WES-2023-107 Response to Reviewers

Dear Reviewers, dear Editor,

thank you for your time managing and reviewing our work and for your feedback. Based on the Reviewers' suggestions, we have done our best to improve the paper. Before answering your observations in detail we would like to point out that we decided to change the title of this study, from the original *"A Code-to-Code Comparison for Floating Offshore Wind Turbine Simulation in Realistic Environmental Conditions: Quantifying the Impact of Modeling Fidelity on Different Substructure Concepts"* to *"Quantifying the Impact of Modeling Fidelity on Different Substructure Concepts for Floating Offshore Wind Turbines - Part II: Code-to-Code Comparison in Realistic Environmental Conditions"*. We believe this change better links this paper with WES-2023-117, also under review in the WES Journal. WES-2023-117 is in fact "Part I" of this two-part study and lays the groundwork for the considerations done in this study. We hope this change also helps address some of Reviewer's #1 concerns regarding our citation of WES-2023-107 to explain some of the differences we observed in this paper. We believe we should have presented the two papers as a two-part study from the start, as they are deeply linked and part of the same project.

We have provided detailed answers to your comments below, in blue colored text for your convenience.

Best regards,

F. Papi, G. Troise, R. Behrens de Luna, J. Saverin, S. Perez-Becker, D. Marten, M.L. Ducasse, A. Bianchini

∘∘∘ ∘∘∘ ∘∘∘

Reviewer #1 comments:

The paper presents an interesting and comprehensive study on the load and performance of three floating off-shore wind turbines, considering a very wide range of operating conditions, constructed by resorting to a well documented data base of ocean conditions (and already used in previous studies). Three different HAWTs are considered, each one featuring a different floating foundation system. Three different simulations packages, covering two main fidelity levels (BEM and LLFVW), are used to simulate the movement and the load acting on the turbines.

The paper highlights both the effects of the floating operation on the turbine behavior and load as well as the capabilities of the codes to predict such behaviors. In the end, the paper documents a study which is fully relevant for the WES journal and certainly of archival value.

Despite the general positive evaluation of the paper, in my opinion the document could be improved. In the following I provide some remarks that the authors may consider in their review.

Thank you for the constructive feedback and for the general appreciation of our work. Based on your suggestions, we have substantially re-worked the manuscript. Please find below our detailed answers to the specific comments you raised.

1. In general, the paper is difficult to read; it is very long and there is an exaggerated use of acronyms, which complicates the comprehension of the text in several instances. Some sections may also be divided in subsections or, at least, to be divided in more paragraphs. I invite the authors to find a way to make the discussion more synthetic and more effective or schematic, eliminating what is not strictly necessary for the paper (maybe referring to the reports of FLOATECH project). In the results section of the paper, for example, most of the comments are descriptions of the figures and of the differences between the codes, and less space is dedicated to the explanation of the observed differences.

Thank you again for the constructive feedback. This work is the summary of a long and – in our opinion – challenging research project. As such, the first version of this document reflected such complexity and, as the Reviewer pointed out, was probably excessively long and complex. To address this, the results section of the paper was completely re-worked. In particular, we divided the extreme results analysis into more sections (L365 to L413 of revised manuscript). We also streamlined discussion of fatigue loads (L459 to L544 of revised manuscript) by removing Fig. 9 and 12, as they were not essential to the discussion of the differences between the codes. We simplified Figs. 19 and 20, merging them into one and re-worked the tower base fatigue loads discussion. Finally, we reduced the use of acronyms to improve readability. Overall, the main body of text more focused on the code-to-code differences. We hope this change meets the Reviewer's expectations.

