The Journal of Open Source Software

SICOPOLIS-AD v2: tangent linear and adjoint modeling framework for ice sheet modeling enabled by automatic differentiation tool Tapenade

Shreyas Sunil Gaikwad [1¶], **Laurent Hascoet** [2], **Sri Hari Krishna Narayanan** [3], **Liz Curry-Logan** [1], **Ralf Greve** [4,5], **and Patrick Heimbach** [1,6,7]

1 Oden Institute for Computational Engineering and Sciences, University of Texas at Austin, USA **2** Institut National de Recherche en Informatique et Automatique, France **3** Mathematics and Computer Science Division, Argonne National Laboratory, USA **4** Institute of Low Temperature Science, Hokkaido University, Japan **5** Arctic Research Center, Hokkaido University, Japan **6** Jackson School of Geosciences, University of Texas at Austin, USA **7** Institute for Geophysics, University of Texas at Austin, USA ¶ Corresponding author

DOI: 10.21105/joss.04679

Software

- Review
- Repository
- Archive

Editor: Chris Vernon

Reviewers:

- @svchb
- @ifthompson

Submitted: 27 July 2022
Published: 07 March 2023

Summary

SImulation COde for POLythermal Ice Sheets (`SICOPOLIS`) is an open-source, 3D dynamic/thermodynamic model that simulates the evolution of large ice sheets and ice caps. SICOPOLIS has been developed continuously and applied to problems of past, present, and future glaciation of Greenland, Antarctica, and others. It uses the finite differences discretization on a staggered Arakawa C grid and employs the shallow ice and shallow shelf approximations, making it suitable for paleoclimatic simulations. We present a new framework for generating derivative code, i.e., tangent linear, adjoint, or Hessian models, of SICOPOLIS. These derivative operators are powerful computational engines to efficiently compute comprehensive gradients or sensitivities of scalar-valued model output, including least-squares model-data misfits or important quantities of interest, to high-dimensional model inputs (such as model initial conditions, parameter fields, or boundary conditions). The new version 2 (`SICOPOLIS-AD v2`) framework is based on the source-to-source automatic differentiation (AD) tool Tapenade which has recently been open-sourced. The switch from a previous AD tool (`OpenAD`) used in SICOPOLIS-AD version 1 to Tapenade overcomes several limitations outlined here. The framework is integrated with the SICOPOLIS model's main trunk and is freely available.

Statement of need

The two contemporary ice sheets, Greenland and Antarctica, are dynamic entities whose evolution is governed by a set of nonlinear partial differential equations (PDEs) that describe the conservation of mass, momentum, and energy, as well as constitutive laws for the material properties of ice. In general, these equations cannot be solved analytically but must be solved numerically. Ice sheet models are a computer representation of these PDEs. They require as input parameters (i) initial conditions of the state of the ice sheet, (ii) surface boundary conditions, such as precipitation, (iii) basal boundary conditions, such as geothermal flux, and (iv) model parameters, such as flow law parameters. Despite advances in numerical modeling of ice sheets, the effects of ad-hoc initialization and the uncertainties in these independent input parameters propagate to quantities of interest (QoI), such as future projections of sea-level rise, which is of economic and societal importance (Schinko et al., 2020). It is thus desirable to evaluate the sensitivities of our QoI to these independent input variables.

In the context of ice sheet modeling, sensitivities of model-data misfits or other QoI are a key ingredient for performing model calibration, state estimation, or uncertainty quantification (UQ), which guide the improvement of model simulations through PDE-constrained gradient-based optimization.

`SICOPOLIS-AD v2` leverages the recently open-sourced AD tool `Tapenade` (Laurent Hascoët & Pascual, 2013) to generate code for the adjoint model of the open-source ice sheet model, `SICOPOLIS` (Greve, 1997; Greve et al., 2011; Greve & Blatter, 2009). Sensitivities can be calculated using a single forward and adjoint model evaluation, instead of the $\mathcal{O}(N)$ forward model evaluations. Empirically, one adjoint model evaluation is about 5-10 times as expensive as a forward model run. The adjoint computation is highly efficient for calculating sensitivities when N is large (typically, $N \sim 10^4 - 10^6$).

