD solvers (see for example *An aerodynamic design optimization framework using a discrete adjoint approach with OpenFOAM* by He et al., 2018).

6. Top of p. 5: It's not really multiplication, but the inner product of the state equations and the adjoint variables.

   Corrected.

7. p. 5, line 22: What is the effect of neglecting changes in eddy viscosity? Shouldn't changing the inflow conditions change the eddy viscosity of the simulation. How is assuming "frozen-turbulence" relevant to the RANS model?

   There is no doubt that the turbulence contribution to the adjoint equations cannot simply be ignored. In some applications, frozen-turbulence can produce the wrong sign for the local sensitivity (Zymaris et al., 2009; Papoutsis-Kiachagias et al., 2015). In the context of inflow calibration, the authors believe that the differentiation of the turbulence model is better to be included when the inflow turbulence properties are also design parameters, which can be explored in future works.

8. Beginning of Section 4: Adjoint methods are generic and applicable to many problems. I don't think it's necessary to point out the specific differences between your application and the Othmer's application.

   This paragraph is removed.

9. I'm not sure that Section 6 is really necessary. Showing that the gradient-based solver can find a solution is sufficient to demonstrate that the method works.

   As one of the reviewers has pointed out, the accuracy of the gradient computation is an important element of any gradient-based optimization. As a common practice in scientific studies, the validation of a newly implemented adjoint solver is presented.

10. It would be nice to show the application of the optimized inflow boundary condition for evaluating a specific site's wind resources or designing a wind farm. This is the real application and importance of this work, so I would make this a bigger point with an example.

    This point has been raised by other reviewers. Due to time limits and available sites in the project, this was not considered for this study. For sure this will be tried in the future.

– **Anonymous Referee #3**

1. I believe the paper would benefit from the addition of some more references in selected places, i.e.:

   a) Section 1 - lines 20-25, where the authors mention that their adjoint approach has not been applied before in the framework of wind resource assessment. It would be nice to add some references to recent works where adjoint optimization is used in the context of wind energy. e.g. blade shape optimization: Dhert, Ashuri, Martins, Wind Energy 2017 wind-farm control: Goit & Meyers, J Fluid Mech 2016; Munters & Meyers, Phil Trans Roy Soc A, 2017; Vali et al., Control Engineering Practice 2019 wind-farm layout optimization: King et al., Wind Energy Science 2017

   b) Section 3 - 3.1.1, where the adjoint method is introduced through the formal Lagrange method. A reference to e.g. Hinze, Michael, et al. Optimization with PDE constraints. Vol. 23. Springer Science & Business Media, 2008 seems appropriate. Also, the explanation of the necessity and philosophy of the adjoint method is quite poor. The authors could improve this by explicitly mentioning that eq. 11 is expensive since the term $\frac{d\psi}{d\alpha}$

requires a PDE simulation for every dimension in alpha, and showing explicitly that this term drops out in the adjoint method. I advise the authors to either expand on their explanation of the adjoint methodology, or to refer in to references where it is explained in detail.

The recommended references are added to the corresponding sections. The adjoint method explanation is improved.

2. page 2, lines 3 - 4: The authors first mention the disadvantage of the FD method as being error-prone. In context of the current manuscript, this is misleading in my opinion. The loss of accuracy due to finite-precision arithmetic can be circumvented by using a complex-step finite differentiation. Also, since the authors use the continuous adjoint method without a grid convergence study, the computed adjoint gradients could certainly be less accurate than a finite-difference approximation. I feel the authors should remove this claim, or at least put more emphasis on the fact that FD computational cost scales with the input dimensionality, whereas this is not the case for the adjoint method.

   It has been mentioned in the manuscript that the complex-step method would circumvent the difficulties of FD. However, it should also be noted that the implementation of a complex-step method on a large code which is not written with complex variables (e.g. OpenFOAM) is not straight forward too.

   In the manuscript, the emphasis is put now on the advantages of the adjoint method in terms of the computational cost for a large number of design parameters.

3. page 3, line 10: ABL flow simulations –> ABL RANS simulations. Please add the term RANS here, to avoid confusion with the generation of inflow conditions for turbulence-resolving simulations (DNS/LES), which is a whole research field on its own.

   Corrected. It is now in section 3.1.1.

4. page 11, line 1: "For gradient evaluation, the 1D velocity profile inflow is rotated by 30°". This seems like a very large step for a finite-difference gradient approximation. Why did the authors not take a much smaller rotation, e.g. of 1°? Intuitively, 1°still seems large enough to avoid influence of round-off errors.

   Please be aware that, the 30°is not the step-size. First, a simulation is carried out with WD=270°. The velocities, above the forest, are taken as pseudo measurements. Then, using these measurements, the adjoint gradient is computed for the simulation with WD=240°. The finite difference gradient is also carried out for WD=240°in which the step-size is 0.2°.

5. page 11, around line 10: The statement: "most importantly, their signs show that they can be used for the purpose of gradient-based calibration" is misleading. The authors seem to state that, in laymans terms, having a gradient that point approximately in the right direction is sufficient for optimization. However, this claim should be nuanced. Gradient inaccuracies can severely impact the performance, stability and convergence of a given optimization algorithm. For instance, in quasi-Newton methods this could lead to instabilities because of poor Hessian

approximations, and in CG methods this could lead to non-conjugate search directions in successive iterations. Furthermore, related to comment nr. 2 and comment nr. 4, I feel the authors should be careful in attributing discrepancies between adjoint and FD gradients to inaccuracies of the FD gradient. Intuitively, I would expect the frozen turbulence assumption and the grid resolution (combined with continuous adjoint approach) to be the main reasons for discrepancies.

No doubt the accuracy of the gradient is as important as its sign due to the reasons which are mentioned above. Otherwise, we would not put such a section in the manuscript. However, the words should have been chosen more carefully.

The point regarding the FD error is considered in the manuscript.

6. page 12, line 18: The authors mention some facts about computational cost of their simulations. These facts can be made more illustrative by also explicitly mentioning the wall-time of a primal flow run, and explaining why the run-time of the adjoint solver is 60% of the primal (e.g. because the adjoint equations are linear)?

   The wall-time of the primal run is given in the new manuscript. Please also refer to the following response:

   - Anonymous Referee #4, question: 10

7. (suggestion) page 13, line 5: "The optimizer may ask for a cost function evaluation with a new inflow boundary which is highly unrealistic for an ABL domain. ... the curve fit capability is used to smooth and fit the new inflow to a boundary which has a log/power law characteristic." The authors manually post-process the new iterate of inflow conditions during the optimization process. Although I agree that it is undesirable to run RANS with unrealistic inflows during optimization, this manual post-processing will can have a significant detrimental impact on the convergence of the optimization. Since this post-processing imposes a log/power law profile, it seems more natural to directly use the parameters for such log/power profiles as decision variables, in contrast to optimizing the individual inflow velocities at every height. This would directly inform the optimizer of the desired log/power law profile, and could improve convergence a lot.

   As it is explained in the manuscript and before in this document, the point of smoothing is to avoid enforcing the boundary to have either a logarithmic or power-law function shape. Otherwise, optimizing the function parameters would be straightforward. As in the Ishihara case, the inflow boundary can be neither a logarithmic nor a power-law, but still being able to reproduce the measurements with acceptable tolerance. Of course, the approach can be improved by for example adding corresponding constraints to the objective function.

8. (suggestion) page 13, line 18: "This can be explained by the fact that in early iterations the derivative of cost function wrt WD is much bigger than wrt inflow". This spike might be avoided and convergence could possibly be improved by using quasi- Newton optimization methods (e.g. BFGS), this is a suggestion for future applications of the methodology.

   Thank you for the helpful suggestion. We think also this will help.

– **Anonymous Referee #4**

Specific comments:

1. The purpose of section 2 is not clear. Do you want to give general description of inflow boundary condition and forest effect, or are these the techniques you will use in your study? If it is the former then they should go to Introduction, if it is the later, then you should explain them together with the discussions in Section 3 and Section 4.

   This part is now moved to the section 4.2 where the optimization steps is discussed.

2. You have Section 3, subsection 3.1 and sub subsection 3.1.1, but no following subsection (e.g. 3.2 etc) or following sub subsection (e.g. 3.1.2 etc). So, I suggest that you put all the contents in this section under a single section heading without any subsection. But my main concern for this section is once again, it is not clear whether you are trying to explain a general method for gradient evaluation in optimization or is it for your specific problem? The section lacks explanation that may be necessary for some one not familiar with Gradient-based optimization. Therefore, some more discussion will be required. For example, what is the role of design variable α, what will your algorithm do to optimize it and why can you write Eq. (11). Furthermore, last sentence in section 3.1 (line 28 and 29) will not come as obvious to many readers.

   The new restructured manuscript should not have this problem. Moreover, both the gradient-based (section 2) and the CG optimizer (4.2) are explained in the paper.

3. Section 4, 1st Paragraph: The purpose of this paragraph is not clear. Summarizing the differences between your work and that of earlier work is not really necessary. Because the two works deal with different optimization problem, all three differences stated in the manuscript are obvious.

   The paragraph is removed.

4. The forest model, Eq. (18) should be a part of original Lagrangian and should also appear in Eq. (15) and (16).

   Corrected.

5. Why do you not have convective (and cross-convective) terms in Eq. (19)? I still see them in Eq. (16). If it is an error, then please correct it. If you have a proper reason why they can be neglected, you must explain that.

   It is not used anymore in the new manuscript (see Eq. (21)) but

   $$-2\mathbf{D}(\overline{\boldsymbol{U}})\boldsymbol{U} = -\nabla\overline{\boldsymbol{U}}\cdot\boldsymbol{U} - (\boldsymbol{U}\cdot\nabla)\overline{\boldsymbol{U}}$$

6. It is not clear what $\omega_j$is.

   It is the volume of the cell in which the measurement is located.

7. It has become more common (at least in academic researches) to use large-eddy simulations (LES) for ABL and wind farm simulations. But authors preferred to use RANS in their work. Can you please discuss why you chose RANS? Was it because RANS is cheaper or was it because it is easy to implement adjoint equations for RANS problem? In reality both inflow profiles and measured velocity at the measurement points VM will vary with time. Furthermore, dynamic behavior of flow field as well and wind farm are receiving more interest in wind energy community (e.g. farm level controller). So, it seems LES would have been preferred simulation method.

   As it is mentioned in the new manuscript, there are some studies that use measurements to calibrate the LES-based models (Bauweraerts and Meyers, 2018). However, due to the computational cost, the most common CFD solvers in the wind energy industry is still steady-state RANS-based models. Also, RANS is the model of choice here because of the scope of the current project. We completely agree the LES is gaining momentum due to its superior predictions in many cases compared to RANS.

