Code-to-code-to-experiment validation of LES-ALM wind farm simulators

Code-to-code-to-experiment validation of LES-ALM wind farm simulators

C Wang[1], A Muñoz-Simón[2], G Deskos[3], S Laizet[2], R Palacios[2], F Campagnolo[1], C L Bottasso[1]

[1] Wind Energy Institute, Technische Universität München, Boltzmannstraße 15, D-85748 Garching bei München, Germany
[2] Department of Aeronautics, Imperial College London, SW7 2AZ, London, UK
[3] National Renewable Energy Laboratory, CO 80401, Golden, USA

E-mail: carlo.bottasso@tum.de

Abstract.
The aim of this work is to present a detailed code-to-code comparison of two Large-Eddy Simulation (LES) solvers. Corresponding experimental measurements are used as a reference to validate the quality of the CFD simulations. The comparison highlights the effects of solver order on the solutions, and it tries to answer the question of whether a high order solver is necessary to capture the main characteristics of a wind farm. Both solvers were used on different grids to study their convergence behavior. While both solvers show a good match with experimental measurements, it appears that the low order solver is more accurate and substantially cheaper in terms of computational cost.

1. Introduction

Large-Eddy Simulation (LES) is actively researched and used because it has the potential to explain wind farm phenomena and to predict effects that enable improved design and operation [1]. However, notwithstanding the success of LES, there is still only a limited understanding of the actual accuracy of such numerical methods in representing the complex physical processes that govern wake-turbine interactions. The present paper presents a code-to-code comparison of two state-of-the-art LES-ALM (Actuator Line Method) codes with significantly different features. Numerical results of both solvers for different mesh resolutions are compared in this work. Corresponding experiments are used to validate the numerical results. The objective is to evaluate the accuracy and uncertainty of LES for wind farm flows and possibly obtain indications for the future evolution of the numerical procedures. While the experiments provide benchmarks of integral quantities like power, thrust and mean wake profile, the code-to-code comparison can also illustrate differences of other quantities like Reynolds shear stresses.

2. Methodology

2.1. Experimental setup

Three laterally-misaligned scaled wind turbines of the G1 type [2], with a rotor diameter of 1.1 m, were operated in the boundary layer wind tunnel of the Politecnico di Milano, which

has a cross-section of 3.8 × 13.8 m^2 and a length of 36 m. The longitudinal distance between each wind turbine was 4D, while the lateral distance was 0.5D, resulting in a partial wake impingement on the two downstream turbines. An inflow condition with 5% turbulence intensity was generated passively with spires positioned close to the inlet of the wind tunnel. All wind turbines were operated in region II with constant pitch and were governed by a standard look-up-table generator torque controller. First, a greedy case was considered, in which the yaw angles of all turbines were set to zero. Then, a yaw misalignment case was tested, in which the yaw angles of the three turbines were set to optimal values with respect to total wind farm power output. In all considered conditions, the power and thrust of each turbine were measured. Additionally, two scanning LiDARs measured the horizontal mean wake profile of all turbines on a horizontal slice 0.1D above hub height [3]. A previous code-to-experiment validation with the same measurement data has been reported in [4].

2.2. Numerical models

The TUM solver [1] is based on OpenFoam [6] and SOWFA [7], and it is described in [1]. The ICL Winc3D solver is described in [5]. Table 1 summarizes the differences between the two solvers. LES with constant Smagorinsky model and the same Smagorinsky constant of 0.16 has been used for both codes.

Item	TUM	ICL
Spatial discretization	Finite volume	Compact finite difference scheme
Time marching scheme	Backward Euler scheme	Third order Runge-Kutta scheme
Pressure-velocity decoupling	Rhie-Chow interpolation	Half-staggered grid
Continuity enforcement	Pressure correction	Pressure correction
Turbine model	ALM, FAST	In-house ALM code
Mesh	Cartesian mesh with refinement	Uniform Cartesian mesh
Spatial convergence order	Second order	Sixth order
Turbulence model	Constant Smagorinsky	Constant Smagorinsky
LES filter	Implicit	Explicit

Table 1: Comparison of the main characteristics of the two codes.

