Dear Editor, Dear Referees,

We would like to thank you for further considering our manuscript, and for the review by the two new referees. Please find below a detailed point-by-point answer to the referees' comments.

The authors.

Nomenclature:

RXCY – Referee number X Comment number Y
AR – Authors' Reply
O-LX – Original manuscript Line X
R-LX – Revised manuscript Line X

Referee #3 (Anonymous)

R3C1: Line 76-77: The author highlights the capability of the C++ core of this package in handling large datasets; however, there is a lack of scalability tests or comparisons with other models to support this claim.

AR: The tool has already been used within our team on operational multi-model ensemble forecasting datasets produced by the national hydrological drought forecasting online platform PREMHYCE (Nicolle et al., 2020; Tilmant et al., 2020) even though this has not lead to a publication yet. However, the tool has also been used successfully in a recent multi-model large sample study (Thébault, 2023). This reference has been added to the article (see R-L78) to support the claim. This comes as evidence that it can handle large datasets. However, we are not claiming that other packages cannot handle large datasets as well, so a comparison with other packages does not seem essential here.

R3C2: Table 2, Table 3: Parentheses should be used for open range indicators.

AR: Thank you for making us aware of this international mathematical convention, different from our local convention on open ranges. Ranges in Tables 2 and 3 have been amended in the revised manuscript to use parentheses for the relevant metric ranges.

R3C3: Line 91: Add "evalhyd" as the name of one of the tools.

AR: Thank you. We have added the name of the tool (see R-L91).

R3C4: Section 3.3: Consider using a general notation of XNxd and provide examples of d as listed.

AR: Thank you for the suggestion. Our notations were indeed not quite followinf any standard notation. We have followed your recommendation and used the more formal mathematical notation for tensors/multi-dimensional arrays (see R-L103-117).

R3C5: Preprocessing functionalities provided in 4.2-4.4 may be considered trivial, as other packages offer more extensive capabilities. It is suggested that the authors continue expanding this package to incorporate additional methods for broader usage.

AR: Yes, we agree with the reviewer that there is scope for additional functionalities, namely regarding preprocessing aspects. In particular, computing metrics on flow statistics (e.g. mean) applied on sliding windows, under a given threshold, on standardised flow indicators (e.g. QMNA [https://fr.wikipedia.org/wiki/QMNA, in French], VCN3 [https://fr.wikipedia.org/wiki/VCN3, in French] commonly used in France) could be useful. These will be considered for future versions of the tool. We have added these suggestions in the conclusions and perspectives section (see R-L317-319).

R3C6: Figure 6: Instead of plotting a 1:1 line, plotting the residual of evalhyd minus Harrigan et al. (2023) results against lead time may be more meaningful.

AR: We have tried to produce the figure with the residuals instead (see **Figure 1**) but we do not believe that these residuals are meaningful. Indeed, the results reported in Harrigan et al. (2023) only provided a precision of two decimals. If we apply a two-decimal rounding on our results with evalhyd, we obtain **Figure 2** instead. We can notice that the residuals are very often equal to zero, with only a handful of exceptions. Those exceptions are most likely due to some marginal numerical precision differences and/or rounding applied at different stages of the data processing. Therefore, we do not believe these to be meaningful and we would prefer to keep the original figure that we find more easily understandable.

