EGU24-18057, updated on 28 Apr 2024
https://doi.org/10.5194/egusphere-egu24-18057
EGU General Assembly 2024

Tools and techniques for modular, portable (Machine Learning) parameterisations

Jack Atkinson[1], Dominic Orchard[1,2], Elliott Kasoar[1,3], and Thomas Meltzer[1]
[1]Institute of Computing for Climate Science, Research Computing, University of Cambridge, Cambridge, United Kingdom
[2]School of Computing, University of Kent, Cantebury, United Kingdom
[3]Science and Technology Facilities Council, London, United Kingdom

Numerical models across the geophysical sciences make extensive use of physical parameterisations to represent sub-grid processes such as eddies, convection, gravity waves, and microphysics. With the advent of machine learning (ML), big data, and artificial intelligence there are increasing efforts to develop data-driven parameterisation schemes or emulate existing, but computationally-intensive, parameterisations using ML techniques.

Despite their name, the irony is that traditional design approaches often lead to parameterisations which are not themselves parameters, but rather tightly integrated, enmeshed pieces of code in a larger model. Such parameterisations and practices pose a threat to the reproducibility, accuracy, ease-of-(re)-use, and general FAIRness (Findable, Accessible, Interoperable, Reusable) [1] of the schemes being developed.

In contrast, a modular approach to parameterisations (and their receivers, e.g., GCMs), would enable them to be more easily (1) interchangeable and Interoperable, to compare different schemes and assess uncertainty due to their inherent approximate behaviour, and (2) portable and Reusable, between models, to reduce engineering effort, and (3) understandable and Accessible, by being clear about dependencies and the physical meaning of program variables, and (4) testable, to aid verification and correctness testing.

Achieving this goal in the context of numerical modelling brings a number of scientific, engineering, and computational challenges. In this talk we aim to set out some best-practice principles for achieving modular parameterisation design for geoscience modellers. We then focus on the particular challenges around modem ML parameterisations.

To this end we have developed a library for easily interoperating ML-based parameterisations in PyTorch with Fortran-based numerical models, called FTorch [2]. By reducing the Fortran-PyTorch Interoperability burden on researchers this framework should reduce errors that arise and increase the speed of development when compared to other approaches such as re-coding models in Fortran. FTorch aims to make emerging ML parameterisation research more Accessible to those who may not have deep expertise of ML, Fortran, and/or computer science. It also means that models developed using PyTorch can leverage its feature-rich libraries and be shared in their

native format, maximising Reusability. We discuss the design principles behind FTorch in the context of modular parameterisations and demonstrate our principles and approach by coupling ML parameterisations to atmospheric and climate models.

In general, we present a number of considerations that could be used to make all parameterisation schemes more easily Interoperable and Re-useable by their developers.

[1] Barker, M. et al, Introducing the FAIR Principles for research software, *Sci Data* **9**, 622 (2022) https://doi.org/10.1038/s41597-022-01710-x

[2] FTorch https://github.com/Cambridge-ICCS/FTorch