8. Pg. 12, Line 14–15: It is not clear why you used velocity profile from a reference simulation instead of the velocity measured by the met-mast. Also, if your reference simulation and actual simulation were performed in the similar condition, then you will obviously get good optimization result. Please provide further information regarding this issue.

   Indeed, the main idea of this work is using measurements to calibrate the inflow boundary of the solver. For the Ishihara case, the measured wind speed U_x was used for the optimization process. Since it is a wind tunnel experiment, it is fair to assume the wind direction to be the x axis. Although the wind speed at six different heights was available for the Kassel case, the wind direction data was present at only two heights. Therefore, we used the pseudo measurement to prove the validity of the methodology. However, the profile from which the pseudo measurements are selected essentially follows the trend of the real measurements of the domain. Please see the comparison in the figure below, which has not been included in the paper because it may confuse the reader.

9. Pg. 12, Line 16–18: You need to provide more information about conjugate- gradient algorithm.

   An elaborate explanation has been added to sec. (4.2).

10. Pg. 12, Line 19: Are you sure that the run-time of adjoint equations is 60% of the primal equations? For most work that I am aware of and from my personal experience, adjoint equations always took longer time to simulate.

    Yes, in most of the adjoint simulations that we have carried out with this adjoint solver the run-time is 60-75% of the original flow run-time. A couple of points should be considered: a) The main driver of this adjoint solver is the source term which is the difference between the measurement and simulated results. This source term appears only for a few cells in which the measurements are located. Though, comparing to other adjoint solvers for shape and topology optimization this can be seen as a simpler solver. b) The turbulence is not differentiated; meaning there are no adjoint turbulence equations to be solved.

    In the new manuscript, instead of a certain percentage, a range is given.

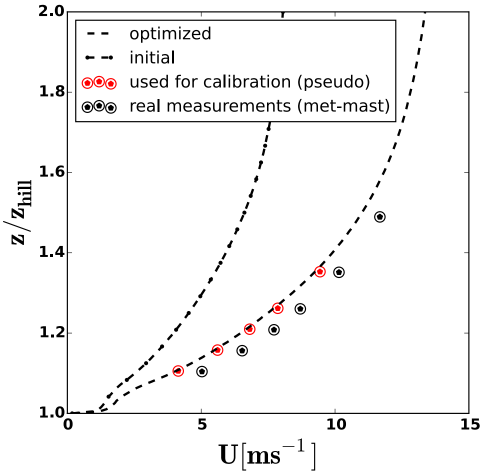

Figure A1. Comparison of the velocity profile over the hill from which the pseudo measurements are chosen and the velocity speeds measured by the met-mast.

11. Pg. 13, Line 4–10: You are fitting the inlet boundary condition from the optimization to a logarithmic or a power law. This may not be a good idea, if you want to exploit the full potential of your optimization scheme. Therefore, instead why do not you add some sort of constrain to your system or add appropriate penalty term to the cost functional?

This point is discussed before in this document. Please refer to the following responses:
- Anonymous Referee #1, question: g
- Anonymous Referee #3, question: 7

12. I do not think you have presented sufficient result to consider this manuscript as a technical paper. You only have figure 8 as the results for one simulation case. Please define and perform optimization for more simulation cases. Also, you need to provide further discussion of your result.

The results for the Ishihara et al. test case is also added in the new manuscript. We hope this is sufficient for the scope of the paper. In future work, we will investigate more cases with different complexity. Our main aim here is to provide the general framework and initial case studies.

Adjoint-based Calibration of Inlet Boundary Condition for Atmospheric CFD Solvers

Siamak Akbarzadeh[1], Hassan Kassem[2], Renko Buhr[1], Gerald Steinfeld[1], and Bernhard Stoevesandt[2]

[1]ForWind, Center for Wind Energy Research, Carl von Ossietzky University Oldenburg, Küpkersweg 70, 26129 Oldenburg, Germany

[2]Fraunhofer Institute for Wind Energy Systems - Fraunhofer IWES, Küpkersweg 70, 26129 Oldenburg, Germany

Correspondence: Siamak Akbarzadeh (siamak.akbarzadeh@uni-oldenburg.de)

Abstract. A continuous adjoint solver is developed for optimization of the inlet velocity profile boundary condition for CFD simulations of the neutral atmospheric boundary layer (ABL). The adjoint solver uses interior domain wind speed observations to compute the gradient of a calibration function with respect to inlet velocity speed and wind direction. The solver has been implemented in the open source CFD package OpenFOAM. The sensitivities computed by the newly developed adjoint solver are validated against the second order finite-difference method. Furthermore, the DAKOTA optimization package is coupled with OpenFOAM, and a number of numerical studies are carried out including the calibration of the inlet velocity profile of a 3D complex domain.

1 Introduction

The wind energy industry is growing very fast and a comprehensive site assessment is a key factor in planning, installation and performance of wind farms. As a result, the interaction of the atmospheric boundary layer (ABL) flow and wind turbines is one of the most important aspects of the site assessments for wind farms. Over the last decades with development of powerful computers the Computational Fluid Dynamics (CFD) has become one of the leading tools for micro-scale simulation of the wind flow over complex terrains. However, in general, developing algorithms that capture all the physics of such complex flow regimes is an ongoing research in CFD.

In CFD simulations of atmospheric flows the correct boundary conditions (BCs) are often unknown but measurement data (e.g wind speed, wind direction etc.) within the area of interest is available. Using the measurement data and an inverse analysis, the unknown BCs can be obtained with an optimization algorithm. This approach is called open boundary optimization and has been successfully tried in oceanography (Seiler, 1993; Chen et al., 2013) and numerical wind prediction (NWP) models (Schneiderbauer and Pirker, 2011). Another approach is to use observations and statistical analysis (Glover et al., 2011) to calibrate the CFD model parameters (e.g., inflow and turbulence model constants) .

The solution to an optimization problem can be found with different methods. However, the methods such as genetic algorithm and evolutionary strategies (Davis, 1991; Michalewicz, 1996) require a large number of function evaluations which in CFD applications can be computationally very expensive. Alternatively, the gradient-based optimizers (Ruder, 2016) use the derivative of cost function with respect to (w.r.t.) the design parameters. Then, the optimal solution can be obtained using the gradient and a relatively less cost function evaluation. Gradient computation with the finite-difference (FD) method is relatively simple. However, for a high number of design parameters is prohibitively expensive. In the adjoint method, the sensitivity of the objective function can be calculated independently from the number of parameters, and this considerably reduces the cost of computation.

Since the first application of adjoint method into compressible CFD models by Pironneau in 1974, the adjoint based optimization methods have been extensively used in shape and topology optimization (Jameson et al., 1998; Kämmerer et al., 2003; Li et al., 2006; Othmer and Grahs, 2005; Othmer et al., 2006). The adjoint gradient computation can be categorized into two main groups: 1) continuous adjoint; 2) discrete adjoint. In the continuous method (Jameson, 1988) the adjoint equations are first derived and then discretized. In the discrete adjoint (Giles and Pierce, 2000), using the chain rule, the adjoint solver is derived by line-by-line differentiation of the discretized original (primal) flow solver code. The discrete adjoint differentiation can also be automated using Algorithmic Differentiation (AD) (Griewank and Walther, 2008).

In principle, both methods can be applied to any algorithm and model which is continuously differentiable. However, the manually derived discrete adjoint differentiation of big CFD codes is laborious and error-prone. Although, the AD tools can be seen as an interesting solution to this problem, their application to the codes which are written in high-level languages (e.g. C++) is still limited in terms of memory requirement and performance. A comparison of these two adjoint approaches can be found here (Giles and Pierce, 2000; Nadarajah and Jameson, 2000, 2001).

The discrete adjoint version of OpenFOAM based solvers has been presented before (Towara and Naumann, 2013; Towara et al., 2015). More recently a hybrid approach has also been introduced (He et al., 2018) in which some parts of the code are differentiated by finite-difference, and better performance is reported in comparison to the pure discrete adjoint version of the code. Despite all these improvements, the continuous adjoint version of OpenFOAM solvers are still more popular.

The adjoint method has been well applied to different problems in atmospheric science such as wind turbine blade shape optimization (Dhert et al., 2017), wind-farm control (Goit et al., 2016; Munters and Meyers, 2017; Vali et al., 2019) and wind-farm layout optimization (King et al., 2017). Bauweraerts and Meyers (2018) have used the adjoint method and LES-based 4D-Var data assimilation to estimate the turbulent flow field of an atmospheric boundary layer domain from LIDAR data. Although compared to the RANS models, the LES can provide more accurate predictions of the flow field, it still mainly used as a research tool due to its demanding computing power.

The effect of inflow boundary wind speed and its direction are significant parameters in ABL CFD simulation, but they are not often known. The focus of this paper is on the adjoint gradient based calibration of the inlet velocity profile and inflow wind direction (WD) for a RANS-based ABL flow, which is a very common CFD solver in the wind energy industry, with only a few wind speed measurements from a met mast at the site. The available continuous adjoint solver of OpenFOAM CFD tool package (*adjointShapeOptimizationFoam*), which is for topology optimization of duct flows, is further developed to compute the gradients.

The structure of the paper is as follows: The gradient-based optimization and the theory of adjoint method are briefly presented in Section 2. The derivation of the adjoint equations and its BCs from the primal flow model is explained in Section 3. Finally, the numerical results and the conclusions are presented in Sections 4 and 5.

2 Gradient-Based Optimization and Adjoint Method Theory

In gradient-based optimization or calibration, one needs to compute the gradient of a smooth cost function, J, with respect to the design parameters, α, at each design step,

$$\alpha^{n+1} = \alpha^n + \mathbb{A}\left(\alpha^n, \left(\frac{dJ}{d\alpha}\right)\right) = \alpha^n + (\Delta\alpha)^n \tag{1}$$

where $\mathbb{A}$ is an optimization algorithm operator that returns a perturbation $\Delta\alpha$ to the current design α^n. The procedure is repeated until a convergence criterion is reached. The design parameters, α, are chosen based on the optimization problem and its parameterization.

