

IndividualDisplacements.jl: a Julia package to simulate and study particle displacements within the climate system

Gael Forget¹

1 MIT, EAPS

DOI: [10.21105/joss.02813](https://doi.org/10.21105/joss.02813)

Software

- [Review](#) ↗
- [Repository](#) ↗
- [Archive](#) ↗

Editor: [Kristen Thyng](#) ↗

Reviewers:

- [@milankl](#)
- [@visr](#)

Submitted: 01 October 2020

Published: 26 April 2021

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

Summary

`IndividualDisplacements.jl` ([Gael Forget, 2021](#)) is focused on computation and analysis of individual displacements across the global climate system. *Individuals* is used here as a generic term to represent points, particles, parcels, materials, etc that tend to be carried around the Earth by geophysical fluids (via oceanic currents or atmospheric flows for example). Their *displacements*, or trajectories, are computed by deriving pointwise velocities from flow fields provided by the user and integrating over time (e.g., [Figure 1](#) and [Figure 2](#)).

This Julia package is notably aimed at the analysis of numerical models that simulate Oceanic and / or Atmospheric transport processes on staggered C-grids – those most often used in both regional and global modeling. Inter-operability with global climate model grids like the cube-sphere and lat-lon-cap grid (e.g., [G. Forget et al., 2015](#)) is a key feature enabled by `MeshArrays.jl` ([JuliaCon 2018 \(video link\)](#), [Gael Forget et al., 2020](#)). The chosen approach also readily supports simpler grids (cartesian, spherical, curvilinear grids) often used in process-oriented models, regional models, or satellite data products.

`IndividualDisplacements.jl` is thus readily suited to exploit climate model output and other gridded data sets in research projects that involve, for example, tracking plankton communities, heat storage, or plastic garbage patches within the Ocean; or dust, water, or chemical compounds within the Atmosphere. To achieve generality and interpretability, `IndividualDisplacements.jl` first defines two data structures (`FlowFields` and `Individuals`) that allow for simple, flexible user specifications of flow fields, initial positions, etc. It then adds a high-level API such that integrating trajectories for individuals I amounts to a single function call, $f!(I)$.

Internally, the package currently employs `OrdinaryDiffEq.jl` to integrate particle motions over time (within the $f!(I)$ function call), `NetCDF.jl` and `CSV.jl` for I/O, and `DataFrames.jl` for diagnostics computed along particle trajectories (within $f!(I)$ or afterwards). The initial test suite (see [Gael Forget, 2021](#)) is based not only on idealized flow fields and toy-models, but also on data-constrained Ocean simulations from the OCCA/ECCO projects ([Gael Forget, 2018, 2010](#); [G. Forget et al., 2015](#)), which are also used in marine ecosystem simulations in CBIOMES ([Gael Forget, 2019](#)). The OCCA/ECCO gridded ocean circulation estimates are retrieved from a permanent dataverse archive ([Gael Forget, 2016a, 2016b](#)) automatically via the Julia Artifacts system.

`IndividualDisplacements.jl` is also intended to facilitate research involving model-data comparison, data assimilation, or machine learning by providing basic interfaces to other displacement data sets. To start the package provides some support for ingesting (1) data collected in the field by the Global Ocean Drifter Program and Argo array of drifting buoys, and (2) trajectories simulated internally, *online*, by the MIT general circulation model.

The examples folder, which is unit tested upon building the documentation hosted online for each code revision, already covers several common grid cases, two-dimensional and three-dimensional flows, steady-state and time-variable flows. It demonstrates interpolation and diagnostics methods, three plotting libraries, and two notebook systems. Minor parameter changes in these examples should suffice to start applying `IndividualDisplacements.jl` to any other MITgcm simulation of e.g. atmospheric or oceanic turbulence (Campin et al., 2020).