2. The conclusions are, in some aspects, weak. For example, the self-excitation phenomenon (one of the most interesting findings of the paper) is predicted in different way by the codes, but the explanation of the over-estimate of Q-Blade code is demanded to another paper (Behrens de Luna et al.), still unpublished; it could have been a relevant discussion and conclusion also of the present paper. As a second example, the results often show differences between the predictions of OpenFAST and DeepLines Wind, despite the two codes are of the same fidelity level; the explanation of these results are demanded to future studies, while one would have expected such discussion in this paper, for example by resorting to the different structural model of OpenFAST or to the compiling/import issues of DeepLines Wind mentioned in the paper (and see below for the remarks on these aspects).

The Reviewer stressed an important point. The conclusion of this study is indeed different from our initial expectation when we started this work and the FLOATECH proposal was written. Before going into this, we were expecting that the additional motion afforded by the floating installation would accentuate the differences found by many between aerodynamic and structural models for onshore wind turbines. Throughout this study, however, we have mostly found small differences between the compared codes. We have tried to highlight this better in the conclusions (L584-588). Regarding the explanation of the self-excitation phenomenon, the paper by Behrens de Luna et al. that we referenced is part of the same work that led to this paper. Based on the reviews we received on both papers, we have decided to change the title of both works to link them together and hopefully improve clarity and scope of both works. Based on the intuition in that paper, we have run additional analyses in OpenFAST and QBlade to shed some light on the possible causes of the phenomenon. Various physical phenomena could, in principle, cause such a difference in excitation. However, by process of exclusion, differences in hydrodynamic excitation are unlikely to be the cause of the increased self-excitation in QBlade, as nearly identical response in QBlade and OpenFAST was noted at the Softwind's pitch natural frequency in part one of this study ([3], Fig. 13). Moreover, the aerodynamic model is also not the cause, as switching to DBEM in QBlade did not improve agreement in this regard with respect to OpenFAST (not shown herein for brevity). As stated previously, OpenFAST doesn't include blade torsion. However, switching to a rigid structure did not improve the agreement of OpenFAST and QBlade. A possible explanation for the difference in blade pitch – platform pitch self-excitation was put forward in part one of this study [3] and is related to increased aerodynamic torque variation in QBlade with respect to the other two codes. Indeed, upon further investigations, differences in the system dynamics, and how they interact with the control system, could explain the observed behavior. As explained in detail by Abbas et al. [2], the controller and turbine can be seen as a closed-loop second-order system, characterized by a natural frequency at a certain operating wind speed:

$$\omega^2 = k_i(U_{op})B = k_i(U_{op})\frac{N_g}{J}\frac{\partial\tau_a}{\partial\beta} \quad (1)$$

where N_g and J are the gearbox ratio and rotor inertia, which are the same in OpenFAST, QBlade, and DeepLines. The higher the natural frequency, the more responsive the system is to an external disturbance such as a platform pitch oscillation. The integral controller gain k_i is also the same in the two codes, as it depends on the controller tuning. The slope of the aerodynamic torque as a function of blade pitch is, however, different in the two codes. The derivative of aerodynamic torque as a function of blade pitch for the mean 11 m/s operating conditions is shown in Fig. 12 (b). As $\frac{\partial\tau_a}{\partial\beta}$ is larger in magnitude for QBlade at the mean operating blade pitch of approximately 0.5°, from eq. 1, ω^2 is also larger, leading to increased self-excitation in QBlade.

