The Journal of Open Source Software

cgeniepy: A Python package for analysing cGENIE Earth System Model output

Rui Ying [1¶]

1 School of Earth Sciences, University of Bristol, UK ¶ Corresponding author

DOI: 10.21105/joss.06762

Software

- Review
- Repository
- Archive

Editor: Rachel Wegener

Reviewers:

- @koldunovn
- @jimmielin

Submitted: 22 April 2024
Published: 26 September 2024

Summary

cGENIE is a numerical model that simulates the Earth system (e.g., atmosphere, ocean, land, biosphere, ice sheet, and their interactions) in different geological ages (Ridgwell et al., 2007). It has been widely used in studying and reconstructing the past ocean and climate states. Here, I provide a Python package *cgeniepy* for reading, analysing, and visualising the cGENIE model output, and performing the model-data comparison. The package is designed to facilitate the post-simulation analysis for all cGENIE users, as used in my recent studies (Ying et al., 2023a; Ying et al., 2023b). The package is designed as object-oriented, thus many features are standalone and can be used without cGENIE background.

Statement of need

Earth System Models are the essential tool used to study the mechanisms regulating the complex climate and their impacts. cGENIE is such a model with intermediate model complexity that strengthens its application in paleoceanography studies. For instance, Henehan et al. (2019) used the model to study the impact of an extreme climatic event (Cretaceous-Paleogene massive extinction in 66 Million years ago). Pohl et al. (2022) used it to study the long-term evolution of ocean oxygen in the past 550 million years. The application of this model has promoted our understanding of climate change in the geological past.

Despite the power of cGENIE, the analysis of its model output has relied on a collection of MATLAB scripts developed by the cGENIE maintainer (https://github.com/derpycode/muffinplot). A systematic package has been long missing. Such gap might hamper the efficiency of the research, in particular for users who are not familiar with MATLAB or need to perform customised analysis (e.g., model ensemble based analysis).

Python is a popular open-source programming language that has a built-in package management system. Therefore, relative to MATLAB, Python packages can be easier to install, use, and demonstrate across platforms based on jupyter notebook/quarto. So far, many climate and ocean Models have their own Python package support (e.g., Caneill (2023) for the NEMO model and Forget (2023) for the MITgcm model). As such, it is useful to develop a similar one for the growing cGENIE community.

Package Design

This package first provides a class `model` to read the cGENIE model output (Figure 1). Then the accessed data will in be stored in corresponding data structure class (`GriddedData` or `ScatterData`). The two data structure classes are based on the `xarray.DataArray` and `pandas.DataFrame` respectively, which are common data structure used in the Python community. The two data structures can also be converted to each other easily using built-in methods.

Once the data classes are initialised (read from GENIE model or not), the users can perform basic operations as they do in `xarray.DataArray` and `pandas.DataFrame`. However, additional features are provided such as the publication-ready visualisation achived by the `GriddedDataVis` and `ScatterDataVis` class. Both contain various options to customise the plot based on the `matplotlib` and `cartopy` packages.

Another common demand for Earth system model users is the model-data comparison. Thus, I provided a `skill` module to conduct the skill score calculation including the correlation coefficient, root mean square error, and the Taylor diagram (see the Examples).

For cGENIE model specifically, its coarse model output can be interpolated using the `Interpolator` class (Figure 1). This is a wrapper of the `scipy.interpolate` subpackage and its purpose is to help increase the grid resolution and create prettier figures. However, a long-term goal is to incorporate more advanced interpolation methods (e.g., the DIVA method) to make cGENIE model output more comparable to the high-resolution model/observational results.

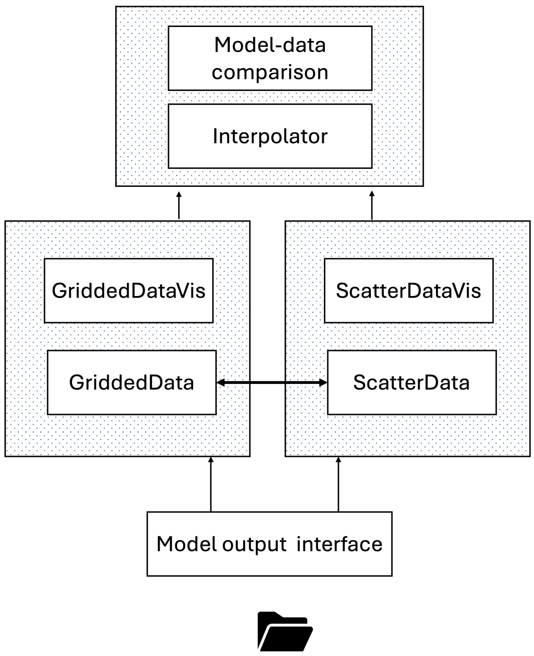

Figure 1: A schematic figure showing the structure of the `cgeniepy` package and its functionalities. `cgeniepy` helps users to access the model output and operate the visualisation and analysis, including interpolation and model-data comparison.

Examples

In this section, I provide two examples to show the core functionalities of `cgeniepy`. More examples however can be found in the package documentation website (https://cgeniepy.readthedocs.io/en/latest/).