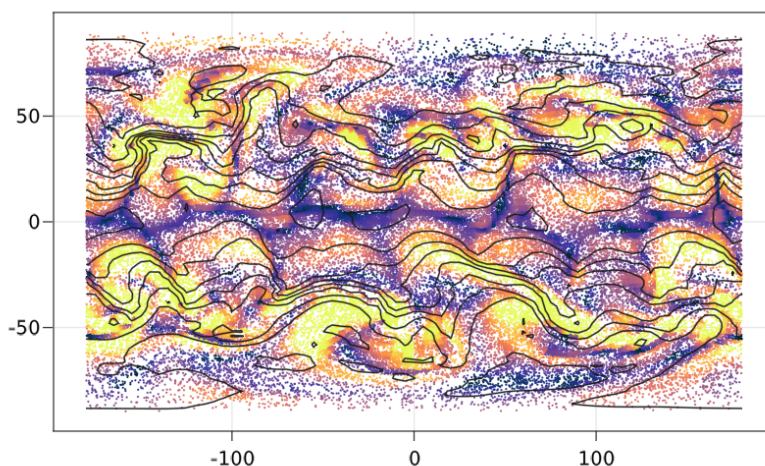


Figure 1: Atmosphere simulation example. Hundred thousand particles are displaced by time-varying 2D flow fields provided by MITgcm on a cube-sphere grid (Campin et al., 2020) using `MITgcmTools.jl`. Particle colors show velocity while contours show temperature.

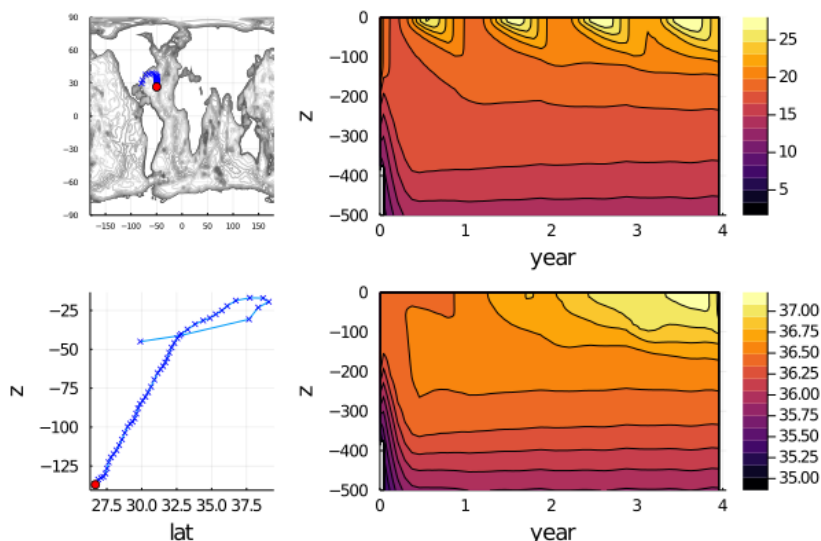


Figure 2: Ocean simulation example. A particle is displaced (left panels) by time-varying 3D flow fields from a global ocean state estimate (G. Forget et al., 2015) using `OceanStateEstimation.jl`. Vertical profiles of temperature and salinity are recorded along the way (right panels).

Statement of need

Lagrangian simulation and analysis frameworks such as `IndividualDisplacements.jl` have been widely used across scientific domains involved with geophysical fluids for decades; see [Pisso et al. \(2019\)](#) or [Van Sebille et al. \(2018\)](#) for recent reviews in, respectively, oceanography or atmospheric sciences. `IndividualDisplacements.jl` is most directly related to the `MITgcm/flt` Fortran package which the author also recently extended ([Campin et al., 2020](#)). This new Julia package was in fact partly motivated by a need to provide an easier and simpler alternative to using `MITgcm/flt` in its offline mode in order to handle global grids not readily supported by other packages ([G. Forget et al., 2015](#); [Gaël Forget, 2020](#); [Rousselet et al., 2020](#)).

