The Journal of Open Source Software

OceanBioME.jl: A flexible environment for modelling the coupled interactions between ocean biogeochemistry and physics

Jago Strong-Wright [1,2¶], **Si Chen** [1,2], **Navid C Constantinou** [3,4], **Simone Silvestri** [5], **Gregory LeClaire Wagner** [5], **and John R Taylor** [1,2]

1 Department of Applied Mathematics and Theoretical Physics, University of Cambridge, Cambridge, United Kingdom **2** Centre for Climate Repair, Cambridge, United Kingdom **3** Australian National University, Australia **4** Australian Research Council Centre for Climate Extremes **5** Massachusetts Institute of Technology, USA ¶ Corresponding author

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor: Open Journals

Reviewers:

- @openjournals

Submitted: 01 January 1970
Published: unpublished

Summary

`OceanBioME.jl` is a flexible modelling environment written in Julia (Bezanson et al., 2017) for modelling the coupled interactions between ocean biogeochemistry, carbonate chemistry, and physics. `OceanBioME.jl` can be used as a stand-alone box model, or integrated into `Oceananigans.jl` (Ramadhan et al., 2020) simulations of ocean dynamics in one, two, or three dimensions. As a result, `OceanBioME.jl` and `Oceananigans.jl` can be used to simulate the biogeochemical response across an enormous range of scales: from surface boundary layer turbulence at the meter scale to eddying global ocean simulations at the planetary scale, and on computational systems ranging from laptops to supercomputers. An example of a problem involving small-scale flow features is shown in Figure 1, which shows a simulation of a sub-mesoscale eddy in a 1km x 1km horizontal domain with an intermediate complexity biogeochemical model and a kelp growth model solved along the trajectories of drifting buoys. `OceanBioME.jl` leverages Julia's multiple dispatch and effective inline capabilities to fuse its computations directly into existing `Oceananigans.jl` kernels, thus maintaining `Oceananigans.jl`'s bespoke performance, memory- and cost-efficiency on GPUs in `OceanBioME.jl`-augmented simulations.