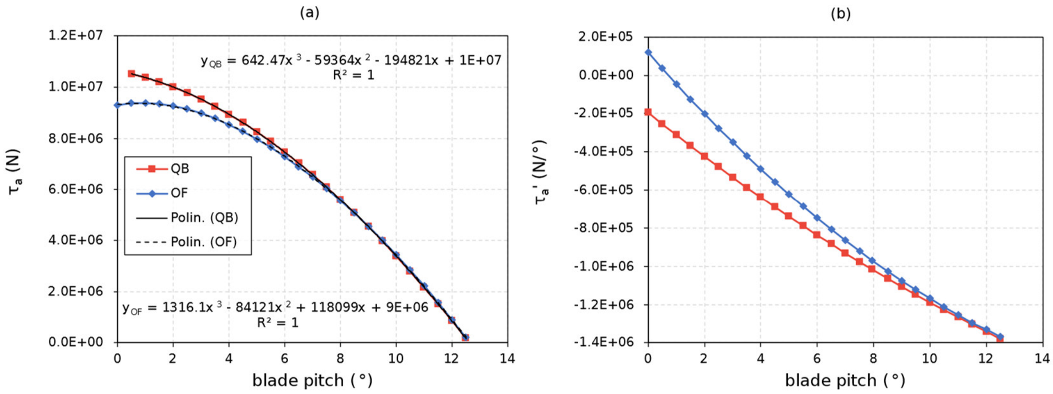

Figure 12: (a) aerodynamic torque as a function of blade pitch for OpenFAST and QBlade for 11 m/s operating TSR, and relative trendlines. (b) derivative of aerodynamic torque as a function of blade pitch computed from analytic derivative of trendlines.

This explanation is now included in the paper (L495-518).

Finally, we are aware of the differences between DeepLines and OpenFAST, despite them being of a similar fidelity level. We have refined the OpenFAST and QBlade results over the span of several months, correcting small bugs that may arise in such a complex set-up and ultimately aligning the models better. DeepLines has not benefitted from such improvements. To this end, we have debated internally whether or not to include DeepLines results. We have ultimately decided to include them despite the set-up issues explained in Section 2.3.2, because general overall agreement with the open-source codes is good and this result is representative of what an industrial partner could achieve with the limited time and budget often connected to industrial processes. We have once more highlighted this better in the conclusions. (L498-522 of revised manuscript)

Detailed comments:

- In section 3.2, the reason for using a simplified structural model in OpenFAST is not convincing; if one wants to see the effect of using a simplified structural model, a comparative analysis must be done changing this particular model within the same simulation framework, and not introducing a further variability in a context of code-to-code comparison. The implications of this choice should be better highlighted in the results section.

In our view, this study is a high-level code-to-code comparison aimed at assessing the impact of multi-fidelity modelling choices in the three codes with respect to each other. In the case of structural modelling in OpenFAST, we could have chosen indeed to use BeamDyn, which is able to model blade torsion in addition to flapwise and edgewise deformation of the blade and is more theoretically sound especially in the case of pre-bent blades. However, the three structural models in the three codes would still have differed, in both theory and implementation. Therefore, we chose to use the simpler approach in OpenFAST, with the intention of investigating the global implications of the overall modelling choices in each code. We agree that the wording used in section 3.2 may be misleading and thus rephrased in the revised manuscript (L254-255 of revised manuscript).

- At the beginning of Section 4, it is mentioned that not all the simulation runs reached convergence in OpenFAST or DeepLines Wind. While this might be expected, it would be interesting to briefly discuss what are the reasons for these failed convergences.

Thank you for the comment. Indeed, disclosing more information in this regard may help others facing similar issues in their simulations. One simulation in OpenFAST did not converge. Upon detailed inspection of the result, the issue seems to be related to numerical instabilities in the structural solver. In DeepLines-wind the issue can be traced back to instabilities in the numerical integration scheme. Despite an initial attempt to solve these issues in both codes, we ultimately did not have the resources to attempt to fine-tune the numerical parameters in the two codes and solve the issues. The beginning of Section 4 has been edited to reflect these changes (L310-314 of revised manuscript).

- In section 4.1, line 307, a compilation issue is mentioned in DeepLines Wind, which has a relevant impact on the results. It is not clear, for the reader, what could be the practical consequences of this issue: if the compilation issue could be easily solved by the Authors, the technical relevance of the results obtained with DeepLines Wind is questionable; if, instead, the issue is an inherent feature of the code, a general improvement to the code is needed. Please explain.