In CFD applications, computing the term $\dfrac{dJ}{d\alpha}$ includes the differentiation of the steady-state PDE governing equation of the flow,

$$\boldsymbol{R}(\psi,\alpha) = 0 \rightarrow \frac{\partial \boldsymbol{R}}{\partial \psi}\frac{\partial \psi}{\partial \alpha} + \frac{\partial \boldsymbol{R}}{\partial \alpha} = 0 \tag{2}$$

where $\boldsymbol{R}$ is the residual vector of the discretized flow equations that is driven to zero and ψ stands for state variables (velocity, pressure, temperature, etc.). The Eq. (2) results in a linear system,

$$\frac{\partial \boldsymbol{R}}{\partial \psi}\frac{\partial \psi}{\partial \alpha} = -\frac{\partial \boldsymbol{R}}{\partial \alpha} \tag{3}$$

in which the term $\frac{\partial \boldsymbol{R}}{\partial \psi}$ is Jacobian and $\frac{\partial \psi}{\partial \alpha}$ represents the perturbation of flow fields. Using the chain rule, the total derivative can be then computed by

$$\frac{dJ}{d\alpha} = \frac{\partial J}{\partial \alpha} + \frac{\partial J}{\partial \psi}\frac{\partial \psi}{\partial \alpha}. \tag{4}$$

Several methods can be used to compute the gradient from Eq. (4). For instance the complex variable technique, which overcomes the problem of choosing step-width in the finite-difference method, or the forward mode (tangent linearization) application of Algorithmic Differentiation (AD). However, all these methods are computationally expensive when the design space is large. This is due the fact that in Eq. (4) the term $\frac{\partial \psi}{\partial \alpha}$ requires an expensive PDE simulation, which satisfies Eq. (3), for every dimension in the design space, α_i. As will be shown in the following, in the adjoint method the sensitivity can be obtained without computing this expensive term.

From Eq. (3) we can write the perturbation term as

$$\frac{\partial \psi}{\partial \alpha} = -\left(\frac{\partial \boldsymbol{R}}{\partial \psi}\right)^{-1}\frac{\partial \boldsymbol{R}}{\partial \alpha} \tag{5}$$

leading to

$$\frac{dJ}{d\alpha} = \frac{\partial J}{\partial \alpha} - \frac{\partial J}{\partial \psi}\left(\frac{\partial \boldsymbol{R}}{\partial \psi}\right)^{-1}\frac{\partial \boldsymbol{R}}{\partial \alpha} = \frac{\partial J}{\partial \alpha} - \left[\frac{\partial J}{\partial \psi}\left(\frac{\partial \boldsymbol{R}}{\partial \psi}\right)^{-1}\right]\frac{\partial \boldsymbol{R}}{\partial \alpha} \tag{6}$$

The terms in the bracket can be identified as the adjoint system of equations from which the adjoint variable, $\overline{\psi}$, can be introduced as:

$$\overline{\psi}^T = \frac{\partial J}{\partial \psi}\left(\frac{\partial \boldsymbol{R}}{\partial \psi}\right)^{-1} \quad \text{or} \quad \left(\frac{\partial \boldsymbol{R}}{\partial \psi}\right)^T \overline{\psi} = \frac{\partial J}{\partial \psi}. \tag{7}$$

In this way, instead of solving a PDE simulation for every design variable, the adjoint system of equations need to be solved only once. As a result, the computational cost of the gradient becomes independent of the number of design parameters (Giles et al., 2003; Mavriplis, 2007; Nielsen et al., 2010):

$$\frac{dJ}{d\alpha} = \frac{\partial J}{\partial \alpha} - \overline{\psi}^T \frac{\partial \boldsymbol{R}}{\partial \alpha}. \tag{8}$$

3 Derivation of the Continuous Adjoint Solver for ABL Inflow Calibration

3.1 Flow Model

The ABL flow model for cases of neutral stratification consists of steady-state Reynolds Averaged Navier-Stokes (RANS) for incompressible fluid flows (Rebollo and Lewandowski, 2014) which results in the following equations for momentum and continuity:

$$(\boldsymbol{R}_1, \boldsymbol{R}_2, \boldsymbol{R}_3)^T = (\boldsymbol{U} \cdot \nabla) \cdot \boldsymbol{U} + \nabla p - \nabla \cdot (2\nu_{eff} \mathbf{D}(\boldsymbol{U})) \tag{9}$$

$$\boldsymbol{R}_4 = -\nabla \cdot \boldsymbol{U} \tag{10}$$

The variables $\boldsymbol{U}$ and p are the state variables velocity vector and pressure, ν_{eff} stands for the sum of kinematic and turbulent viscosity and $\mathbf{D}$ is the rate of strain tensor, $\mathbf{D} = \frac{1}{2}(\nabla \boldsymbol{U} + (\nabla \boldsymbol{U})^T)$. Throughout this work, the standard $k - \varepsilon$ turbulence model with the addition of a forest model is used.

3.1.1 Inflow Boundary

The properties of the inflow boundary have an important effect on the solution of the interior domain. With the increasing application of LES, hybrid RANS-LES and DNS methods, the inflow turbulence generation has become the subject of many researchers in recent decades. For a review of such methods and their application in wind energy, we refer to (Wu, 2017; Stevens and Meneveau, 2017). However, due to computational cost, it is still popular in the wind energy industry to use the RANS model with an inflow boundary obtained from either an analytical formula or a one-dimensional (1D) simulation. The idea of the latter method is to solve a zero-pressure gradient equation for a 1D domain with periodic boundary conditions in the stream- and span-wise directions (Chang et al., 2018):

$$\frac{\partial \overline{\boldsymbol{U}'_x \boldsymbol{U}'_z}}{\partial z} = 0 \tag{11}$$

where x (horizontal) and z (vertical) are the Cartesian coordinates. The inflow profile and its turbulent characteristics are obtained from this 1D simulation and then are mapped to the 3D inlet boundary.

3.1.2 Forest Effect

Forest canopies modify the available free volume of the terrain domain and introduce an explicit drag term to the momentum equation as below:

$$\boldsymbol{F}_D = -\frac{1}{2}\rho C_d A(z)|\boldsymbol{U}|\boldsymbol{U} \tag{12}$$

with density ρ, leaf-level canopy drag coefficient C_d and leaf area density $A(z)$. The effects of the forest in the turbulence models such as $k-\varepsilon$ for ABL flows has been extensively discussed in the literature and several formulas are presented to make the turbulence model consistent with the modified momentum equation. For more details, the reader may be referred to (Lopes da Costa, 2007).

3.2 Adjoint Model

Calibration algorithms seek to maximize the agreement between simulation outputs and measurements. In the context of ABL based model calibration the data are often wind speed and direction at one or more locations of a potential wind farm site. The CFD-based calibration can be formulated as a constrained optimization problem with a scalar objective function:

$$\text{minimize} \;\; J(\boldsymbol{U}_M, \boldsymbol{U}_S, \alpha) = \sum_{i=1}^{k} [\boldsymbol{U}_{M_i} - \boldsymbol{U}_{S_i}]^2 \quad ; \quad \text{subject to} \;\; R_{1,2,3,4}(\boldsymbol{U}, p, \alpha) = 0 \tag{13}$$

where R stands for the spatial residual of the flow equations with $\boldsymbol{U}$ and p the discretized velocity and pressure, respectively. $\boldsymbol{U}_{M_i}$ and $\boldsymbol{U}_{S_i}$ are the measured and simulated wind velocities at the same location in the domain. The variable α represents the design variables which are considered to be the velocity at inlet faces of the CFD mesh through this work.

The derivation of adjoint equations as in the preceding section is arguably the most straightforward way to introduce the adjoint equations and understand their advantages. The first developments for using the adjoint equations in CFD applications were done using a Lagrange multiplier argument (Hinze et al., 2008). From this point of view, the inner product of the PDE of flow equations and a new set of variables vanishes the variations of state variables, $\frac{\partial \boldsymbol{U}}{\partial \alpha}$ and $\frac{\partial p}{\partial \alpha}$. By introducing the adjoint variables $\overline{\boldsymbol{U}}$ and $\overline{p}$ for adjoint velocity and adjoint pressure respectively, the cost function can be reformulated to a Lagrange function as

$$L := J + \int_{\Omega} (\overline{U}_x R_1 + \overline{U}_y R_2 + \overline{U}_z R_3 + \overline{p} R_4)\, d\Omega = J + \int_{\Omega} (\overline{\boldsymbol{U}}, \overline{p}) \boldsymbol{R}\, d\Omega \tag{14}$$

where $\overline{U}_x$, $\overline{U}_y$ and $\overline{U}_z$ are the adjoint velocity components and Ω is the flow domain. For the sensitivities of the cost function w.r.t. the design parameters, we have to compute the total variation of L:

$$\frac{dL}{d\alpha} = \frac{\partial J}{\partial \alpha} + \frac{\partial J}{\partial \boldsymbol{U}}\frac{\partial \boldsymbol{U}}{\partial \alpha} + \frac{\partial J}{\partial p}\frac{\partial p}{\partial \alpha} + \int_{\Omega} \left[(\overline{\boldsymbol{U}}, \overline{p}) \left(\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{U}}\frac{\partial \boldsymbol{U}}{\partial \alpha} + \frac{\partial \boldsymbol{R}}{\partial p}\frac{\partial p}{\partial \alpha} \right) + (\overline{\boldsymbol{U}}, \overline{p}) \frac{\partial \boldsymbol{R}}{\partial \alpha} \right] d\Omega \tag{15}$$

Choosing the Lagrange multipliers $\overline{\boldsymbol{U}}$ and q such that the variation w.r.t. the state variables vanishes, leads to

$$\frac{\partial J}{\partial \boldsymbol{U}}\frac{\partial \boldsymbol{U}}{\partial \alpha}+\frac{\partial J}{\partial p}\frac{\partial p}{\partial \alpha}+\int\limits_{\Omega}\Big[(\overline{\boldsymbol{U}},\overline{p})(\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{U}}\frac{\partial \boldsymbol{U}}{\partial \alpha}+\frac{\partial \boldsymbol{R}}{\partial p}\frac{\partial p}{\partial \alpha})\Big]\,d\Omega=0 \tag{16}$$

Then the sensitivity of the cost function can be given by

$$\frac{dL}{d\alpha}=\frac{\partial J}{\partial \alpha}+\int\limits_{\Omega}(\overline{\boldsymbol{U}},\overline{p})\frac{\partial \boldsymbol{R}}{\partial \alpha}\,d\Omega \tag{17}$$

which excludes the state variables sensitivities.