The functionality to generate a tangent linear version of the forward model is also included, which was not available in `SICOPOLIS-AD v1`. This is valuable for UQ of the inferred parameters, as well as uncertainty propagation to QoIs. It can also be used to verify the results of the adjoint model.

Target Audience

This package is intended as a resource that enables sensitivity analysis, model calibration, and uncertainty quantification of a continent-scale ice sheet model. Our package is also intended to serve as a guide for future work in the application of open-source AD tools for physics-based simulation codes written in `Fortran`.

State of the field

`SICOPOLIS` is among the early thermo-mechanical models to simulate contemporary and paleo continental-scale ice sheets (Greve, 1997). Like similar models developed at the time, including Glimmer and its successor, the Community Ice Sheet Model (`CISM`) (Rutt et al., 2009), `GRISLI` (Ritz et al., 1996), the model by Huybrechts (1990), or by Pollard & DeConto (2009), `SICOPOLIS` has been based (until recently) on the so-called shallow ice approximation to simplify the Cauchy stress tensor in the momentum conservation equation, implemented on a regular, finite-difference mesh. See Hindmarsh (2004) for other approximations commonly employed in ice sheet models. This approximation enabled the efficient computation of ice sheet evolution over long, glacial/deglacial cycles.

The last decade has seen substantial advances in continental-scale ice sheet modeling, with the development of several new ice sheet models (some of which are on unstructured grids using finite element methods), notably the Ice Sheet System Model `ISSM` (Larour et al., 2012), the Parallel Ice Sheet Model `PISM` (Bueler et al., 2007), Elmer/Ice (Gagliardini et al., 2013), or the MPAS-Albany Land Ice `MALI` (Hoffman et al., 2018). While designed to capture the evolution of short-term, fast-flowing, or fast-changing outlet glaciers via horizontal stress contributions, these models have so far found little application in paleo-ice sheet simulations due to their extensive computational costs. A compilation of the suite of ice sheet models used for the latest Ice Sheet Model Intercomparison Project, Phase 6 (`ISMIP6`) in support of the IPCC's Sixth Assessment Report is available in Payne et al. (2021) and Nowicki et al. (2016).

Relevant to this paper, of all the time-evolving models listed, apart from `SICOPOLIS-AD` (Heimbach & Bugnion, 2009; Logan et al., 2020), only the `ISSM` model and variants thereof possess adjoint model codes which have been generated, in part, using automatic differentiation (L. Hascoët & Morlighem, 2018; Larour et al., 2014). Multi-centennial and longer integrations with the adjoint model have so far been conducted only with `SICOPOLIS-AD`.

Features

AD tools such as the commercial TAF (Giering & Kaminski, 1999) and the open-source OpenAD (Utke et al., 2008) have been used previously with SICOPOLIS (Heimbach & Bugnion, 2009; Logan et al., 2020). OpenAD is no longer actively developed because it is based on the Open64 compiler which ceased development in 2011. The differentiation of SICOPOLIS, therefore, must be performed using a different tool. Compared to OpenAD, the Tapenade enabled implementation has the following advantages:

- It is up-to-date with the latest SICOPOLIS code.
- The AD tool Tapenade is open-source and actively maintained.
- A new tangent linear code generation capability is introduced.
- The AD-generated codes can accept NetCDF inputs.
- The external library LIS, its tangent linear code, and adjoint code are correctly incorporated which can improve the simulation of Antarctic ice shelves and Greenland outlet glaciers.
- Gitlab-CI, a Docker, and the pytest framework are leveraged for Continuous Integration (CI) to track changes in the trunk that "break" the AD-based code generation.
- The entire code is parsed by Tapenade, preventing cumbersome manual maintenance of subroutines to initialize the adjoint runs.
- Python scripts are provided for quick setup of the compilation, I/O, and execution processes based on user-provided metadata.
- The setup is well-documented, along with tutorials.

Software requirements and external usage

SICOPOLIS-AD v2 is built on top of the ice sheet model SICOPOLIS and uses Tapenade to differentiate this model. All the prerequisites of using SICOPOLIS and Tapenade need to be satisfied. A Python installation is needed to use the automation tools.