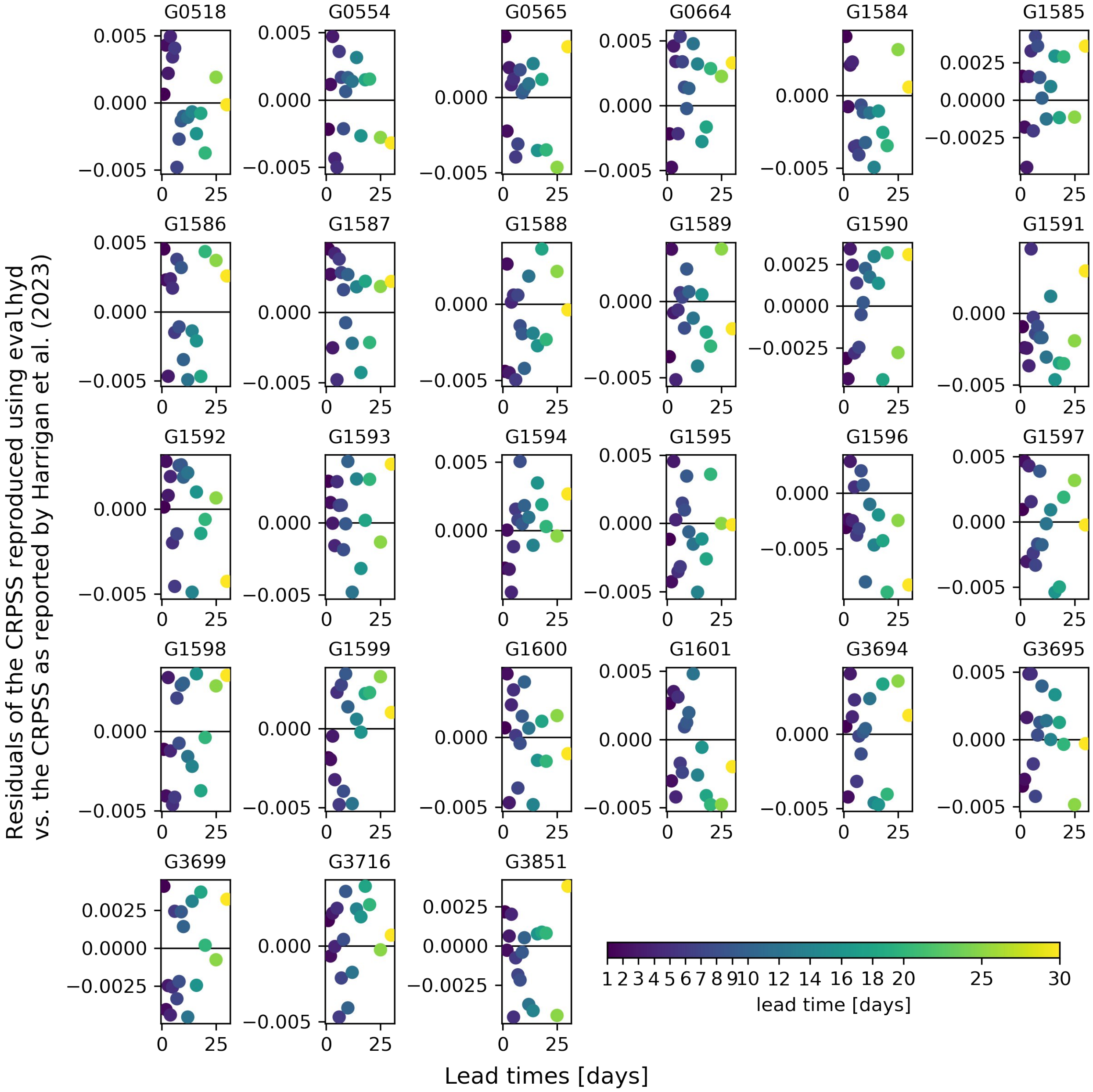

Figure 1: CRPSS residuals (i.e. difference between the CRPSS reported in Harrigan et al. (2023) and the CRPSS calculated using evalhyd, applying no rounding on the latter).