One of the critical differences between the two codes is the different temporal and spatial discretizations. The backward Euler time marching scheme of the TUM code has a lower order compared to the Runge-Kutta scheme used by ICL. The second-order finite volume spatial discretization method adopted by the TUM code also has a lower order compared to the sixth order compact finite difference scheme used by ICL [5]. The ICL code also has better scalability than OpenFOAM. In contrast, the finite volume method of the TUM code allows for mesh refinement, while the ICL code is limited to a uniform Cartesian mesh because of its spectral formulation. The TUM code used fine meshes only at necessary locations and a coarse mesh for other regions of the domain, while the ICL code used the same fine mesh for the whole computational domain. This results in significantly different cell numbers and computational costs. To avoid staggered solutions, the TUM code adopts the Rhie-Chow interpolation method [8], while the ICL code uses a half-staggered grid. The ALM approach has been used by both codes to model the rotor.

2.3. Simulation setup

In contrast to the quite different numerical schemes, the inputs to both codes are exactly the same, including all the turbine setup, the inflow data, initial conditions, boundary conditions, smearing length scales ϵ and time steps for each mesh resolution. To guarantee the same rotor operating condition, a prescribed constant rotor speed and blade pitch angle were used. The values of rotor speed are identical to the mean values measured in the experiments. In reality, the rotor speed oscillates in response to the turbulent fluctuations in the experiments. However, specifying an average rotor speed is sufficiently accurate to capture mean quantities, as shown in previous work [4]. Furthermore, to avoid subtle differences that could be caused by the models of the nacelle, tower and tip and hub loss models, all these features were switched off. The mesh resolutions were the same upstream, downstream and around all rotors.

3. Results

3.1. Impact of mesh refinement

The impact of mesh refinement on the inflow should be quantified since turbine power is very sensitive to inflow conditions. Figure 1 shows the inflow profile and turbulence intensity along a horizontal line slightly above hub-height and 2.55D in front of the first rotor. This location is about 1.4 D behind the inlet plane of the domain. The vertical dashed lines in both plots of Fig. 1 indicate the bounds of mesh refinement used for the TUM code. Within the bounds, the grid size is 2 cm (55 cells per rotor diameter); outside the bounds, the grid size is 4 cm. This results in a total number of cells of 10.3 million, which is significantly less than the 111 million cells used by the ICL code for a globally uniform Cartesian mesh. The values of mean velocity and turbulence intensity show little difference within the refinement bounds, while the turbulence intensity predicted by the TUM code is slightly lower than that of the ICL code because of the coarse mesh outside of the bounds. A drop of turbulence intensity due to a coarser mesh is an expected effect, while the good match in the fine mesh region indicates little adverse impact of mesh refinement on the results.

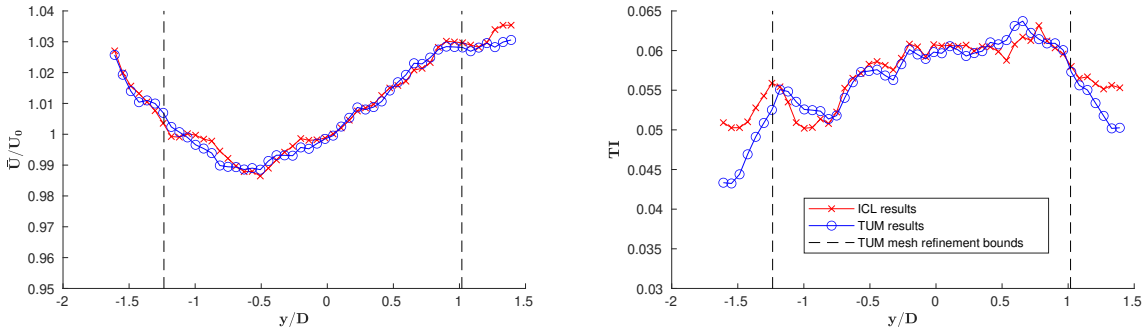

Figure 1: Lateral inflow profile and turbulence intensity along a horizontal line 2.55D in front of the first rotor.