Access, analyse and visualise the cGENIE model output

The following code example demonstrates using cgeniepy in a common use case for cGENIE users. It initialises the cGENIE model instance, read the sea surface temperature data, and plot the last time slice as map. The data is adapted from Ying et al. (2023b) and Gutjahr et al. (2017) (Figure 2). The users can easily change the variable name to access other model outputs.

```
## import the package
from cgeniepy.model import GenieModel

## initialise a model instance bu providing the path to the model output
model = GenieModel("/Users/foo/model_experiment_id")

## get the time-slice variable
ocn_sst = model.get_var("ocn_surf_temp")

## plot the last time slice
ocn_sst.isel(time=-1).plot()
```

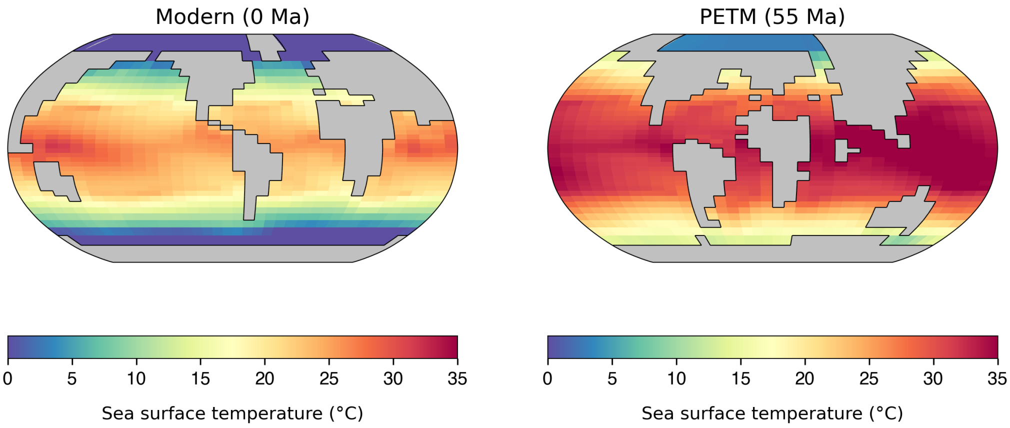

Figure 2: The simulated sea surface temperature in the Modern (left) and Paleogene-Eocene Thermal Maximum event (right) in cGENIE and visualised by cgeniepy. The data is adapted from Ying et al. (2023b) and Gutjahr et al. (2017) respectively.

Model-data comparison

In this example, I demonstrate how to conduct a model-data comparison using cgeniepy. I compare the ocean carbon isotope in sediment cores (i.e., observation) (Peterson et al., 2014) with cGENIE model ouptputs (Ying et al., 2023b) in the Last Glacial Maximum (21 ka). First, both observations and model results are read by the cgeniepy package. Then I search the nearest cGENIE model grid boxes for each sediment core and append the matched results to the existing dataframe. Finally, I visualise the model-data comparison by plotting the scatter plot with multiple metrics calculated (Figure 3).

```
from cgeniepy.model import GenieModel
from cgeniepy.table import ScatterData

## The example model and data are archived in
## https://zenodo.org/records/13786013 and
## https://zenodo.org/records/8189647

## initialise a model instance
```

```python
lgm_model = GenieModel("path/to/the/model/")
## get the variable and select the last time slice of the spin-up model
lgm_d13C = lgm_model.get_var("ocn_DIC_13C").isel(time=-1)

## read in the proxy data and construct ScatterData object
proxy_d13C = ScatterData("path/to/the/data")
proxy_d13C.set_index(["Lat", "Lon", "Depth"])

## find the model data for each core location
model_data = []
for i in proxy_d13C.data.index:
    lat, lon, depth = i
    pos = (depth, lat, lon)
    data = lgm_d13C.search_point(pos, ignore_na=True)
    model_data.append(data)

## add the model data to the dataframe
proxy_d13C.data["GENIE_d13C"] = model_data

## rename the column for y axis label
proxy_d13C.data.rename(columns={"LGM":"Observational d13C"}, inplace=True)

## conduct the model-data comparison and plot the 1:1 lineplot
## by default, model data is in the col, and observational col is in the second
proxy_d13C.compare("GENIE_d13C","LGM").plot()
```