The Journal of Open Source Software

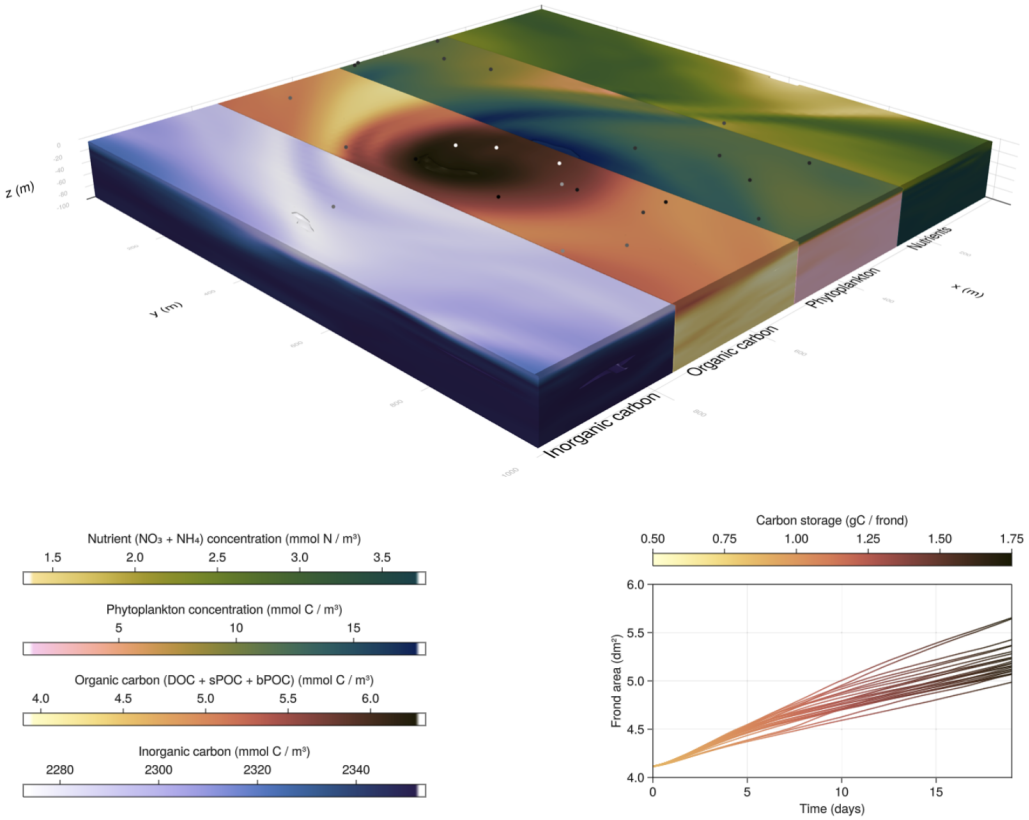

Figure 1: Here we replicate the Eady problem where a background buoyancy gradient and corresponding thermal wind generate a sub-mesoscale eddy, roughly following the setup of Taylor (2016). To this physical setup, we added a medium complexity (9 tracers) biogeochemical model, some of which are shown above. On top of this, we added particles modelling the growth of sugar kelp which are free-floating and advected by the flow, and carbon dioxide exchange from the air. Thanks to Julia's speed and efficiency the above model (1 km $\times$ 1 km $\times$ 100 m with 64 $\times$ 64 $\times$ 16 grid points) took about 30 minutes of computing time to simulate 10 days of evolution on an Nvidia P100 GPU. Figure made with `Makie.jl` (Danisch & Krumbiegel, 2021).

`OceanBioME.jl` is built with a highly modular design that allows user control and customization. There are two distinct module types implemented in `OceanBioME.jl`:

- First, tracer-based ecosystem modules are formulated in `AdvectedPopulations` as a set of coupled ordinary differential equations. These equations can be solved by `OceanBioME.jl` as box models, which is particularly useful for testing. The same equations can be integrated by `Oceananigans.jl` to provide tracer-based ecosystem models.
- The second module type is `Individual` "biologically active" particles. These consist of individual-based models which are solved along particle paths and can be coupled with the tracer-based modules and physics from `Oceananigans.jl`. The biologically active particles can be advected by the currents, and/or they can move according to prescribed dynamics. For example, migrating zooplankton or fish can be modelled with biologically active particles and `OceanBioME.jl` allows these to interact with tracer-based components such as phytoplankton or oxygen.

`AdvectedPopulations` are supported by `Boundaries` modules which provide information at the top and bottom of the ocean. For example, the GasExchange submodule calculates the flux of carbon dioxide and oxygen at the sea surface, while the `Sediments` modules calculates fluxes of carbon and oxygen at the seafloor.

We provide a simple framework and utilities (such as light attenuation integration) to build the necessary components of biogeochemical models. With the provided models, currently a simple Nutrient-Phytoplankton-Zooplankton-Detritus (Kuhn et al., 2015) model, and an intermediate complexity model, LOBSTER (Lévy et al., 2005), we have set up a straightforward "plug and play" framework to add additional tracers such as carbonate and oxygen chemistry systems, and additional forcing. Additionally, we have implemented comprehensive air-sea flux models (e.g. Wanninkhof, 1992) and sediment models (e.g. Soetaert et al., 2000) which can easily be applied to tracers in the models. We focus on the simulation of idealized sub-mesoscale systems, but this flexible framework allows users to model problems of any scale. This framework is made possible by our contributions to `Oceananigans.jl`, adding a streamlined user interface to swap biogeochemical models with no modification to other model configurations. This interface also facilitates rapid prototyping, as models can be implemented and swapped easily by just extending a few key functions. This flexibility and ease-of-use is unmatched in existing biogeochemical models.