The theory presented here is based on the work of C. Othmer 2008 which derives an adjoint solver for topology optimization of duct flows to reduce the pressure loss between inlet and outlet boundaries. By neglecting the turbulent viscosity variation, assuming the "frozen-turbulence" hypothesis, and replacing the derivative of Eq. (9) with forest source term and Eq. (10) into Eq. (16) gives:

$$\begin{aligned}\frac{\partial J}{\partial \boldsymbol{U}}\frac{\partial \boldsymbol{U}}{\partial \alpha}+\frac{\partial J}{\partial p}\frac{\partial p}{\partial \alpha}+\int\limits_{\Omega}\Big[\overline{\boldsymbol{U}}\cdot(\frac{\partial \boldsymbol{U}}{\partial \alpha}\cdot\nabla)\boldsymbol{U}+(\boldsymbol{U}\cdot\nabla)\frac{\partial \boldsymbol{U}}{\partial \alpha}-\nabla\cdot(2\nu_{eff}\mathbf{D}(\frac{\partial \boldsymbol{U}}{\partial \alpha}))+\frac{1}{2}C_DA\big(\cancelto{\sim 0}{|\frac{\partial \boldsymbol{U}}{\partial \alpha}|}\boldsymbol{U}+|\boldsymbol{U}|\frac{\partial \boldsymbol{U}}{\partial \alpha}\big)\Big]\,d\Omega\\ -\int\limits_{\Omega}\overline{p}\nabla\cdot\frac{\partial \boldsymbol{U}}{\partial \alpha}d\Omega+\int\limits_{\Omega}\overline{\boldsymbol{U}}\cdot\nabla\frac{\partial p}{\partial \alpha}\,d\Omega=0\end{aligned} \tag{18}$$

Decomposition of parts into interior domain, Ω, and its boundaries, Γ, leads to reformulation of Eq. (18) as follows

$$\begin{aligned}&\int\limits_{\Gamma}\big[\overline{\boldsymbol{U}}\cdot\boldsymbol{n}+\cancelto{0}{\frac{\partial J_\Gamma}{\partial p}}\big]\frac{\partial p}{\partial \alpha}\,d\Gamma+\int\limits_{\Gamma}\big[\boldsymbol{n}(\overline{\boldsymbol{U}}\cdot\boldsymbol{U})+\overline{\boldsymbol{U}}(\boldsymbol{U}\cdot\boldsymbol{n})+2\nu_{eff}\boldsymbol{n}\cdot\mathbf{D}(\overline{\boldsymbol{U}})-\overline{p}\boldsymbol{n}+\cancelto{0}{\frac{\partial J_\Gamma}{\partial \boldsymbol{U}}}\big]\frac{\partial \boldsymbol{U}}{\partial \alpha}\,d\Gamma\\ &+\int\limits_{\Gamma}\big[-2\nu_{eff}\boldsymbol{n}\cdot\mathbf{D}(\frac{\partial \boldsymbol{U}}{\partial \alpha})\cdot\overline{\boldsymbol{U}}\big]\,d\Gamma+\int\limits_{\Omega}\big[-\nabla\cdot\overline{\boldsymbol{U}}+\cancelto{0}{\frac{\partial J_\Omega}{\partial p}}\big]\frac{\partial p}{\partial \alpha}\,d\Omega\\ &+\int\limits_{\Omega}\big[-\nabla\overline{\boldsymbol{U}}\cdot\boldsymbol{U}-(\boldsymbol{U}\cdot\nabla)\overline{\boldsymbol{U}}-\nabla\cdot(2\nu_{eff}\mathbf{D}(\overline{\boldsymbol{U}}))+\frac{1}{2}C_DA|\boldsymbol{U}|+\nabla\overline{p}+\frac{\partial J_\Omega}{\partial \boldsymbol{U}}\big]\frac{\partial \boldsymbol{U}}{\partial \alpha}d\Omega=0\end{aligned} \tag{19}$$

J_Γ and J_Ω stand, respectively, for the part of the cost function which is dependent on the flow state values on boundary and volume of the domain. Due to the definition of the cost function Eq. (13), its direct variation comes only from the interior domain. Moreover, it does not have any derivative of the pressure field. The corresponding terms are zeroed out in Eq. (19). The only derivative of the cost function is w.r.t the inflow and velocity in the interior of the domain, and at the locations where the measurements are available. From the latter we have

$$\frac{\partial J_\Omega}{\partial \boldsymbol{U}}=-2(\boldsymbol{U}_{M_i}-\boldsymbol{U}_{S_i})\quad i=1,2,\ldots \tag{20}$$

Using Eqs. (19 and 20) the adjoint equations can be derived as

$$-\nabla\overline{\boldsymbol{U}}\cdot\boldsymbol{U}-(\boldsymbol{U}\cdot\nabla)\overline{\boldsymbol{U}}=-\nabla\overline{p}+\nabla\cdot(2\nu_{eff}\mathbf{D}(\overline{\boldsymbol{U}}))+(\frac{2}{\omega_i})(\boldsymbol{U}_{M_i}-\boldsymbol{U}_{S_i})-\frac{1}{2}C_DA|\boldsymbol{U}|\overline{\boldsymbol{U}} \tag{21}$$

$$\nabla\cdot\overline{\boldsymbol{U}}=0 \tag{22}$$

where ω_i is the volume of the cell in which the measurement is located.

3.2.1 Boundary Conditions

The boundary integrals of Eq. (19) can be mathematically re-formulated and reduced to

$$\int_\Gamma \left[\overline{\boldsymbol{U}}\cdot\boldsymbol{n}\right]\frac{\partial p}{\partial\alpha}\,d\Gamma = 0 \tag{23}$$

$$\int_\Gamma \left[\boldsymbol{n}(\overline{\boldsymbol{U}}\cdot\boldsymbol{U}) + \nu_{eff}(\boldsymbol{n}\cdot\nabla)\overline{\boldsymbol{U}} - \overline{p}\boldsymbol{n}\right]\cdot\frac{\partial\boldsymbol{U}}{\partial\alpha}\,d\Gamma - \int_\Gamma \left[\nu_{eff}(\boldsymbol{n}\cdot\nabla)\frac{\partial\boldsymbol{U}}{\partial\alpha}\cdot\overline{\boldsymbol{U}}\right]d\Gamma = 0 \tag{24}$$

where $\boldsymbol{n}$ is the unit normal vector from the boundary faces. Except for the inlet, which is the design space, the adjoint BCs should be chosen such that the above equations are held.

Generally, for an ABL CFD domain no-slip wall (zero fixed velocity) and zero pressure gradient conditions are imposed on the ground. For a wall type of boundary in which $\frac{\partial\boldsymbol{U}}{\partial\alpha}$ is zero the first integral of Eq. (24) is cancelled. Then, the only way to satisfy the following conditions

$$\overline{\boldsymbol{U}}\cdot\boldsymbol{n} = 0 \tag{25}$$

$$(\boldsymbol{n}\cdot\nabla)\frac{\partial\boldsymbol{U}}{\partial\alpha}\cdot\overline{\boldsymbol{U}} = 0 \tag{26}$$

is to apply a no slip ($\overline{\boldsymbol{U}} = 0$) condition on the ground. No BC can be derived on the ground for the adjoint pressure but consistent with the primal a zero gradient condition is applied.

For the top and outlet boundaries of the domain a zero gradient velocity ($(\boldsymbol{n}\cdot\nabla)\frac{\partial\boldsymbol{U}}{\partial\alpha} = 0$) and zero fixed pressure ($p = 0$) are the common conditions for the primal system. These conditions fulfil Eq. (23) and cancel the second integral of Eq. (24). The only term that remains is the first term of Eq. (24) which needs to be zeroed out,

$$\left[\boldsymbol{n}(\overline{\boldsymbol{U}}\cdot\boldsymbol{U}) + \nu_{eff}(\boldsymbol{n}\cdot\nabla)\overline{\boldsymbol{U}} - \overline{p}\boldsymbol{n}\right]\cdot\frac{\partial\boldsymbol{U}}{\partial\alpha} = 0 \tag{27}$$

After decomposition into tangent and normal components it can be shown that the relations below should hold

$$\overline{p} = \overline{\boldsymbol{U}}\cdot\boldsymbol{U} + U_nU_n + \nu_{eff}(\boldsymbol{n}\cdot\nabla)U_n \tag{28}$$

$$0 = U_nU_t + \nu_{eff}(\boldsymbol{n}\cdot\nabla)U_t \tag{29}$$

where subscripts n and t represent the normal and in-plane components respectively. The adjoint BCs can be summarized as

$$\text{ground (wall):}\qquad \overline{\boldsymbol{U}} = 0 \qquad \boldsymbol{n}\cdot\nabla q = 0 \tag{30}$$

$$\text{top/outlet:}\qquad q = \overline{\boldsymbol{U}}\cdot\boldsymbol{U} + U_nU_n + \nu_{eff}(\boldsymbol{n}\cdot\nabla)U_n \qquad U_t = 0 \tag{31}$$

It is worth mentioning that the last term of the adjoint pressure which includes the kinematic viscosity, in implementation is often neglected (Nilsson et al., 2014). Moreover, the adjoint variables at the inlet should not be chosen to zero out the inlet velocity perturbations because the design variables are the inlet velocities. Instead, the zero gradient condition is imposed on the inlet for both adjoint velocity and adjoint pressure to have a well-posed system. Finally, from the integral over the boundary term in Eq. (19) it is clear one needs to evaluate the following expression

$$\frac{\partial J}{\partial \alpha} = \frac{\partial J}{\partial \boldsymbol{U}_{inlet}} = \boldsymbol{n}(\overline{\boldsymbol{U}}_{inlet} \cdot \boldsymbol{U}_{inlet}) + \overline{\boldsymbol{U}}_{inlet}(\boldsymbol{U}_{inlet} \cdot \boldsymbol{n}) + 2\nu_{eff}\boldsymbol{n} \cdot \mathbf{D}(\overline{\boldsymbol{U}}_{inlet}) \tag{32}$$

to compute the sensitivity.

3.2.2 Wind Direction Effect

As it was mentioned before, in ABL CFD simulations it is common to simulate first a 1D domain with a periodic boundary to obtain the inflow boundary condition. Then the cell center velocity of the 1D run is copied directly to its counterpart boundary face in the 3D domain. As a requirement, the number of cells in the 1D mesh and faces in the vertical direction of the 3D inflow boundary should be the same (see Figure 1). Moreover, and ideally, the face center heights in the 3D mesh are equal to their counterpart cell height in 1D. Although, in current work, a circular n inflow-outflow boundary is considered, with some small modification in the code the method can also be applied to other boundary shapes.

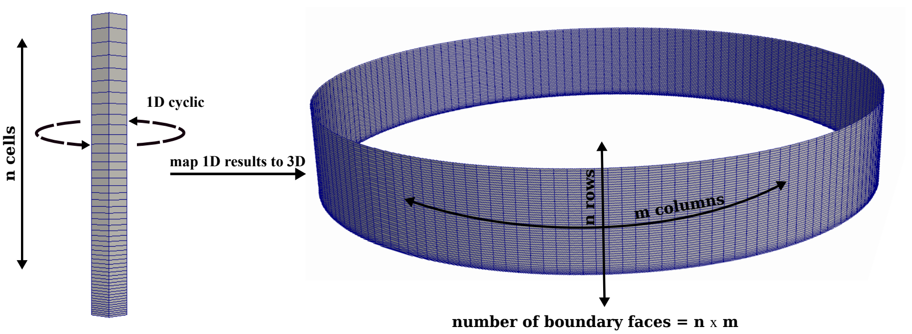

Figure 1. The inflow velocities of each cell from 1D precursor run (left) are copied to the boundary of the 3D domain (right) which has the similar number of cells in vertical direction. Ideally the height of each face in 3D domain boundary is exactly the same as its counterpart cell in the 1D mesh.