3.2. Comparison and convergence of mean quantities

Figure 2 shows the mean velocity fields slightly above hub-height measured in the experiments and simulated by the codes with a 2 cm grid resolution. Table 2 shows the mean power and thrust and the relative errors of both codes in percentage with respect to the experiments for different grid resolutions.

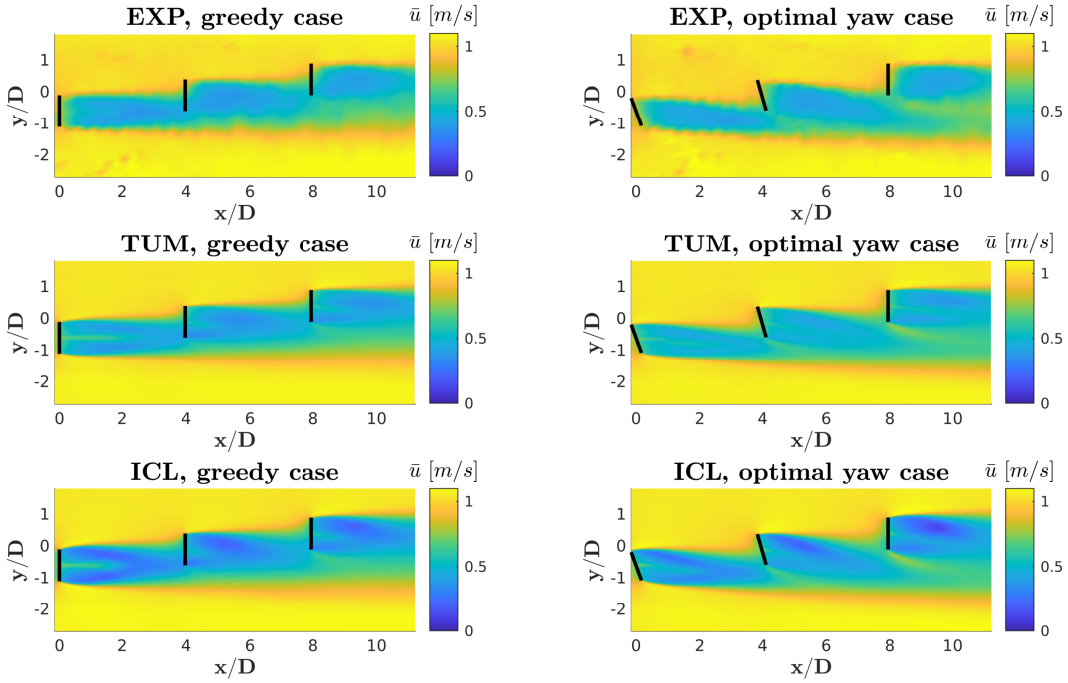

Figure 2: Top view of the mean velocity fields on a plane 0.1D above hub height. The black lines represent the rotor planes. The grid resolution is 2 cm.

Mesh (cm)		4		3	2		1.5	-
Time step (ms)		1.000		0.750	0.500		0.375	
ϵ (cm)		6		5	4		3	
Case		TUM	ICL	TUM	TUM	ICL	TUM	EXP
$\gamma_1 = 20°$ $\gamma_2 = 16°$ $\gamma_3 = 0°$	$\bar{P}_1$	+1.2%	0.0%	+0.5%	+0.7%	-0.7%	-1.7%	40.4 W
	$\bar{P}_2$	-25.5%	-20.6%	-16.5%	-6.6%	-18.4%	-8.2%	36.4 W
	$\bar{P}_3$	-21.9%	-16.6%	-15.9%	-4.6%	-17.0%	-7.1%	45.2 W
	$\bar{T}_1$	-7.2%	-6.6%	-7.2%	-7.2%	-6.6%	-8.4%	16.6 N
	$\bar{T}_2$	-21.9%	-18.7%	-17.4%	-13.5%	-18.7%	-14.2 %	15.5 N
	$\bar{T}_3$	-8.8%	-5.7%	-5.0%	+0.6%	-5.7%	-0.6%	15.9 N
$\gamma_1 = 0°$ $\gamma_2 = 0°$ $\gamma_3 = 0°$	$\bar{P}_1$	+1.1%	-1.4%	+1.1%	0.5%	-2.3%	-2.0%	44.3 W
	$\bar{P}_2$	-27.9%	-23.4%	-21.5%	-10.9%	-24.9%	-12.1%	26.5 W
	$\bar{P}_3$	-16.5%	-12.2%	-10.2%	-1.0%	-11.2%	-3.6%	30.3 W
	$\bar{T}_1$	-4.5%	-4.0%	-4.5%	-4.5%	-4.5%	-5.1%	17.6 N
	$\bar{T}_2$	-23.5%	-22.1%	-20.6%	-16.4%	-23.0%	-17.2 %	12.2 N
	$\bar{T}_3$	-11.8%	-11.0%	-9.4%	-6.3%	-10.2%	-7.1 %	12.7 N