Example

We illustrate the use of our tool with the example of a steady-state simulation of the Greenland ice sheet under modern climate conditions. The corresponding SICOPOLIS configuration header file, v5_grl16_bm5_ss25ka, is provided as a reference template in the standard SICOPOLIS distribution. We shorten the total integration time to 100 simulated years to keep the computational cost of the tangent linear and finite differences reasonable. Our QoI (i.e., dependent variable) is the total volume of the ice sheet at the end of the run (fc). The sensitivity is evaluated with respect to the geothermal heat flux, q_geo (independent variable), a 19,186-dimensional field. The results are shown in Figure 1.

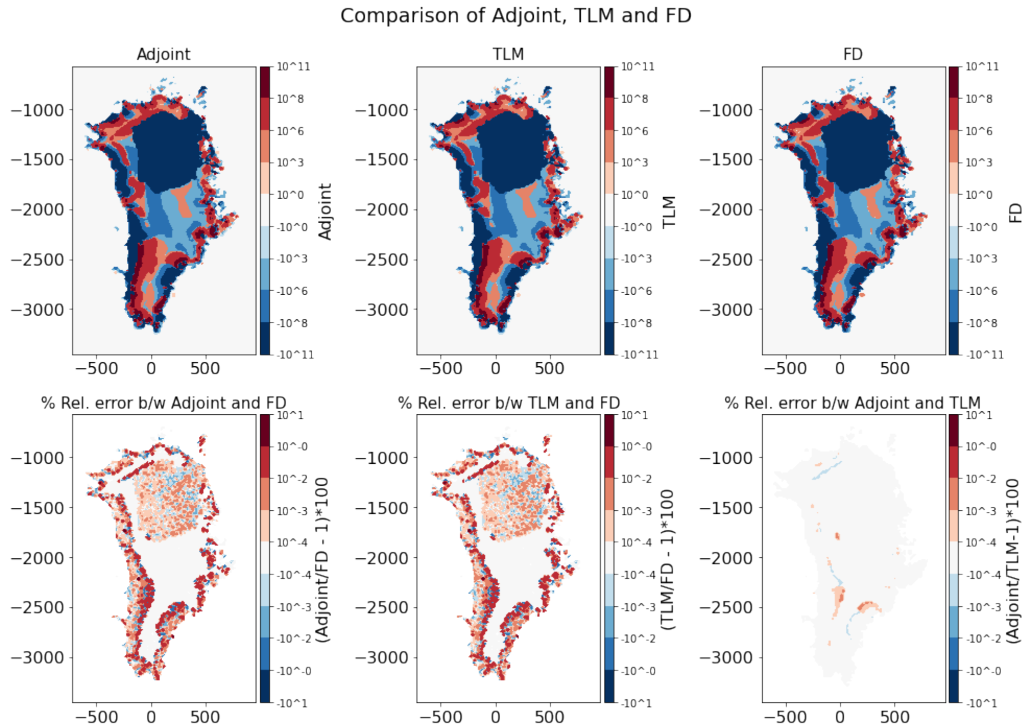

Figure 1: Validation exercise for adjoint (ADM) and tangent linear (TLM) models using the finite differences (FD) results for the sensitivity of `fc` with respect to `q_geo`. The upper row shows the sensitivities computed using the adjoint model (reverse-mode AD), tangent linear (forward-mode AD), and finite differences, respectively. The bottom run illustrates the relative error between (ADM, FD), (TLM, FD), and (ADM, TLM) respectively. For the bottom row, note that the values of relative error are only shown for points where the value of the gradient is "significant", i.e. within 4 orders of magnitude of the maximum absolute value of the gradient.