The inflow wind direction (WD) effect can be expressed by a rotation angle, θ, which rotates the inflow from its default west to east (WD=270°) direction,

$$\mathrm{WD} = 270^\circ - \theta \tag{33}$$

$$(\boldsymbol{U}_{\mathrm{inlet}})_x = \boldsymbol{U}_{\mathrm{1D}} \times \cos(\theta) \tag{34}$$

$$(\boldsymbol{U}_{\mathrm{inlet}})_y = \boldsymbol{U}_{\mathrm{1D}} \times \sin(\theta) \tag{35}$$

$$(\boldsymbol{U}_{\mathrm{inlet}})_z = 0 \tag{36}$$

The differentiation of Eqs. (34 and 35) gives

$$\frac{\partial(\boldsymbol{U}_{\mathrm{inlet}})_x}{\partial \boldsymbol{U}_{\mathrm{1D}}} = \cos(\theta) \quad ; \quad \frac{\partial(\boldsymbol{U}_{\mathrm{inlet}})_x}{\partial \theta} = -\boldsymbol{U}_{\mathrm{1D}}\sin(\theta) \tag{37}$$

$$\frac{\partial(\boldsymbol{U}_{\mathrm{inlet}})_y}{\partial \boldsymbol{U}_{\mathrm{1D}}} = \sin(\theta) \quad ; \quad \frac{\partial(\boldsymbol{U}_{\mathrm{inlet}})_y}{\partial \theta} = \boldsymbol{U}_{\mathrm{1D}}\cos(\theta) \tag{38}$$

The adjoint solver which was explained in the previous section computes the derivative of the cost function w.r.t. 3D inflow velocities at each face of the boundary:

$$\frac{\partial J}{\partial(\boldsymbol{U}_{\mathrm{inlet}})_x}|_{i,j} \quad ; \quad \frac{\partial J}{\partial(\boldsymbol{U}_{\mathrm{inlet}})_y}|_{i,j} \quad ; \quad i=1,\ldots,n \quad ; \quad j=1,\ldots,m \tag{39}$$

where i and j represent the row- and column-wise position of a face on the boundary and the total number of the faces in the 3D circular boundary is $N = n \times m$.

Using the chain rule, one can compute the sensitivity w.r.t each cell of the 1D inflow velocity as follows

$$\begin{aligned}\frac{\partial J}{\partial \boldsymbol{U}_{\mathrm{1D}}}|_i &= \Big(\sum_{j=1}^{m} \frac{\partial J}{\partial(\boldsymbol{U}_{\mathrm{inlet}})_x}|_{i,j}\Big)\frac{\partial(\boldsymbol{U}_{\mathrm{inlet}})_x}{\partial \boldsymbol{U}_{\mathrm{1D}}} + \Big(\sum_{j=1}^{m} \frac{\partial J}{\partial(\boldsymbol{U}_{\mathrm{inlet}})_y}|_{i,j}\Big)\frac{\partial(\boldsymbol{U}_{\mathrm{inlet}})_y}{\partial \boldsymbol{U}_{\mathrm{1D}}} \\ &= \Big(\sum_{j=1}^{m} \frac{\partial J}{\partial(\boldsymbol{U}_{\mathrm{inlet}})_x}|_{i,j}\Big)\cos(\theta) + \Big(\sum_{j=1}^{m} \frac{\partial J}{\partial(\boldsymbol{U}_{\mathrm{inlet}})_y}|_{i,j}\Big)\sin(\theta)\end{aligned} \tag{40}$$

The Eq. (40) means the x and y gradient components of the 3D inflow faces at the same column j are accumulated and then multiplied by $\cos(\theta)$ and $\sin(\theta)$ respectively before being summed. For the sake of clarity, Eq. (40) can be re-written as

$$\frac{\partial J}{\partial \boldsymbol{U}_{\mathrm{1D}}} = \frac{\partial J}{\partial(\boldsymbol{U}_{\mathrm{3D}})_x}\cos(\theta) + \frac{\partial J}{\partial(\boldsymbol{U}_{\mathrm{3D}})_y}\sin(\theta) \tag{41}$$

Using the same analogy, it can be shown that the sensitivity w.r.t rotation angle can be obtained by

$$\frac{\partial J}{\partial \theta} = -\Big[\frac{\partial J}{\partial(\boldsymbol{U}_{\mathrm{3D}})_x}\cdot \boldsymbol{U}_{\mathrm{1D}}\Big]\sin(\theta) + \Big[\frac{\partial J}{\partial(\boldsymbol{U}_{\mathrm{3D}})_y}\cdot \boldsymbol{U}_{\mathrm{1D}}\Big]\cos(\theta) \tag{42}$$

where the dot sign stands for the inner product of the two vectors.

4 Numerical Results

The adjoint solver and Eq. (32) are implemented based on the "*simpleFoam*" incompressible CFD solver of OpenFOAM-4.1. In this section, first the accuracy of the gradients obtained by the developed solver is tested against the second-order FD method in a simple 3D domain. Then, the inflow calibration of a real complex terrain is presented. For all the simulations in this work the general roughness length of the domain is $z_0 = 0.05$ [m] and the turbulent eddies are modelled by standard k-ε model with canopy model of Liu et al. (1996). Moreover, an ABL wall function is used to apply the roughness-related logarithmic law near to the ground, which is consistent with Monin-Obukhov similarity theory (Chang et al., 2018).

4.1 Gradient Verification

A cylindrical domain with 1000 [m] radius and 300 [m] height is chosen. The mesh of domain has 209K hexahedral cells in which the inflow/outflow boundary has 49 rows and 172 columns and a total number of 8428 faces ($N = 49 \times 172 = 8428$). The *topoSet* utility of OpenFOAM is used to select a number of cells as forest in a box size of 300 [m] in x and y and 40 [m] in z direction. The canopy drag and leaf area density for all forest cells are $C_d = 0.3$ and $A = 0.0033$ $[\mathrm{m}^{-1}]$. The operational Reynolds number based on the free-stream velocity, forest area height and air kinematic viscosity is $Re_h = 4.8 \times 10^5$.

For the primal CFD simulation of a circular domain the standard "*inletOutlet*" BC is used which checks if the flow is flowing into domain or out of it and switches between fixed value and zero gradient respectively. This BC is further developed to apply the derived adjoint BCs in a similar way and based on the flow direction on the boundary.

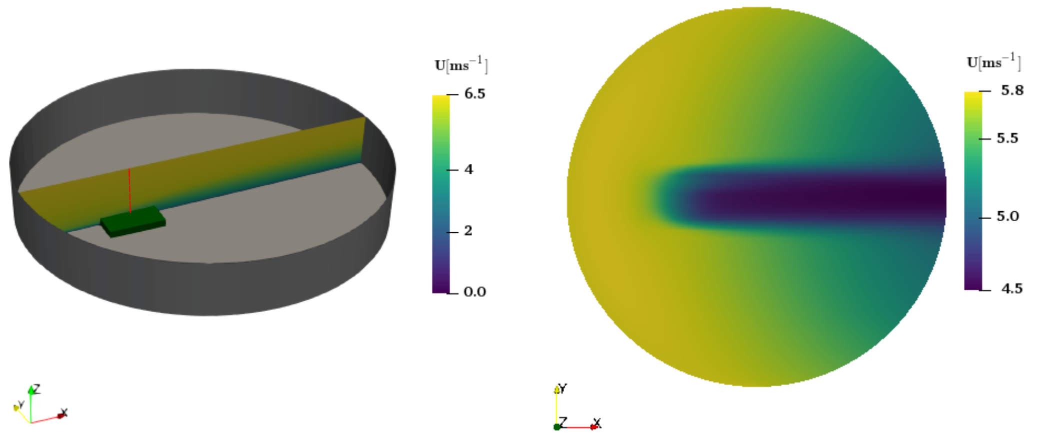

Figure 2. The CFD simulated velocity obtained from the reference simulation (WD=270°): The general view of the domain including the cubic forest area and the velocity field on a plane at the center of domain, $y = 0$, (left). The red line on the plane represents the location of desired target velocity profile (see figure 3). The velocity field on a plane at $z = 45$ [m] above the ground (right).

The gradient evaluation is carried out for a simulation in which the inflow wind direction is WD=240°. To have some reference wind speeds the terrain is simulated assuming the wind blows from west to east (WD=270°). The domain and the velocity field on the planes $y = 0$ [m] and $z = 45$ [m] are shown in Figure 2. The 1D inflow boundary and the target velocity profiles in the domain are plotted in Figure 3. The forest effect can be seen in the flow field and on the wind shear of the profile. Please note that the 30°difference between these two simulations is not the step-size for the finite difference computation.The finite difference step-size for gradient validation of wind direction is 0.3°.

The gradients obtained by the developed adjoint solver are plotted against the second order FD gradients in Figure 4. In general, the trends of the sensitivity profiles are similar. Moreover, the gradients are in excellent agreement except only for the heights between 50 [m] and 100 [m] in which the maximum relative error is $\varepsilon_{rel} = 0.10$. This difference can be traced back to grid resolution and the derivation of the adjoint equations and BCs which includes some simplifications such as the assumption of the frozen-turbulence.

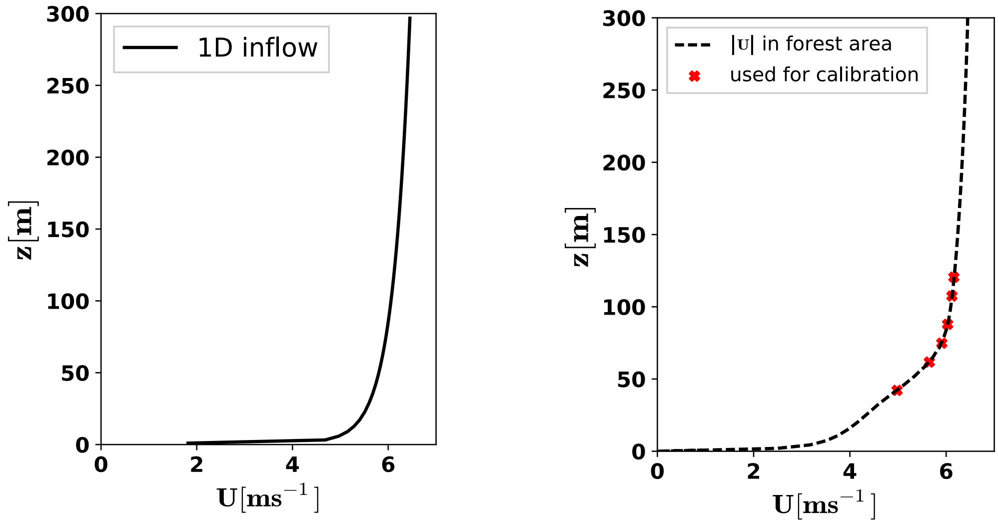

Figure 3. The 1D inflow boundary (left) and the velocity profile and its selected target speeds in the domain (right).