Table 2: Grid convergence and comparisons among the two codes and measurements. Percentage errors are computed with respect to the experimental results.

Figure 2 shows that the expansion, deflection and recovery of the wakes are captured by both codes, although the ICL one somewhat underestimates wake speed. While the TUM code was used on four different meshes, the ICL code was run only on two of them. The time step is strictly proportional to the mesh size, giving almost the same CFL number for all simulations. The basic uniform Gaussian body force projection method [10] was used for both solvers. The projection parameter ϵ was chosen to match the power of the first turbine. As ϵ does not exactly scale with grid size Δ, the ratio ϵ/Δ is between 1.5 and 2 for all cases.

As the grid is refined, the errors of the TUM code drop significantly. Differences between the results on the 2 cm and 1.5 cm grids are small enough to indicate a final convergence of the solution. In contrast, the ICL code, thanks to its higher order discretization, converges much earlier, since the differences between results on the 4 cm and 2 cm grids are already very small.

For a grid resolution of 2 cm, both the TUM and ICL codes predict the power and thrust of the first turbine with almost the same accuracy for both the greedy and the optimal yaw cases. However, the TUM code moderately underestimates the power and thrust of the two downstream turbines, while the ICL code significantly underestimates these quantities. The larger error of the ICL code is expected, since Fig. 2 already shows that the wakes computed by this solver are too slow. Note that all simulations were conducted without nacelle and tower models. On the one hand, the wakes shed by the tower and nacelle decelerate the flow in the near wake region, but on the other hand, the stronger turbulence intensity caused by the wakes of nacelle and tower enhances wake recovery. These two phenomena have opposite effects on the wake. As indicated in [11], ambient turbulence intensity also contributes to making one prevail over the other. For the 5% turbulence intensity considered here, the two effects almost cancel each other. Therefore, the errors of the 2 cm grid resolution case are close to the errors reported in [4], in which CFD-ALM simulations were conducted with nacelle and tower models for the same scenario as in this paper.

3.3. Comparison of wake and Reynolds shear stresses

Figure 3 shows the mean velocity and two components of the Reynolds shear stress $\overline{u'u'}$ and $\overline{u'v'}$ for the optimal yaw case, while Fig. 4 shows the same quantities for the greedy case. Data has been sampled on nine horizontal crossflow lines 0.1D above hub-height. Each column of the figure corresponds to different longitudinal downstream distances as in Fig.2. The positions of the three turbines are 0D, 4D, and 8D, so there are three sample lines 1D, 2D and 3D behind each turbine. The quantity $\overline{u'u'}$ is the square of turbulence intensity along the longitudinal direction, and $\overline{u'v'}$ reflects the lateral turbulent flux that drives wake recovery.