The results show good agreement between all three modes used to evaluate this sensitivity. The error is less than 6% between AD-generated (adjoint/tangent linear codes) and finite differences at all but one point with "significant" gradient values, i.e. within 4 orders of magnitude of the maximum absolute value of the finite differences gradient. The relative error between the AD-generated adjoint and tangent linear models is less than 0.002% at all points with values within 4 orders of magnitude of the maximum absolute value of the finite differences gradient. However, the adjoint model is much faster than the other two, as shown in Table 1, because the number of evaluations of the latter two scales linearly with the parameter dimension ($\sim\mathcal{O}(N)$). The discrepancy will be even larger if a finer mesh is used.

Table 1: Comparison of the time taken by various methods to evaluate the gradient for a scalar objective function with respect to a 19,186-dimensional 2D field (16 km mesh) in a typical SICOPOLIS run. The runs are performed on Intel Xeon CPU E5-2695 v3 nodes (2.30 GHz clock rate, 35.84 MB L3 cache, 63.3 GB memory).

Gradient calculation method	Time (in seconds) for 16 km mesh
Finite Differences	1.640×10^5
Tangent Linear Model	9.793×10^4
Adjoint Model	2.214×10^1

Acknowledgements

This work was supported by the Applied Mathematics activity within the U.S. Department of Energy, Office of Science, Advanced Scientific Computing Research Program, under contract number DE-AC02-06CH11357, by National Science Foundation OPP/P2C2 grant #1903596, by Japan Society for the Promotion of Science (JSPS) KAKENHI grant numbers JP17H06104 and JP17H06323, and by the Japanese Ministry of Education, Culture, Sports, Science and Technology (MEXT), Arctic Challenge for Sustainability project ArCS II (program grant number JPMXD1420318865).

References

Bueler, E., Brown, J., & Lingle, C. (2007). Exact solutions to the thermomechanically coupled shallow-ice approximation: Effective tools for verification. *Journal of Glaciology*, *53*(182), 499–516. https://doi.org/10.3189/002214307783258396

Gagliardini, O., Zwinger, T., Gillet-Chaulet, F., Durand, G., Favier, L., Fleurian, B. de, Greve, R., Malinen, M., Martín, C., Råback, P., Ruokolainen, J., Sacchettini, M., Schäfer, M., Seddik, H., & Thies, J. (2013). Capabilities and performance of Elmer/Ice, a new-generation ice sheet model. *Geoscientific Model Development*, *6*(4), 1299–1318. https://doi.org/10.5194/gmd-6-1299-2013

Giering, R., & Kaminski, T. (1999). Recipes for adjoint code construction. *ACM Transactions on Mathematical Software*, *24*(4), 437–474. https://doi.org/10.1145/293686.293695

Greve, R. (1997). Application of a polythermal three-dimensional ice sheet model to the Greenland ice sheet: Response to steady-state and transient climate scenarios. *Journal of Climate*, *10*(5), 901–918. https://doi.org/10.1175/1520-0442(1997)010%3C0901:AOAPTD%3E2.0.CO;2

Greve, R., & Blatter, H. (2009). *Dynamics of ice sheets and glaciers*. Springer. https://doi.org/10.1007/978-3-642-03415-2

Greve, R., Saito, F., & Abe-Ouchi, A. (2011). Initial results of the SeaRISE numerical experiments with the models SICOPOLIS and IcIES for the Greenland ice sheet. *Annals of Glaciology*, *52*(58), 23–30. https://doi.org/10.3189/172756411797252068

Hascoët, L., & Morlighem, M. (2018). Source-to-source adjoint Algorithmic Differentiation of an ice sheet model written in C. *Optimization Methods and Software*, *33*(4-6), 829–843. https://doi.org/10.1080/10556788.2017.1396600

Hascoët, Laurent, & Pascual, V. (2013). The Tapenade Automatic Differentiation tool: principles, model, and specification. *ACM Transactions on Mathematical Software*, *39*(3). https://doi.org/10.1145/2450153.2450158

Heimbach, P., & Bugnion, V. (2009). Greenland ice-sheet volume sensitivity to basal, surface and initial conditions derived from an adjoint model. *Annals of Glaciology*, *50*(52), 67–80. https://doi.org/10.3189/172756409789624256

Hindmarsh, R. C. A. (2004). A numerical comparison of approximations to the Stokes equations used in ice sheet and glacier modeling. *Journal of Geophysical Research: Earth Surface*, *109*(F1), F01012. https://doi.org/10.1029/2003JF000065