`OceanBioME.jl` was designed specifically to study ocean carbon dioxide removal (OCDR) strategies. Assessing the effectiveness and impacts of OCDR is challenging due to the complexities of the interactions between the biological, chemical, and physical processes involved in the carbon cycle. Moreover, field trials of OCDR interventions are generally small-scale and targeted, while the intervention required to have a climate-scale impact is regional or global. We have built `OceanBioME.jl` to meet these challenges by creating tools that provide a modular interface to the different components within the ocean modelling framework provided by `Oceananigans.jl`. This allows easy access to a suite of biogeochemical models ranging from simple idealized to full-complexity models. Figure 2 shows a simple column model with an OCDR intervention (macroalgae growth) added after a warm-up period, which increases the carbon export of the system.

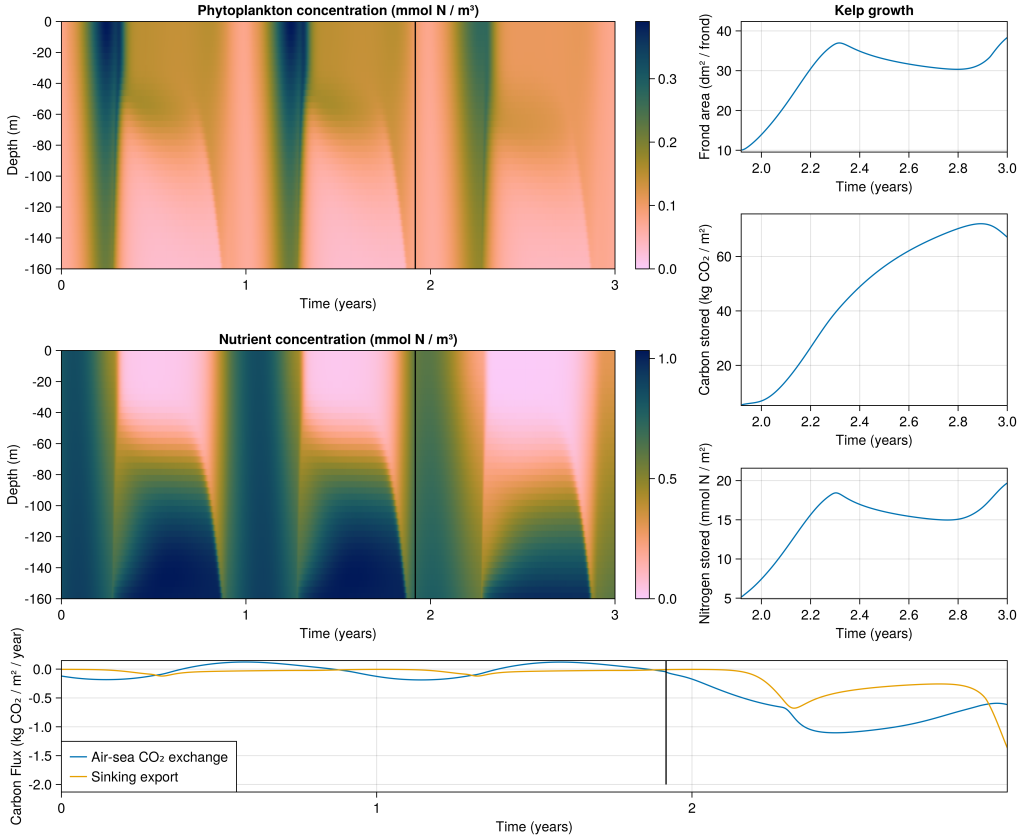

Figure 2: Here we show the results of a 1D model, forced by idealised light and mixing, which qualitatively reproduces the biogeochemical cycles in the North Atlantic. We then add kelp (500 frond / m^2 in the top 50 m of water) in December of the 2nd year (black vertical line) which causes an increase in air-sea carbon dioxide exchange and sinking export, as well as a change in the phytoplankton growth cycle. Figure made with `Makie.jl` (Danisch & Krumbiegel, 2021).

The biologically active particles built into `OceanBioME.jl` are particularly useful for OCDR applications. Accurate carbon accounting is essential for assessing the effectiveness of OCDR strategies. Biologically active particles can be used to track carbon from a particular source while accounting for interactions with its surroundings. Biologically active particles can also be used to model OCDR deployment strategies including seaweed cultivation, alkalinity enhancement, and marine biomass regeneration. `OceanBioME.jl` currently includes an extended version of the sugar kelp model presented by Broch & Slagstad (2012) as an example of the utility and implementation of these features.