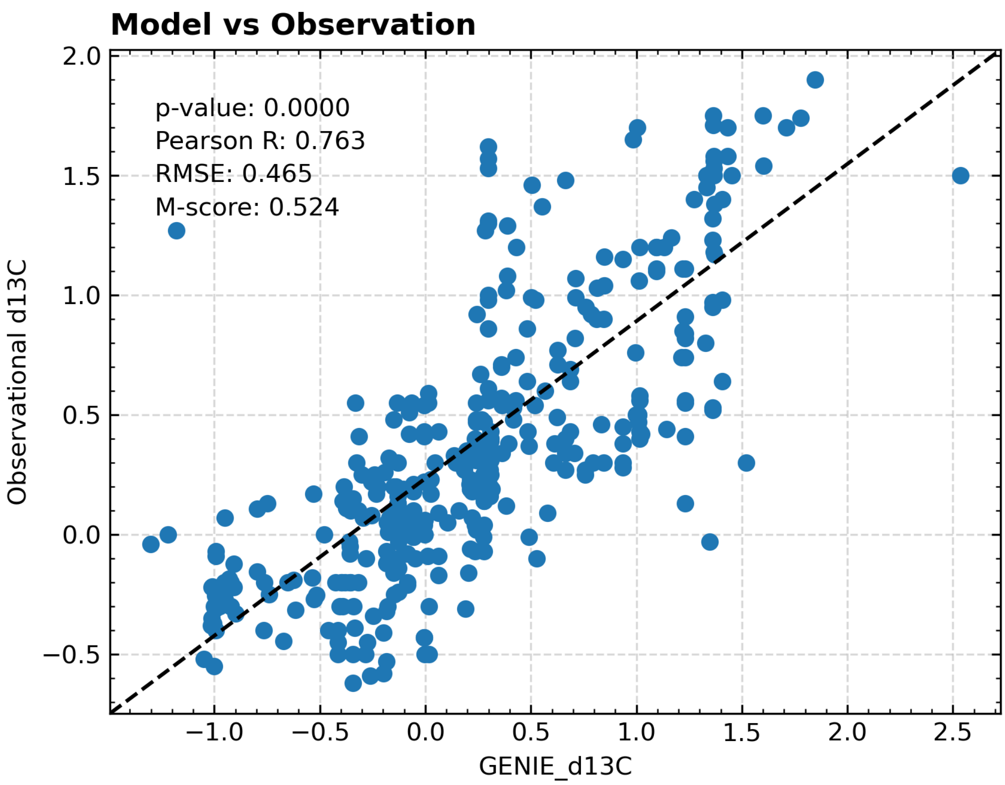

Figure 3: A cgeniepy example of model-data comparison for the Last Glacial Maximum carbon isotope data. The model output is adapted from Ying et al. (2023b) and the data is from Peterson et al. (2014).

Acknowledgements

R.Y. acknowledges financial support from China Scholarship Council (202006380070). R.Y. also thanks Shao Jun for his suggestions on the package.

References

Caneill, R. (2023). Interface to open NEMO ocean global circulation model output dataset and create a xgcm grid. In *GitHub repository*. GitHub. https://doi.org/10.5281/zenodo.5724577

Forget, G. (2023). Tools to interact with MITgcm (setup, run, output, plot, etc). In *GitHub repository*. GitHub. https://doi.org/10.5281/zenodo.13766391

Gutjahr, M., Ridgwell, A., Sexton, P. F., Anagnostou, E., Pearson, P. N., Pälike, H., Norris, R. D., Thomas, E., & Foster, G. L. (2017). Very large release of mostly volcanic carbon during the Palaeocene–Eocene Thermal Maximum. *Nature*, *548*(7669, 7669), 573–577. https://doi.org/https://doi.org/10.1038/nature23646

Henehan, M. J., Ridgwell, A., Thomas, E., Zhang, S., Alegret, L., Schmidt, D. N., Rae, J. W. B., Witts, J. D., Landman, N. H., Greene, S. E., Huber, B. T., Super, J. R., Planavsky, N. J., & Hull, P. M. (2019). Rapid ocean acidification and protracted Earth system recovery followed the end-Cretaceous Chicxulub impact. *Proceedings of the National Academy of Sciences*, *116*(45), 22500–22504. https://doi.org/https://doi.org/10.1073/pnas.1905989116

Peterson, C. D., Lisiecki, L. E., & Stern, J. V. (2014). Deglacial whole-ocean δ13C change estimated from 480 benthic foraminiferal records. *Paleoceanography*, *29*(6), 549–563. https://doi.org/10.1002/2013PA002552

Pohl, A., Ridgwell, A., Stockey, R. G., Thomazo, C., Keane, A., Vennin, E., & Scotese, C. R. (2022). Continental configuration controls ocean oxygenation during the Phanerozoic. *Nature*, *608*(7923), 523–527. https://doi.org/10.1038/s41586-022-05018-z

Ridgwell, A., Hargreaves, J. C., Edwards, N. R., Annan, J. D., Lenton, T. M., Marsh, R., Yool, A., & Watson, A. (2007). Marine geochemical data assimilation in an efficient Earth System Model of global biogeochemical cycling. *Biogeosciences*, *4*(1), 87–104. https://doi.org/https://doi.org/10.5194/bg-4-87-2007

Ying, R., Monteiro, F. M., Wilson, J. D., & Schmidt, D. N. (2023a). ForamEcoGEnIE 2.0: Incorporating symbiosis and spine traits into a trait-based global planktic foraminiferal model. *Geoscientific Model Development*, *16*(3), 813–832. https://doi.org/10.5194/gmd-16-813-2023

Ying, R., Monteiro, F., Wilson, J., & Schmidt, D. (2023b). *Marine zooplankton acclimated to geological warming while facing limits by the next century* [Preprint]. Climate. https://doi.org/10.31223/X5D10G