A high-velocity region is visible behind the center of each rotor, since there is no nacelle model. Except for this difference, the wakes simulated by the TUM code have a good match with the experiments from 1D to 10D. However, the wakes at 11D are significantly slower than the measured ones, for both the optimal yaw and the greedy cases. Consistently with the observation from Fig. 2, the near wakes simulated by the ICL code are slower than the experiments. The situation improves for the far wakes (3D, 7D), although the flow speed is still slower than for the TUM code and the experiments, which explains the larger power underestimation shown in Table 2. The Reynolds shear stress $\overline{u'u'}$ simulated by the ICL code is significantly higher than for the TUM code, which indicates a stronger turbulence intensity. Similarly, $\overline{u'v'}$ computed by the ICL code is moderately higher than for the TUM code in absolute value terms. Both components indicate a stronger wake recovery for the ICL code, which is consistent with the observed wake characteristics. Although experiments and simulations are both subject to errors, the shapes of the curves shown in Fig. 4 and Fig. 3 are very similar and the differences of various physical quantities are within an acceptable range. The wake recovery is simulated better by the TUM code probably because of the Smagorinsky constant, since its value has been calibrated for second order schemes.

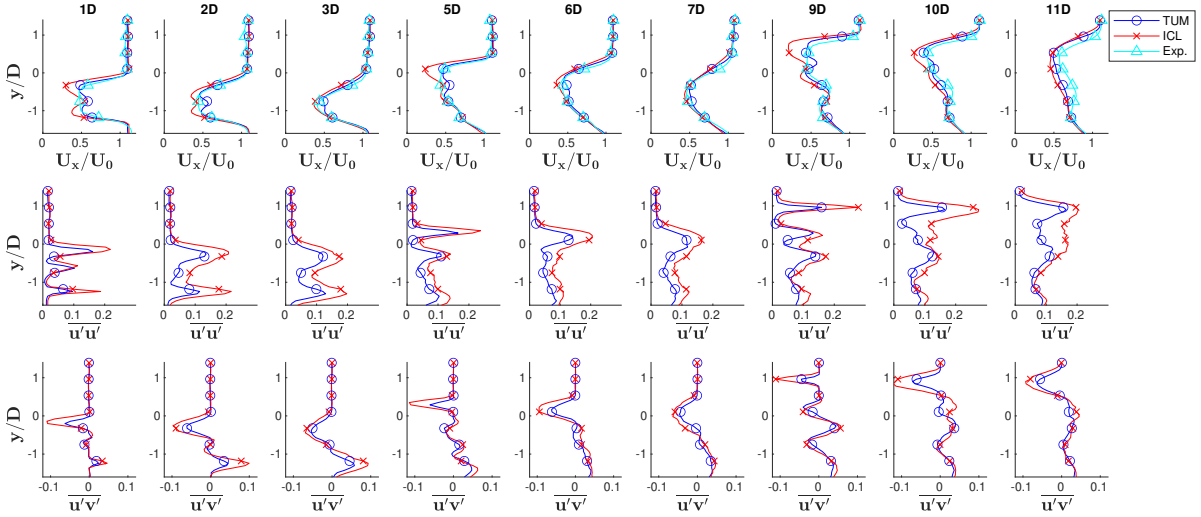

Figure 3: Wake profiles slightly above hub-height at three downstream stations behind each turbine for the optimal yaw case.