The implementation of OceanBioME.jl models allows for seamless integration with data assimilation packages, such as `EnsembleKalmanProcesses.jl` (Dunbar et al., 2022). This feature facilitates rapid calibration of model parameters, providing a powerful utility for integrating observations and models, with the potential to improve model skill and identify key sources of uncertainty.

A key metric for the validity of biogeochemical systems is the conservation of elements such as carbon and nitrogen in the system. We therefore continuously test the implemented models in a variety of simple scenarios (i.e. isolated, with/without air-sea flux, with/without sediment) to ensure that conservation conditions are met, and we will continue to add tests for any new models. Additionally, we check `OceanBioME.jl` utilities through standard tests such as comparison to analytical solutions for light attenuation, and conservation of tracers for active particle exudation and sinking.

Finally, this software is currently facilitating multiple research projects into ocean CDR which would have been significantly harder with other solutions. For example, Chen (In prep.) is using the active particle coupling provided to investigate the effects of location and planting density of kelp in the open ocean on their carbon drawdown effect, as in the example above. Additionally, Strong-Wright (In prep.) is using the coupling of both the biogeochemistry and easy interface to couple the physics to study flow interactions with a fully resolved giant kelp forest model including the effects on nutrient transport and distribution.

Acknowledgements

We would like to thank the Climate Modeling Alliance team and `Oceananigans.jl` contributors for their fantastic project. We are also very grateful for the support and funding of the Centre for Climate Repair, Cambridge and the Gordon and Betty Moore Foundation.

References

Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. B. (2017). Julia: A fresh approach to numerical computing. *SIAM Review*, *59*(1), 65–98.

Broch, O. J., & Slagstad, D. (2012). Modelling seasonal growth and composition of the kelp saccharina latissima. *Journal of Applied Phycology*, *24*, 759–776. https://doi.org/10.1007/s10811-011-9695-y

Danisch, S., & Krumbiegel, J. (2021). Makie.jl: Flexible high-performance data visualization for julia. *Journal of Open Source Software*, *6*(65), 3349. https://doi.org/10.21105/joss.03349

Dunbar, O. R. a., Lopez-Gomez, I., Garbuno-Iñigo, A., Huang, D. Z., Bach, E., & Wu, J. (2022). EnsembleKalmanProcesses.jl: Derivative-free ensemble-based model calibration. *Journal of Open Source Software*, *7*(80), 4869. https://doi.org/10.21105/joss.04869

Kuhn, A. M., Fennel, K., & Mattern, J. P. (2015). Model investigations of the north atlantic spring bloom initiation. *Progress in Oceanography*, *138*, 176–193. https://doi.org/10.1016/j.pocean.2015.07.004

Lévy, M., Gavart, M., Mémery, L., Caniaux, G., & Paci, A. (2005). A four-dimensional mesoscale map of the spring bloom in the northeast atlantic (POMME experiment): Results of a prognostic model. *Journal of Geophysical Research C: Oceans*, *110*, 1–23. https://doi.org/10.1029/2004JC002588

Ramadhan, A., Wagner, G. L., Hill, C., Jean-Michel, C., Churavy, V., Souza, A., Edelman, A., Ferrari, R., & Marshall, J. (2020). Oceananigans.jl: Fast and friendly geophysical fluid dynamics on GPUs. *Journal of Open Source Software*, *5*(53), 2018. https://doi.org/10.21105/joss.02018

Soetaert, K., Middelburg, J. J., Herman, P. M. J., & Buis, K. (2000). On the coupling of benthic and pelagic biogeochemical models. In *Earth-Science Reviews* (Vol. 51, pp. 173–201). https://doi.org/10.1016/S0012-8252(00)00004-0

Wanninkhof, R. (1992). Relationship between wind speed and gas exchange over the ocean. *Journal of Geophysical Research*, *97*, 7373–7382. https://doi.org/10.1029/92JC00188