`IndividualDisplacements.jl` notably aims to provide a bridge between the vast community of domain experts, who are typically used to older languages like Fortran, C, Matlab, or Python, and the rapidly growing Julia community and package ecosystem. Julia's native parallelism, GPU support, custom array types, differential programming tools, and plotting libraries indeed all offer great new opportunities for exploiting state of the art HPC climate model simulations and the large data sets that they routinely generate.

The development of `IndividualDisplacements.jl` was motivated not just by our scientific needs and ongoing research projects (see below) but also by the need for more effective tools for climate education, citizen science, and advocacy. In the documentation and examples folder, this is highlighted by the use of unicode in the API, `Pluto.jl` notebooks, and `Makie.jl` animations, which showcase Julia's expressivity and reactivity.

`IndividualDisplacements.jl` readily supports all common `MITgcm` grids and configurations via `MeshArrays.jl` and `MITgcmTools.jl` (incl. Ocean, Atmosphere, sea-ice, bio-geochemistry, and ecology). This, by itself, yields a large pool of potential scientific applications and expected users; including via the various ongoing research projects that rely on adjoint-optimized, data-constrained solutions from the OCCA/ECCO projects ([Gaël Forget, 2010](#); [G. Forget et al., 2015](#)).

At this stage, `IndividualDisplacements.jl` is considered production-ready, such that we could immediately start transferring several ongoing research collaborations that use `MITgcm` and ECCO ([Gaël Forget & Ferreira, 2019](#); [Rousselet et al., 2020](#)) from Fortran to Julia. Extension to other models may start with MERRA2 / `MITgcm` coupled model runs ([Strobach et al., 2020](#)) and near-term applications could also include data assimilation to better simulate plastic garbage patches ([Peytavin et al., 2021](#)).

The documentation of `IndividualDisplacements.jl` is also now considered sufficient for welcoming additional contributors. Integration with Julia packages developed in `JuliaClimate`, `JuliaOcean`, `JuliaGeo`, `JuliaStats`, `JuliaDynamics`, and other relevant organizations is thus expected to intensify moving forward.

Acknowledgements

We acknowledge contributions from the open source community at large, the paper reviewers and journal editor, as well as developers of Julia and its packages upon which `IndividualDisplacements.jl` crucially depends.

Funding that supported this work was provided by the Simons Collaboration on Computational Biogeochemical Modeling of Marine Ecosystems (CBIOMES) (grant no. 549931) and the National Aeronautic and Space Administration (grant NASA 19-PO19-0001 from NNH19ZDA001N-PO, grant NASA 80NSSC17K0561 from NNH16ZDA001N-IDS, and grant 19-MAP19-0018 from NNH19ZDA001N-MAP).