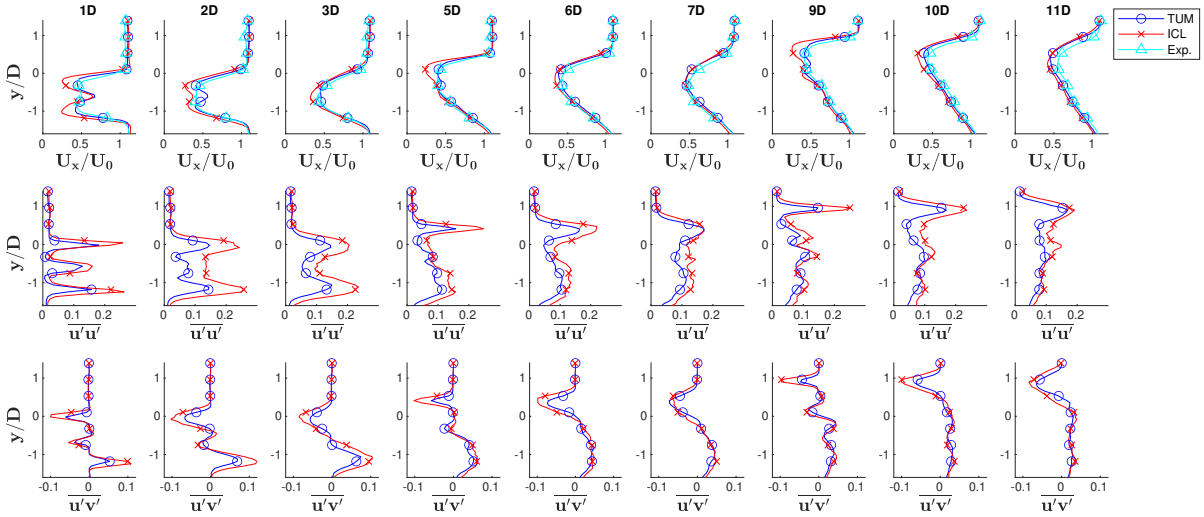

Figure 4: Wake profiles slightly above hub-height at three downstream stations behind each turbine for the greedy case.

Useful information can be extracted from the differences. First, the excessively slow near wake of the ICL code indicates that the axial induction factor is overestimated; in comparison, the TUM code seems to better resolve this quantity. Second, the higher-order scheme of the ICL code predicts stronger Reynolds shear stresses because it is less dissipative. However, it appears that wake recovery in this case is significantly quicker than in the experiment, which partially offsets the overestimated axial induction factor. In both simulation cases, the wake recovery of the first and second turbines are closer to the experiments for the TUM code, while the wake of the third turbine recovers too slowly. This indicates that the TUM second order scheme is slightly over-dissipative and the ICL sixth order scheme is moderately under-dissipative.

3.4. Turbulence spectra and autocorrelation

The turbulence spectrum and autocorrelation of the longitudinal velocity component were computed at a number of points in the wakes of the turbines. The match of these two quantities between the two codes is very good for all considered points. As a representative example, Fig. 5 shows the spectra and autocorrelations at a point 1D downstream of the center of the second rotor for the greedy case. The corresponding simulations have a grid resolution of 2 cm. A linear decay slope of -5/3 can be seen in both spectra in the inertial range. The TUM code predicts lower energy compared to the ICL code for most frequencies, which is consistent with the lower $\overline{u'u'}$ shown in Fig. 4. The autocorrelation curves of TUM and ICL have a good match for both mesh resolutions, thus giving a very similar integral time scale.

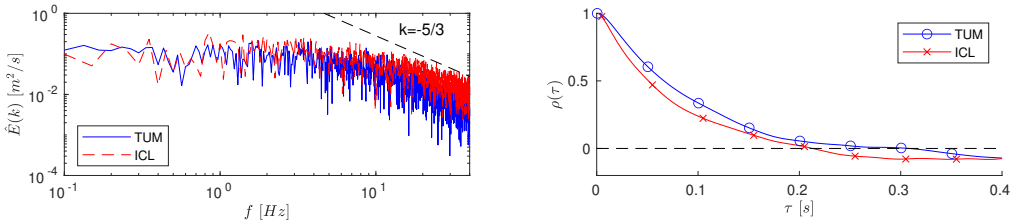

Figure 5: Turbulence spectra and autocorrelation of the longitudinal velocity component.

3.5. Computational efficiency

Table 3 shows a comparison between computation efficiency between the two codes for a grid size of 2 cm. For both codes, the number of cells N_c, the time consumption per time step Δt, the number of processors used N_p and the High Performance Conjugate Gradients (HPCG) benchmark [12] are listed. The last column of the table, $N_p N_s \Delta t / N_c$, is an estimation of the number of floating-point operations per time step per cell, where N_s is the number of floating point operations of the processor per second. This value is slightly higher for the ICL code, which makes intuitive sense because there is a computational cost to implement a high order scheme.