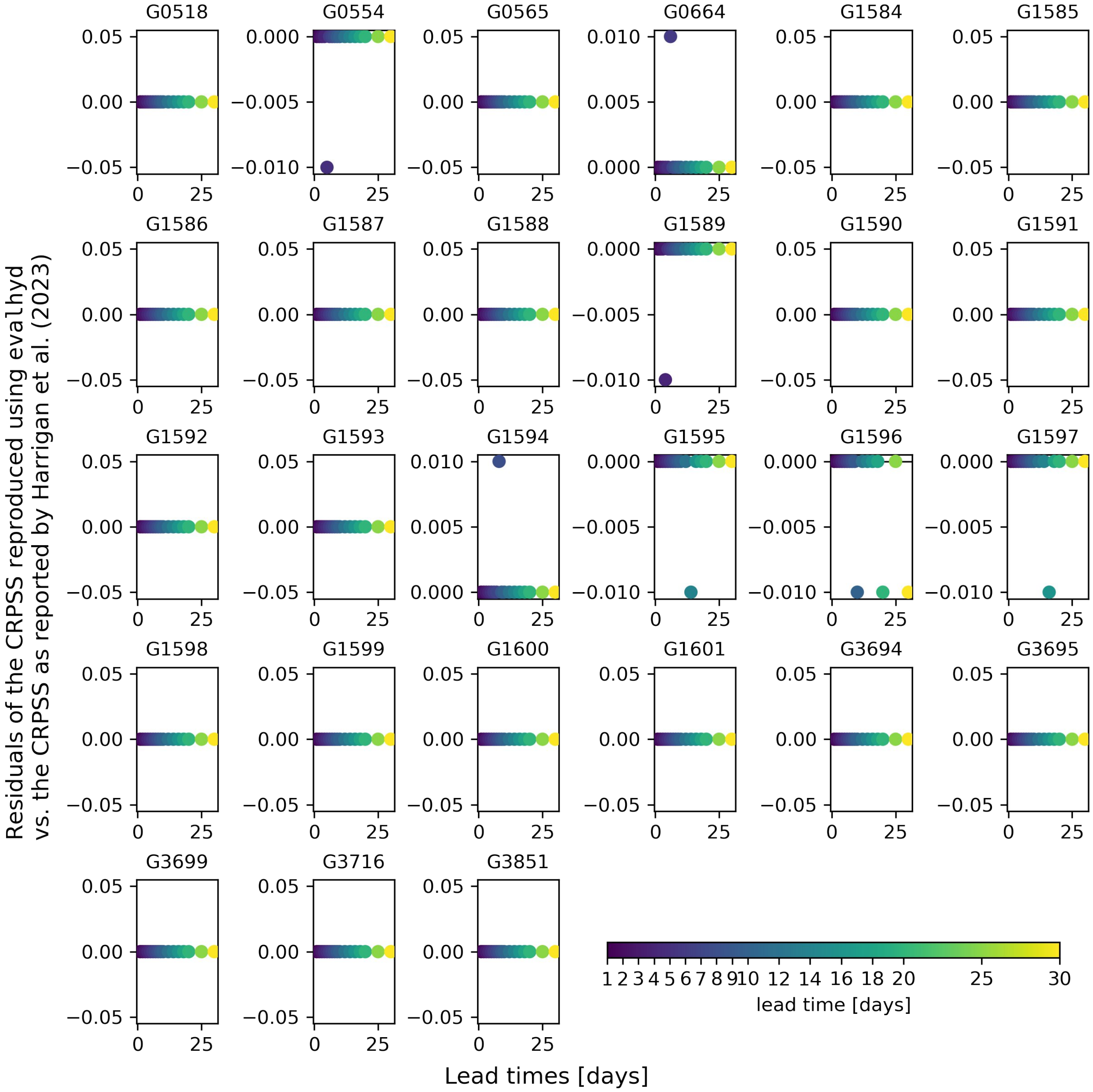

Figure 2: CRPSS residuals (i.e. difference between the CRPSS reported in Harrigan et al. (2023) and the CRPSS calculated using evalhyd, applying a rounding at the second decimal on the latter to match the precision provided in the former).

Referee #4 (Anonymous)

R4C1: The tool developed in this article is an open-source tool with multiple language versions. As mentioned in the article, the tool can be used in Python, R, and C++. I would like to know if it is possible to use it in more language environments in the future, such as Java, JavaScript, etc.