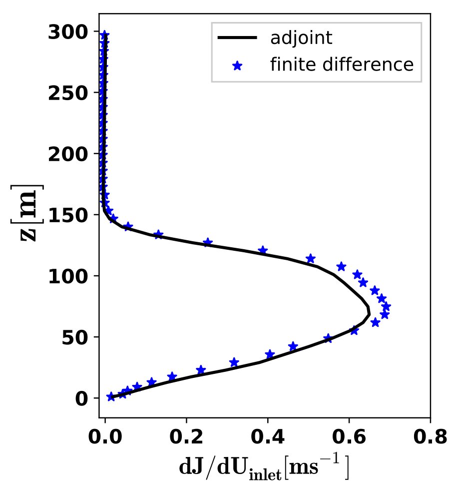

Figure 4. Comparison of the inflow boundary gradients by finite differences (FD) and via the adjoint approach.

The wind direction gradients are tabulated in Table 1. The derivative of calibration cost function w.r.t the change in wind direction is much higher than the change in the inflow boundary. Here also, there is a good agreement between the FD and the adjoint gradients and the relative error of wind direction sensitivity is close to that of inflow gradients. This is of course not surprising because it was shown in the previous section the sensitivity w.r.t. θ is obtained by mathematical operations only after the adjoint gradients are available. That said, the accuracy of the gradients computed by the adjoint solver and their signs show that they can be used for the purpose of the gradient-based calibration.

	adjoint	finite difference	ε_{rel}
$\frac{dJ}{d\theta}$	215	202	0.06

Table 1. Comparison of the wind direction gradients by finite differences (FD) and via the adjoint approach.

4.2 Inflow Calibration

For the cases, studied in this section, the in-house *terrainMesher* of the Fraunhofer IWES is used to generate the mesh. The primal flow field is simulated with an in-house CFD solver (Chang et al., 2018). The solver is a customized ABL-based version of the *simpleFoam* solver in the OpenFOAM package with a modified $k-\varepsilon$ turbulence model which behaves like a standard model for the neutral condition.

To optimize the inlet velocity profile, the primal and adjoint solvers are coupled with the DAKOTA optimization package Adams et al. (2017). The local gradient-based "CONMIN-frcg" solver of DAKOTA is used which is based on the conjugate-gradient algorithm of Fletcher-Reeves (Reeves and Fletcher, 1964; Hager and Zhang, 2006). Starting from an initial guess, the algorithm updates the design parameters, α^n, using the recurrence of Eq. (1) in which

$$(\Delta\alpha)^n = s^n \mathbb{D}^n \tag{43}$$

and the positive step size s^n is obtained by a line search, and the directions $\mathbb{D}$ are generated by the rule:

$$\mathbb{D}^{n+1} = -g^{n+1} + \beta^n \mathbb{D}^n \quad ; \quad g^n = \left[\nabla J(\alpha^n)\right]^T \quad ; \quad \beta^n = \frac{\|g^{n+1}\|^2}{\|g^n\|^2} \quad ; \quad \mathbb{D}^0 = -g^0 \tag{44}$$

Figure 5 shows the flow chart of the calibration, in which all the steps are followed sequentially. The optimizer starts with an initial guess (both inlet velocity and WD) and repeatedly asks either for the cost function value or the gradients. The primal and the adjoint solvers provide the required information, whenever it is needed, and this process continues until a certain convergence criterion is satisfied.

As a simplistic method, the inflow boundary of a RANS ABL domain can be represented by an empirical power-law function or an analytically obtained logarithmic function, which is based on the Monin-Obukhov similarity theory (MOST) (Foken, 2006). During optimization, the optimizer may ask for a cost function evaluation with a new inflow boundary, which is highly unrealistic for an ABL domain, leading to poor numerical stability or even divergence. One may assume that the inflow boundary is an analytical/empirical functions and, instead of the inflow velocities, calibrate the parameters of that function. Having the gradient via adjoint solver and using the chain rule, the gradient of the cost function with respect to these parameters can be easily obtained. However, the inflow boundary of a real 3D complex terrain is neither a power-law nor a logarithmic function and such parameterization may fail.

A simple approach is used in this study. This part is called the feasibility check in the optimization flow chart and is a Python script. First of all, the optimizer output is smoothed to avoid having any spikes in the inflow profile. Then it is checked whether a logarithmic, $f_1(x) = A\ ln(Bx+C)$, or a power-law, $f_2(x) = A(\frac{x}{B})^C$, function could be fitted into it. If either of these

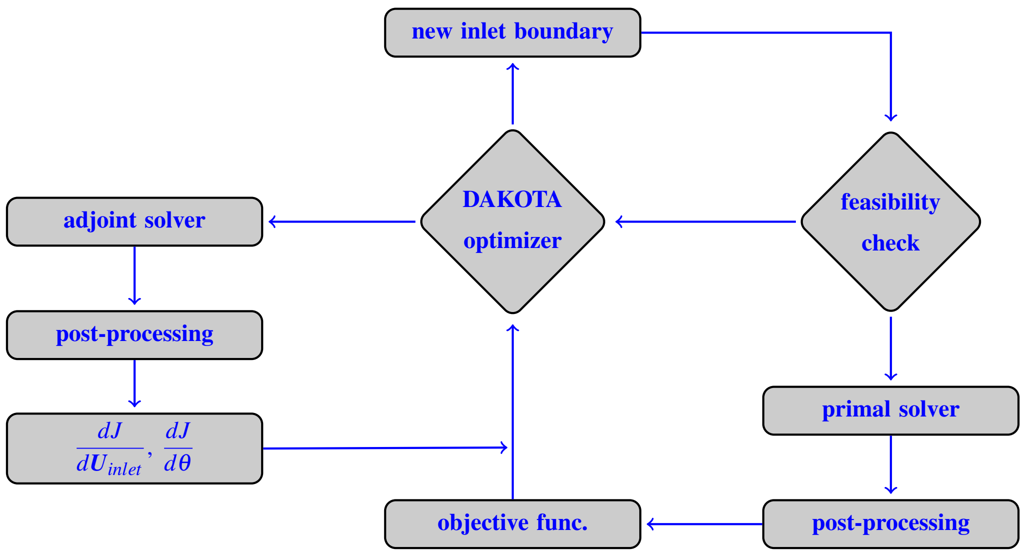

Figure 5. The calibration flowchart of the inflow boundary using DAKOTA optimization package and the developed adjoint solver.

functions is fitted and its coefficient of determination is above 0.96 the smoothed inflow from optimizer (not the fitted profile!), is accepted for the CFD solver. Otherwise, the optimization takes the last fitted profile and asks for a new gradient evaluation. In this way, the inflow boundary is not necessarily a logarithmic or power-law profile, and, moreover, it is not so unrealistic to be problematic for the solver. As an alternative, constraints or penalization term can be added to the objective function. This will be explored in future works when for instance the inflow turbulence properties are also considered as design parameters.

4.2.1 Ishihara Case

As a case study, an ABL domain with a 3D hill at the center is considered (see Figure 6). The hill has the shape $z = h\cos^2(\sqrt{x^2+y^2}/2L)$ with $h = 40$ m and $L = 100$ m. The scaled wind tunnel study of the case has been presented by Ishihara et al. (1999). The domain is meshed with two millions hexahedral elements. The roughness value of the domain is set to be $z_0 = 0.04$ m. The operational Reynolds number based on the free-stream velocity, hill height and air kinematic viscosity is $Re_h = 1.5 \times 10^4$. Using Ishihara et al. 1999 wind tunnel measurements, the x component of the velocity over the hill (U_x) is used for the inflow velocity calibration. The velocity flow field at the center of the domain and the wake behind the hill can be seen in Figure 6. Both the primal and the adjoint solvers have parallel scalability of OpenFOAM toolbox. The run-time of the adjoint solver is 60-70% of the primal flow simulation which has 30 minutes wall-time run with 24 CPU cores. It is worth noting that, in the adjoint solver there is no adjoint turbulence equation to be solved.

The stopping criterion is such that the absolute error between the measurements and the simulated velocities over the hill is $\varepsilon_{abs} < 0.1$ [ms^{-1}]. Figure 7 shows the history of optimization and the comparison of inlet velocity profiles. The optimization has converged with 14 primal and 12 adjoint calls. The optimal velocity profile is in good agreement with the experiment.

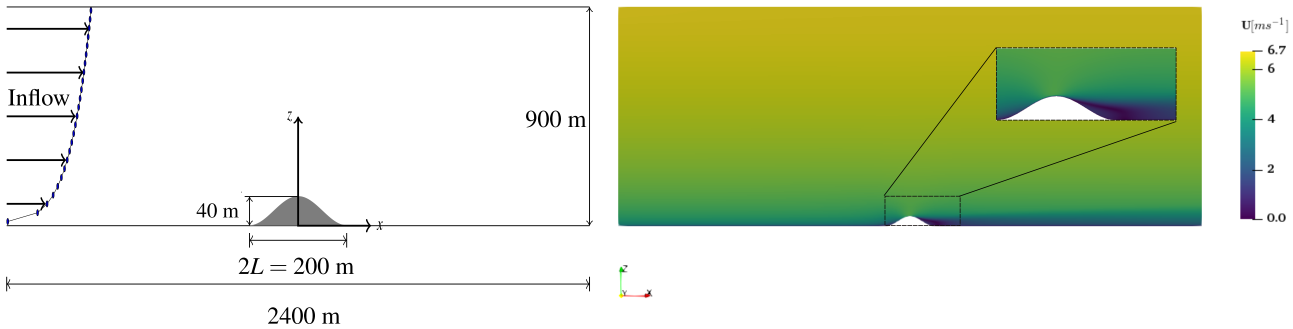

Figure 6. Main dimensions of the test case with 1000 m length in y direction. The velocity flow field on the plane at the center of domain, $y = 0$.

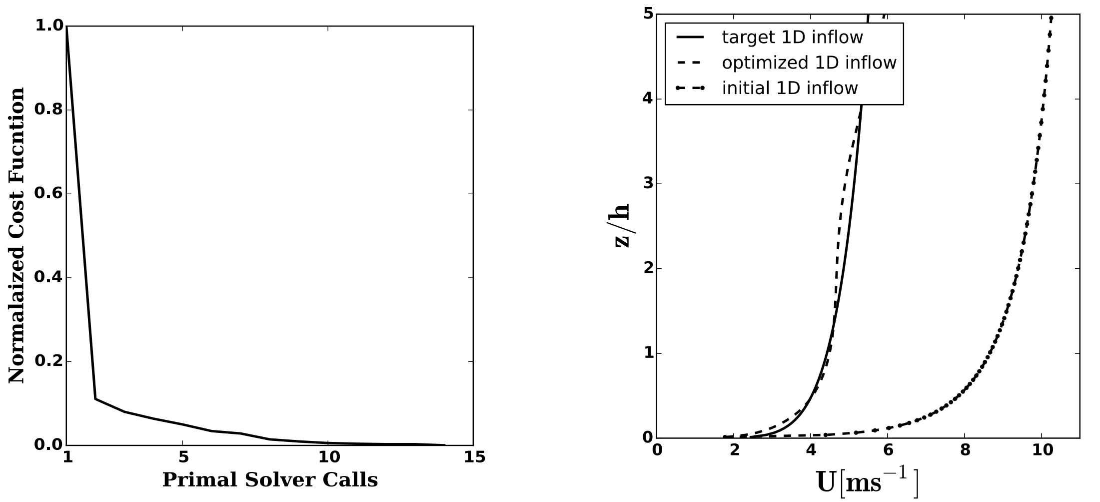

Figure 7. Optimization history (left) and inlet velocity profiles comparison (right).