Code	Number of cells N_c	Time per time step Δt (s)	Number of processors N_p	Cluster HPCG N_s (GFlop/s)	$\frac{N_p N_s \Delta t}{N_c}$ (Flop)
TUM	10.3×10^6	1.789	192	0.680	2.27×10^4
ICL	111×10^6	2.085	2208	0.684	2.84×10^4

Table 3: Comparison of the computational efficiency of the TUM and ICL code

4. Conclusions

A code-to-code-to-experiment comparison has been conducted for the two LES-ALM codes and wind tunnel experiments of three wake-interacting wind turbines. The main difference between the two codes is the discretization scheme (with its implications on the grid), which is of the second and sixth order, respectively. Both codes are able to simulate this small wind farm with reasonable accuracy. A grid convergence study shows that the higher order scheme converges at a coarser mesh resolution, as expected. The comparison of power, thrust, mean wake profiles, Reynolds shear stresses, flow spectra and velocity autocorrelations show a high level of similarity

between the two codes. The sixth order scheme is moderately under-dissipative, while the second order scheme is slightly over-dissipative. The second order scheme has a better match with the experiments and also has a much lower computational cost. This indicates that the second order CFD is a good choice for wind energy applications. The reasons for the better matching of the lower order scheme might be due to the calibration of the Smagorinsky constant, which was set to 0.16 for both codes in this work. It is possible that this constant should be recalibrated for the high order spectral method. Future work will try to clarify whether this is indeed the case, or whether other reasons can explain the better results of the lower order scheme.

5. Acknowledgements

This work is supported by the TUM Global Incentive Fund 2019 and Imperial-Technische Universität München Collaboration Fund. This work has also been partially supported at TUM by the CL-WINDCON project, which receives funding from the European Union Horizon 2020 research and innovation program under grant agreement No. 727477. This research has received funding at ICL from the European Unions Horizon 2020 research and innovation program under the Marie Sklodowska-Curie grant agreement No. 765579 (CONFLEX project). The TUM authors express their appreciation to the Leibniz Supercomputing Centre (LRZ) for providing access and computing time on the SuperMUC Petascale System. The authors would like to thank the UK turbulence consortium (grant number EP/R029321/1) for the computational time on the UK Supercomputing facility ARCHER.

References

[1] Wang J et al. 2019 Wake behavior and control: comparison of LES simulations and wind tunnel measurements *Wind Energy Sci.* **4**(1) 71-88
[2] Campagnolo F et al. 2016 Wind tunnel testing of a closed-loop wake deflection controller for wind farm power maximization *J. Phys. Conf. Seri.* **753**(3) 032006
[3] Dooren M F et al. 2017 Demonstration and uncertainty analysis of synchronised scanning LiDAR measurements of 2-D velocity fields in a boundary-layer wind tunnel Wind Energy Sci. **2** 329
[4] Wang C et al. 2018 Validation of large-eddy simulation of scaled waked wind turbines in different yaw misalignment conditions *J. Phys.: Conf. Series.* **1037**(6) 062007
[5] Deskos G, Laizet S and Palacios R 2020 WInc3D: A novel framework for turbulence-resolving simulations of wind farm wake interactions *J. Wind Energy* **23**(3) 779-794
[6] Jasak H 2009 OpenFOAM: open source CFD in research and industry *Int. J. Nav. Arch. Ocean* **1**(2) 89-94
[7] Fleming P et al. 2014 Evaluating techniques for redirecting turbine wakes using SOWFA *Renewable Energy* **70** 211-218
[8] Rhie C M, Chow W L 1983 Numerical study of the turbulent flow past an airfoil with trailing edge separation *AIAA J.* **21**(11) 1525-1532
[9] Guntur S et al. 2016 FAST v8 verification and validation for a MW-scale Wind turbine with aeroelastically tailored blades Wind Energy Symp. (San Diego, USA) 1008
[10] Troldborg N, Sørensen J N and Mikkelsen R 2007 Actuator line simulation of wake of wind turbine operating in turbulent inflow *J. Phys.: Conf. Ser.* **75**(1) 012063
[11] Wang J et al. 2018 High-fidelity Simulation for Wind Turbine Wake Interaction Research. Technische Universität München
[12] Dongarra J, Michael A H and Luszczek P 2016 High-performance conjugate-gradient benchmark: A new metric for ranking high-performance computing systems *Int. J. High Perform. C.* **30**(1) 3-10