AR: We do already touch upon the scope for more languages to be supported in the future at the end of section 3.1 (i.e. Julia and Octave). As *evalhyd* relies on the C++ core library *xtensor* for vectorised numerical computations, only Julia is reasonably close to offer an interface as the bindings between this language and C++ already exist, languages such as Octave, Java, and JavaScript would require for such bindings to be developed by the *xtensor* team or some independent contributors. While we cannot deny that there are likely Java/Javascript users in the hydrological community, these do not appear as the main languages favoured by the community in recent years. For instance, while the Ensemble Verification System (EVS) was developed by NOAA in Java (Brown et al., 2010), this was back in 2010, and more recent tools are now more often developed in Python or in R.

R4C2: In the introduction section, the article introduces some existing tools and introduces the shortcomings of these existing tools. I personally believe that the new tool developed in this article can be used to make a detailed comparison with existing tools, including comparing features, performance indicators, or usability, highlighting the advantages of the new tool.

AR: We already mention the similarities and the innovations of our tool compared to existing noteworthy tools. We agree with the reviewer that an exhaustive comparison with other existing tools would be useful. However, we believe that the choice of criteria for comparison would need to be made by a diverse group of international experts in order not to be biased towards one tool or another and establish an independent benchmark. Indeed, it is likely that if we come up with the list of criteria, despite our best efforts, we would certainly mostly consider the aspects/functionalities relevant to our own practices, and overlook others we did not think of. It is clear that all tools have their pros and cons, and they all have their best usage context. But what crucial design aspect of our tool that stands out is the polyglot character of it, making it potentially accessible to a large pool of users. The regular workshops of the HEPEX community may offer the opportunity to produce an independent benchmark.

R4C3: Personally, I think that nonprofessional programmers may be limited by the interactive performance of this tool. I can consider setting up some user guides or tutorials, which may make the tool easier to use by a wider audience and facilitate future expansion of the tool.

AR: The online documentation accessible at https://hydrogr.github.io/evalhyd/, as mentioned in the conclusions of the article, provides user guides and API references. However, we agree with the reviewer that tutorials were lacking. We have added a tutorial using the GloFAS data presented in the paper in the Python section of our online documentation. We will produce equivalent tutorials in the near future for the other *evalhyd* bindings.

R4C4: In the display section of the tool, a series of image results were used for display. Personally, I think it is possible to consider providing a more detailed introduction to the display results so that readers can understand the effectiveness of the tool.

AR: All of the figures presented in the illustrative example section are already introduced in the text to provide such introduction to the results. In addition, we believe that the captions of those figures are already quite lengthy and provide the details needed to make sure that they can be understood without relying on the main text. We are unsure how to further improve on these aspects in order to remedy this comment.

References

Brown, J., Demargne, J., Seo, D.-J., Liu, Y. (2010). The Ensemble Verification System (EVS): A software tool for verifying ensemble forecasts of hydrometeorological and hydrologic variables at discrete locations. Environmental Modelling & Software. 25. 854-872. https://doi.org/10.1016/j.envsoft.2010.01.009.

Harrigan, S., Zsoter, E., Cloke, H., Salamon, P., and Prudhomme, C. (2023). Daily ensemble river discharge reforecasts and real-time forecasts from the operational Global Flood Awareness System, Hydrology and Earth System Sciences, 27, 1–19, https://doi.org/10.5194/hess-27-1-2023.

Nicolle, P., et al. (2020). PREMHYCE: An operational tool for low-flow forecasting, Proc. IAHS, 383, 381–389, https://doi.org/10.5194/piahs-383-381-2020.

Tilmant, F., et al. (2020). PREMHYCE: an operational tool for low-flow forecasting, La Houille Blanche, 106:5, 37-44. https://10.1051/lhb/2020043.

Thébault, C. (2023). Quels apports d'une approche multi-modèle semi-distribuée pour la prévision des débits ? PhD thesis (in French), Sorbonne Université. https://theses.hal.science/tel-04519745.