There is a small deviation starting from the height $\frac{z}{h} > 2$. However, the comparison of the normalized velocity profiles over the hill, shown in Figure 8, confirms that the optimized inlet boundary is able to reproduce the experiment velocity profiles.

4.2.2 Kassel Case

To optimize the inlet boundary, the neutral condition of "Kassel Experiment" is considered. The domain, located near Kassel in Germany, is one of the cases of the New European Wind Atlas (NEWA) project (EU-ERA-NET, Accessed September 9, 2018). There are two meteorological masts at the site including a 200 [m] height mast (MM200). The wind rose of the site, provided by NEWA, indicates that most of the time the wind blows from south-west (SW) to north-east (NE).

The site is represented by a cylindrical domain with 15 km radius and 4 km height. The structured mesh is generated with the Fraunhofer IWES in-house software "*terrainMesher*" with 80 cells in the vertical direction. A mesh independence study

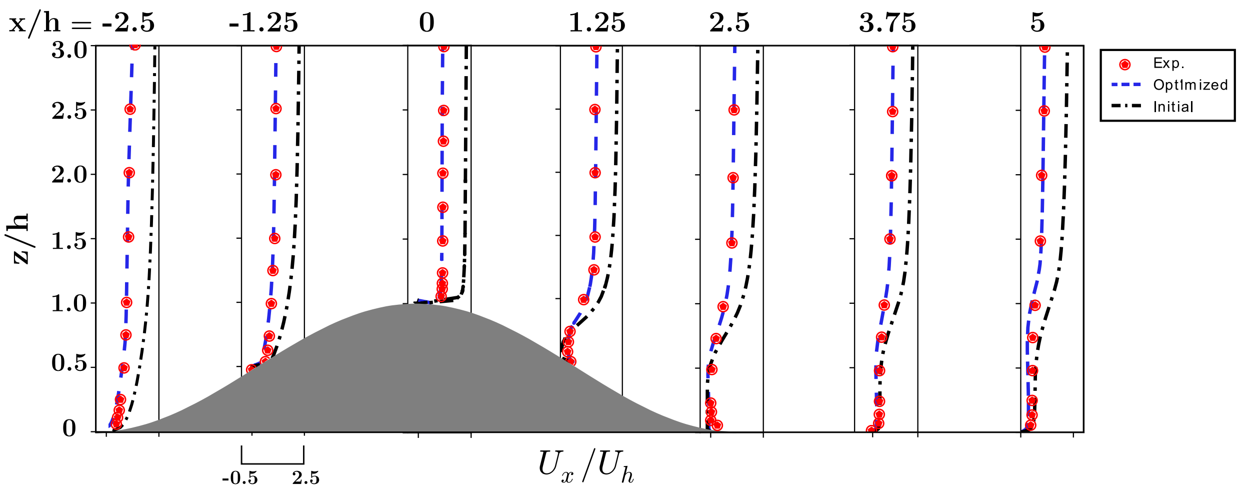

Figure 8. Normalized vertical profiles of longitudinal velocity component on the central plane of the hill. V_h is the velocity at the hill height in undisturbed region of the domain between inlet and hill.

is conducted to verify the suitability of a mesh of seven million hexahedral elements. The vicinity of the hill ($z_{hill} = 428$ $[m]$) where the MM200 mast is installed consists of forested area. The leaf area density of the site is obtained from the airborne LIDAR data and is provided by NEWA project (Dörenkämper, Accessed September 9, 2018).

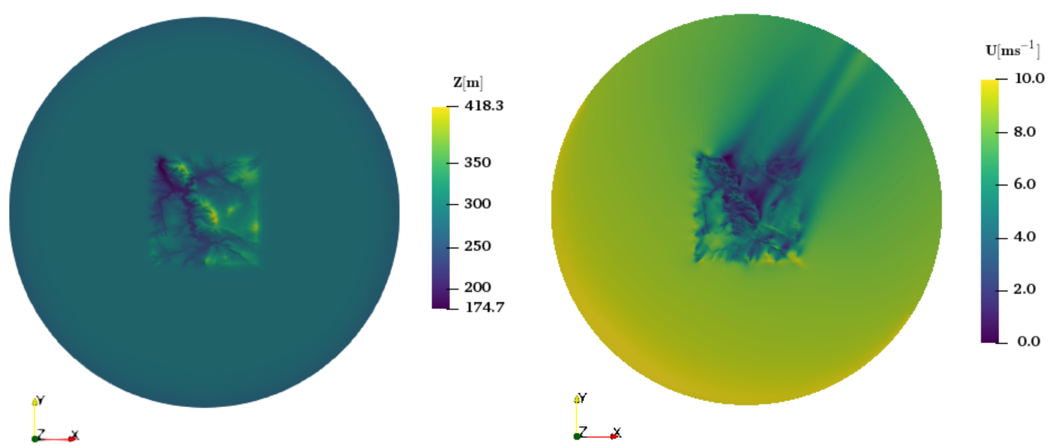

Figure 9. The cylindrical domain of the Kassel terrain (left) and the velocity field at the site at 40 [m] perpendicular distance from the terrain's surface points which is simulated with WD=213° (right). This simulation was used as the reference for the calibration.

Although the wind speed measurements of the MM200 mast from the site are available there is not enough information for wind direction at certain heights. This information is necessary for the adjoint solver in which the difference between the components of the measured and simulated velocities is a force term on the right-hand side of the adjoint momentum Eq. (21). Moreover, in the calibration of the solver with real measurements, it would become difficult to discuss the source of error when

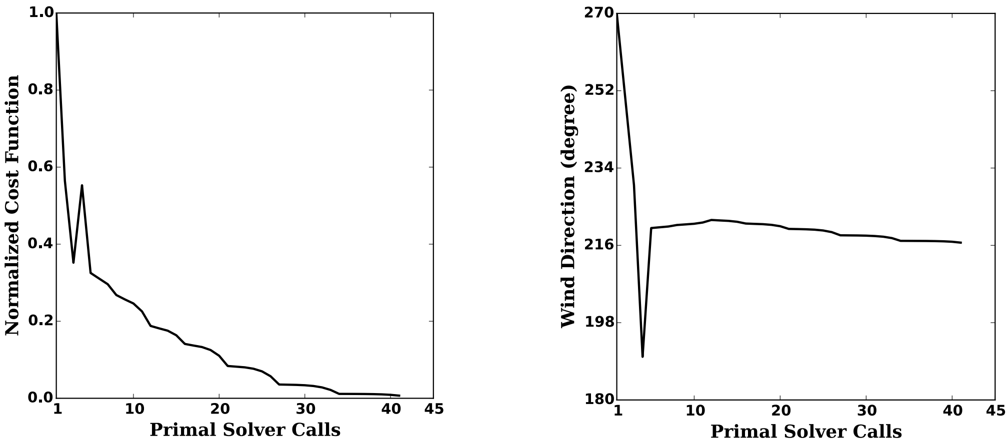

Figure 10. Inflow boundary calibration for Kassel domain; cost function convergence (left) and wind direction history (right).

there is discrepancy between the target and the calibrated profiles.This is because, aside from the inflow calibration process, which is the aim of the current study, many other parameters (e.g. turbulence model and the accuracy of forest and ground roughness map) are involved. Instead of using the real measurements of the mast, the velocity profile near to the mast from a reference simulation is considered. The selected wind speeds can be regarded as some pseudo measurements.

The initial guess WD is defined as 270° meaning wind blows from west to east. The turbulent properties of the inflow boundary are not part of the design parameters, but instead, in each new flow solver call of the optimization, the turbulent parameters of the k-ε model inside of the domain are initialized with the last converged solution. In this way, the turbulence model parameters are also gradually updated toward the end of the optimization when the inlet boundary velocities have reached their optimum value. The adjoint solver run-time for this case is also 60-70% of the primal solver wall-time which is 33 minutes with 120 CPU cores.

The convergence criterion is defined such that the optimization stops when the absolute difference between simulated and measured value, ε_{abs}, at all heights is below a certain value. Figure 10 shows the history of optimization. The optimization has called for a total number of 41 primal solver runs for cost function evaluation. In addition 8 adjoint gradient evaluations were required. The optimization convergence graph shows that there is a spike both in the cost function and the wind direction. This can be explained by the fact that in early iterations the derivative of cost function w.r.t the WD is much bigger than w.r.t. inflow. The optimizer updates the WD based on the first gradient computation. This continues for a few iterations until the cost function value, instead of decreasing, increases. Then the optimizer calls for a new gradient computation. From that point onward both the inlet velocities and the wind direction are gradually updated. The optimum WD is found as 216°.

The initial and optimal results are compared in Figure 11. The calibration is stopped based on a criterion, which is defined as $\varepsilon_{abs} < 0.2$ [ms^{-1}]. Although, there is a very small deviation between target and optimized inflow profiles the velocity profile near to MM200 mast is in good agreement with the pseudo measurements at all five selected heights and its error is well below the accepted threshold. Here a couple of points should also be noted. Firstly, a tighter error criterion would increase the

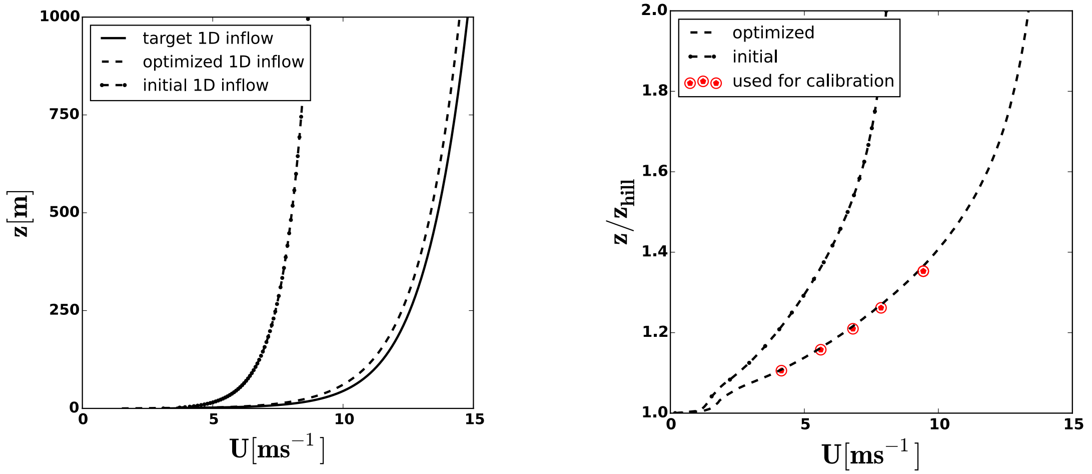

Figure 11. The target (obtained from a reference simulation), initial and optimized 1D inflow velocity (total inflow height is 4 km). The velocity profile over the hill from which the pseudo measurements are chosen (right).

calibration iterations and subsequently the number of CFD solver calls. Secondly, the sensitivity of wind speed at a certain point in the domain to a very small change in the inflow boundary is dependent on many parameters such as terrain complexity, wind direction, CFD model, etc., and cannot be easily generalized.