References

- Campin, J.-M., Heimbach, P., Losch, M., Forget, G., edhill3, Adcroft, A., amolod, Menemenlis, D., dfer22, Hill, C., Jahn, O., Scott, J., stephdut, Mazloff, M., Fox-Kemper, B., antnguyen13, Doddridge, E., Fenty, I., Bates, M., ... dussin, raphael. (2020). *MITgcm/MITgcm: Mid 2020 version* (Version checkpoint67s) [Computer software]. Zenodo. <https://doi.org/10.5281/zenodo.3967889>
- Forget, Gaël. (2021). *JuliaClimate/IndividualDisplacements.jl: v0.2.10* (Version v0.2.10) [Computer software]. Zenodo. <https://doi.org/10.5281/zenodo.4517441>
- Forget, Gaël. (2018). *Gaelforget/ECCOV4: Documentation updates* (Version v1.8) [Computer software]. Zenodo. <https://doi.org/10.5281/zenodo.1211363>
- Forget, Gaël. (2019). *CBIOMES/global-ocean-model: Update MITgcm & DarwinProject elements* (Version v0.1.0) [Computer software]. Zenodo. <https://doi.org/10.5281/zenodo.2653669>
- Forget, Gaël. (2010). Mapping Ocean Observations in a Dynamical Framework: A 2004–06 Ocean Atlas. *Journal of Physical Oceanography*, 40(6), 1201–1221. <https://doi.org/10.1175/2009JPO4043.1>
- Forget, Gaël. (2020). *Gaelforget/ECCOV4flt_offline: Initial public release* (Version v0.2) [Computer software]. Zenodo. <https://doi.org/10.5281/zenodo.4193582>
- Forget, Gaël. (2016a). *ECCO version 4 release 2: Monthly 1992-2011 time series*. <https://doi.org/10.7910/DVN/NXYKDW>
- Forget, Gaël. (2016b). *OCCA monthly ocean atlas* (Version V1) [Data set]. Harvard Dataverse. <https://doi.org/10.7910/DVN/RNXA2A>
- Forget, G., Campin, J.-M., Heimbach, P., Hill, C. N., Ponte, R. M., & Wunsch, C. (2015). ECCO version 4: an integrated framework for non-linear inverse modeling and global ocean state estimation. *Geoscientific Model Development*, 8(10), 3071–3104. <https://doi.org/10.5194/gmd-8-3071-2015>
- Forget, Gaël, & Ferreira, D. (2019). Global ocean heat transport dominated by heat export from the tropical Pacific. *Nature Geoscience*. <https://doi.org/10.1038/s41561-019-0333-7>
- Forget, Gaël, Milechin, L., Ekre, F., & TagBot, J. (2020). *JuliaClimate/MeshArrays.jl: v0.2.12* (Version v0.2.12) [Computer software]. Zenodo. <https://doi.org/10.5281/zenodo.3995076>
- Peytavin, A., Sainte-Rose, B., Forget, G., & Campin, J.-M. (2021). Ocean plastic assimilator v0.1: Assimilation of plastics concentration data into lagrangian dispersion models. *Geoscientific Model Development Discussions*, 2021, 1–17. <https://doi.org/10.5194/gmd-2020-385>
- Pisso, I., Sollum, E., Grythe, H., Kristiansen, N. I., Cassiani, M., Eckhardt, S., Arnold, D., Morton, D., Thompson, R. L., Groot Zwaaftink, C. D., Evangeliou, N., Sodemann, H., Haimberger, L., Henne, S., Brunner, D., Burkhart, J. F., Fouilloux, A., Brioude, J., Philipp, A., ... Stohl, A. (2019). The lagrangian particle dispersion model FLEXPART version 10.4. *Geoscientific Model Development*, 12(12), 4955–4997. <https://doi.org/10.5194/gmd-12-4955-2019>
- Rousselet, L., Cessi, P., & Forget, G. (2020). Routes of the upper branch of the atlantic meridional overturning circulation according to an ocean state estimate. *Geophysical Research Letters*, 47(18), e2020GL089137. <https://doi.org/10.1029/2020GL089137>

- Strobach, E., Molod, A., Trayanov, A., Forget, G., Campin, J.-M., Hill, C., & Menemenlis, D. (2020). Three-to-six-day air–sea oscillation in models and observations. *Geophysical Research Letters*, 47(10), e2019GL085837. <https://doi.org/10.1029/2019GL085837>
- Van Sebille, E., Griffies, S. M., Abernathey, R., Adams, T. P., Berloff, P., Biastoch, A., Blanke, B., Chassignet, E. P., Cheng, Y., Cotter, C. J., Deleersnijder, E., Döös, K., Drake, H. F., Drijfhout, S., Gary, S. F., Heemink, A. W., Kjellsson, J., Koszalka, I. M., Lange, M., ... Zika, J. D. (2018). Lagrangian ocean analysis: Fundamentals and practices. *Ocean Modelling*, 121, 49–75. <https://doi.org/10.1016/j.ocemod.2017.11.008>