DeepLines Wind uses a different convention for pitch angle than OpenFAST and QBlade. Thus, the ROSCO controller needed to account for this to be coupled to the former. In the re-compiled version, minimum rotor speed is not enforced. This can be very clearly seen in Fig. 3 (b, e). To our best knowledge, the control routine is the same as that used in OpenFAST and QBlade for wind speeds higher than 7 m/s. The discrepancy has a relevant impact on fatigue loads, while we believe it does not affect extreme loads as only low wind speeds are affected. This is acknowledged throughout the results section of the paper. We realize that this is not ideal, but there is no way for us to rectify this issue without removing DeepLines Wind entirely from the comparison, which we would prefer not to do as we believe it is a worthy addition to the code-to-code comparison. We changed the text in section 4.1 to better highlight the influence and causes of this discrepancy (L334-341 of revised manuscript).

- In section 4.2, line 374, an issue of DeepLines Wind in importing the wind field is mentioned. Again, is it an inherent limitation of the code with respect the other ones, or was it a simple issue in the set-up that could have been solved by the Authors?

In medium-fidelity wind turbine simulation tools such as those used in this study wind fields are imported as three-dimensional wind boxes, where the first two dimensions are the height and width of the field and the third is time. Wind boxes are often shifted on import of a time equals to R/U so that the turbine is fully immersed in the wind box even in case of yaw misalignment. Despite using the same wind fields, differences on import ultimately cause the wind fields to be shifted in DeepLines-Wind. This is not a limitation of the examined codes, nor something that we could address with set-up changes. We have added a paragraph to section 4.2 (L405-407 of revised manuscript) to better reflect this.

- Figure 10 (and following similar ones): personally I do not fully appreciate the use of 'cumulative PSD' in the plots, but I recognize its effectiveness with respect to the standard PSD diagram; for clarify, I recommend to specify in the paper, when commenting Figure 1 for the first time, how to read a cumulative PSD diagram.

Thank you for the feedback. Cumulative PSDs are the most effective way we found to showcase the differences between the codes. We agree that they can be confusing for readers that are not accustomed to such metric. We added an explanation on how to read CPSDs when introducing them in Fig. 9 (L459-462 of revised manuscript).

- Figure 20: a significant difference exists between the Q-blade and DeepLines Wind in the frequency range 0.4-0.6 Hz, the authors are encouraged to comment this aspect in the paper.

Indeed, there is a difference in this frequency range in the tower base fore-aft bending moment of QBlade and DeepLines wind. In this frequency range, a fore-aft mode of the Hexafloat structure, where the upper floater structure oscillates 180° out-of-phase with the rotor collective flapwise mode. Although we were unable to find the root cause of this difference, the difference between the two codes is most likely liked to them capturing this mode differently. It must be noted that the difference is amplified by the semi-log scale of Fig. 20 and is in reality very small.

∘∘∘

[1] Papi, F., Bianchini, A., Troise, G., Mirra, G., Marten, D., Saverin, J., Behrens de Luna, R., Ducasse, M.-L., and Honnet, J.: Deliverable 2.4 Full report on the estimated reduction of uncertainty in comparison to the state-of-the-art codes OpenFAST and DeepLines Wind, 2023.

[2] Abbas, N. J., Zalkind, D. S., Pao, L., and Wright, A.: A reference open-source controller for fixed and floating offshore wind turbines, Wind Energy Science, 7, 53–73, https://doi.org/10.5194/wes-7-53-2022, 2022.

[3] Behrens De Luna, R., Perez-Becker, S., Saverin, J., Marten, D., Papi, F., Ducasse, M.-L., Bonnefoy, F., Bianchini, A., Nayeri, C. N., and Paschereit, C. O.: Verifying QBlade-Ocean: A Hydrodynamic Extension to the Wind Turbine Simulation Tool QBlade, Wind Energy Science Discussions, 1–36, https://doi.org/10.5194/wes-2023-117, 2023.