5 Conclusions

In this paper, it has been shown that the ABL CFD solvers can be calibrated via adjoint-based inlet boundary optimization. Based on the frozen-turbulence hypothesis, the adjoint equations and its boundary conditions for such problems are derived. The developed solver has been coupled with the DAKOTA optimization package and applied to the 3D Kassel terrain for a neutral stratification condition. Using the reference wind speeds and its direction over a hill within the terrain, the optimal inlet velocities have been found which are in good agreement with the inflow boundary profile of the reference simulation. The presented adjoint solver has the potential to be further developed by including Coriolis force, turbulence model and thermal stratification.

Acknowledgements. The work presented in this paper was part of the ETESIAN (FKZ 0324000D) project. The ETESIAN project was funded by the German Federal Ministry for Economic Affairs and Energy due to a decision of the German Bundestag. Moreover, the Kassel domain which was used for the numerical results is a test case from the NEWA project. The NEWA project was funded by the German Federal Ministry for Economic Affairs and Energy (ref. no. 0325832A/B) on the basis of a decision by the German Bundestag with further financial support from NEWA ERA-NET Plus, topic FP7-ENERGY.2013.10.1.2. The simulations were performed at the HPC cluster EDDY, located at the University of Oldenburg (Germany) and funded by the Federal Ministry for Economic Affairs and Energy (Bundesministerium für Wirtschaft und Energie) under grant number 0324005.

References

Adams, B. M., Ebeida, M. S., Eldred, M. S., Geraci, G., Jakeman, J. D., Maupin, K. A., Monschke, J. A., Swiler, L. P., Stephens, J. A., Vigil, D. M., and Wildey, T. M.: DAKOTA, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis, Tech. rep., Sandia Technical Report SAND2014-4633, 2017.

Bauweraerts, P. and Meyers, J.: Towards an adjoint based 4D-Var state estimation for turbulent flow, in: Journal of Physics: Conference Series, vol. 1037, p. 072055, IOP Publishing, 2018.

Chang, C.-Y., Schmidt, J., Dörenkämper, M., and Stoevesandt, B.: A consistent steady state CFD simulation method for stratified atmospheric boundary layer flows, Journal of Wind Engineering and Industrial Aerodynamics, 172, 55–67, 2018.

Chen, H., Miao, C., and Lv, X.: Estimation of open boundary conditions for an internal tidal model with adjoint method: a comparative study on optimization methods, Mathematical Problems in Engineering, 2013, 2013.

Davis, L.: Handbook of genetic algorithms, 1991.

Dhert, T., Ashuri, T., and Martins, J. R.: Aerodynamic shape optimization of wind turbine blades using a Reynolds-averaged Navier–Stokes model and an adjoint method, Wind Energy, 20, 909–926, 2017.

Dörenkämper, M.: NEWA-Rödeser Berg 2017-Blind Test, https://windbench.net/newa-r-deser-berg-2017-blind-test/, Accessed September 9, 2018.

EU-ERA-NET: The New European Wind Atlas (NEWA) project, http://www.neweuropeanwindatlas.eu/, Accessed September 9, 2018.

Foken, T.: 50 years of the Monin–Obukhov similarity theory, Boundary-Layer Meteorology, 119, 431–447, 2006.

Giles, M. B. and Pierce, N. A.: An introduction to the adjoint approach to design, Flow, turbulence and combustion, 65, 393–415, 2000.

Giles, M. B., Duta, M. C., Müller, J.-D., and Pierce, N. A.: Algorithm developments for discrete adjoint methods, AIAA journal, 41, 198–205, 2003.

Glover, N., Guillas, S., and Malki-Epshtein, L.: Statistical calibration of CFD modelling for street canyon flows, in: Proceedings of Building Simulation 2011: 12th Conference of International Building Performance Simulation Association, pp. 1513–1520, 2011.

Goit, J., Munters, W., and Meyers, J.: Optimal coordinated control of power extraction in LES of a wind farm with entrance effects, Energies, 9, 29, 2016.

Griewank, A. and Walther, A.: Evaluating derivatives: principles and techniques of algorithmic differentiation, SIAM, 2008.

Hager, W. W. and Zhang, H.: A survey of nonlinear conjugate gradient methods, Pacific journal of Optimization, 2, 35–58, 2006.

He, P., Mader, C. A., Martins, J. R., and Maki, K. J.: An aerodynamic design optimization framework using a discrete adjoint approach with OpenFOAM, Computers & Fluids, 168, 285–303, 2018.

Hinze, M., Pinnau, R., Ulbrich, M., and Ulbrich, S.: Optimization with PDE constraints, vol. 23, Springer Science & Business Media, 2008.

Ishihara, T., Hibi, K., and Oikawa, S.: A wind tunnel study of turbulent flow over a three-dimensional steep hill, Journal of Wind Engineering and Industrial Aerodynamics, 83, 95 – 107, 1999.

Jameson, A.: Aerodynamic Design via Control Theory, Journal of Scientific Computing, 3, 233–260, 1988.

Jameson, A., Martinelli, L., and Pierce, N.: Optimum Aerodynamic Design using the Navier-Stokes equations, Theor. Comp. Fluid. Dyn., 10, 213–237, 1998.

Kämmerer, S., Mayer, J., Paffrath, M., Wever, U., and Jung, A.: Three-dimensional optimization of turbomachinery bladings using sensitivity analysis, vol. 6 Part B, pp. 1093–1101, 2003.

King, R. N., Dykes, K., Graf, P., and Hamlington, P. E.: Optimization of wind plant layouts using an adjoint approach, Wind Energy Science, 2, 115–131, 2017.

Li, H.-D., He, L., Li, Y., and Wells, R.: Blading aerodynamics design optimization with mechanical and aeromechanical constraints, vol. 6 Part B, pp. 1319–1328, 2006.

Liu, J., Chen, J., Black, T., and Novak, M.: E-ε modelling of turbulent air flow downwind of a model forest edge, Boundary-Layer Meteorology, 77, 21–44, 1996.

Lopes da Costa, J.: Atmospheric Flow Over Forested and Non-Forested Complex Terrain, Ph.D. thesis, PhD thesis, University of Porto, 2007.

Mavriplis, D. J.: Discrete adjoint-based approach for optimization problems on three-dimensional unstructured meshes, AIAA journal, 45, 740, 2007.

Michalewicz, Z.: Evolution strategies and other methods, in: Genetic algorithms+ data structures= evolution programs, pp. 159–177, Springer, 1996.

Munters, W. and Meyers, J.: An optimal control framework for dynamic induction control of wind farms and their interaction with the atmospheric boundary layer, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 375, 20160 100, 2017.

Nadarajah, S. and Jameson, A.: A Comparison of the Continuous and Discrete Adjoint Approach to Automatic Aerodynamic Optimization, AIAA CP-00-0667, 2000.

Nadarajah, S. and Jameson, A.: Studies Of The Continuous And Discrete Adjoint Approaches To Viscous Automatic Aerodynamic Shape Optimization, 2001.

Nielsen, E. J., Diskin, B., and Yamaleev, N. K.: Discrete adjoint-based design optimization of unsteady turbulent flows on dynamic unstructured grids, AIAA journal, 48, 1195, 2010.

Nilsson, U., Lindblad, D., and Petit, O.: Description of adjointShapeOptimizationFoam and how to implement new objective functions, course at Chalmers University of Technology, Enero, 2014.

Othmer, C.: A continuous adjoint formulation for the computation of topological and surface sensitivities of ducted flows, International Journal for Numerical Methods in Fluids, 58, 861–877, 2008.

Othmer, C. and Grahs, T.: Approaches to fluid dynamic optimization in the car development process, in: Eurogen, edited by et. al., R. S., Munich, 2005.

Othmer, C., Kaminski, T., and Giering, R.: Computation of topological sensitivities in fluid dynamics: cost function versatility, in: Eccomas CFD 2006, P. Wesseling and E. Oñate and J. Périaux, 2006.

Pironneau, O.: On Optimum Design in Fluid Mechanics, Journal of Fluid Mechanics, 64, 97–110, 1974.

Rebollo, T. C. and Lewandowski, R.: Mathematical and numerical foundations of turbulence models and applications, Springer, 2014.

Reeves, C. M. and Fletcher, R.: Function minimization by conjugate gradients, The Computer Journal, 7, 149–154, 1964.

Ruder, S.: An overview of gradient descent optimization algorithms, arXiv preprint arXiv:1609.04747, 2016.

Schneiderbauer, S. and Pirker, S.: Determination of open boundary conditions for computational fluid dynamics (CFD) from interior observations, Applied Mathematical Modelling, 35, 763–780, 2011.

Seiler, U.: Estimation of open boundary conditions with the adjoint method, Journal of Geophysical Research: Oceans, 98, 22 855–22 870, 1993.

Stevens, R. J. and Meneveau, C.: Flow structure and turbulence in wind farms, Annual review of fluid mechanics, 49, 311–339, 2017.

Towara, M. and Naumann, U.: A Discrete Adjoint Model for OpenFOAM, Procedia Computer Science, 18, 429 – 438, https://doi.org/http://dx.doi.org/10.1016/j.procs.2013.05.206, http://www.sciencedirect.com/science/article/pii/S1877050913003499, 2013 International Conference on Computational Science, 2013.

Towara, M., Schanen, M., and Naumann, U.: Mpi-parallel discrete adjoint openfoam, Procedia Computer Science, 51, 19–28, 2015.

Vali, M., Petrović, V., Boersma, S., van Wingerden, J.-W., Pao, L. Y., and Kühn, M.: Adjoint-based model predictive control for optimal energy extraction in waked wind farms, Control Engineering Practice, 84, 48–62, 2019.

Wu, X.: Inflow turbulence generation methods, Annual Review of Fluid Mechanics, 49, 23–49, 2017.