Hoffman, M. J., Perego, M., Price, S. F., Lipscomb, W. H., Zhang, T., Jacobsen, D., Tezaur, I., Salinger, A. G., Tuminaro, R., & Bertagna, L. (2018). MPAS-Albany Land Ice (MALI): A variable-resolution ice sheet model for Earth system modeling using Voronoi grids. *Geoscientific Model Development*, *11*(9), 3747–3780. https://doi.org/10.5194/gmd-11-3747-2018

Huybrechts, P. (1990). A 3-D model for the Antarctic ice sheet: A sensitivity study on the glacial-interglacial contrast. *Climate Dynamics*, *5*(2), 79–92. https://doi.org/10.1007/BF00207423

Larour, E., Seroussi, H., Morlighem, M., & Rignot, E. (2012). Continental scale, high order, high spatial resolution, ice sheet modeling using the Ice Sheet System Model (ISSM). *Journal of Geophysical Research: Earth Surface*, *117*(F1), F01022. https://doi.org/10.1029/2011JF002140

Larour, E., Utke, J., Csatho, B., Schenk, A., Seroussi, H., Morlighem, M., Rignot, E., Schlegel, N., & Khazendar, A. (2014). Inferred basal friction and surface mass balance of the Northeast Greenland Ice Stream using data assimilation of ICESat (Ice Cloud and land Elevation Satellite) surface altimetry and ISSM (Ice Sheet System Model). *The Cryosphere*, *8*(6), 2335–2351. https://doi.org/10.5194/tc-8-2335-2014

Logan, L. C., Narayanan, S. H. K., Greve, R., & Heimbach, P. (2020). SICOPOLIS-AD v1: An open-source adjoint modeling framework for ice sheet simulation enabled by the algorithmic differentiation tool OpenAD. *Geoscientific Model Development*, *13*(4), 1845–1864. https://doi.org/10.5194/gmd-13-1845-2020

Nowicki, S. M. J., Payne, A., Larour, E., Seroussi, H., Goelzer, H., Lipscomb, W., Gregory, J., Abe-Ouchi, A., & Shepherd, A. (2016). Ice Sheet Model Intercomparison Project (ISMIP6) contribution to CMIP6. *Geoscientific Model Development*, *9*(12), 4521–4545. https://doi.org/10.5194/gmd-9-4521-2016

Payne, A. J., Nowicki, S., Abe-Ouchi, A., Agosta, C., Alexander, P., Albrecht, T., Asay-Davis, X., Aschwanden, A., Barthel, A., Bracegirdle, T. J., Calov, R., Chambers, C., Choi, Y., Cullather, R., Cuzzone, J., Dumas, C., Edwards, T. L., Felikson, D., Fettweis, X., … Zwinger, T. (2021). Future sea level change under Coupled Model Intercomparison Project Phase 5 and Phase 6 scenarios from the Greenland and Antarctic ice sheets. *Geophysical Research Letters*, *48*(16), e2020GL091741. https://doi.org/10.1029/2020GL091741

Pollard, D., & DeConto, R. (2009). Modelling West Antarctic ice sheet growth and collapse through the past five million years. *Nature*, *458*(7236), 329–332. https://doi.org/10.1038/nature07809

Ritz, C., Fabre, A., & Letréguilly, A. (1996). Sensitivity of a Greenland ice sheet model to ice flow and ablation parameters: Consequences for the evolution through the last climatic cycle. *Climate Dynamics*, *13*(1), 11–23. https://doi.org/10.1007/s003820050149

Rutt, I. C., Hagdorn, M., Hulton, N. R. J., & Payne, A. J. (2009). The Glimmer community ice sheet model. *Journal of Geophysical Research: Earth Surface*, *114*(F2), F02004. https://doi.org/10.1029/2008JF001015

Schinko, T., Drouet, L., Vrontisi, Z., Hof, A., Hinkel, J., Mochizuki, J., Bosetti, V., Fragkiadakis, K., Vuuren, D. van, & Lincke, D. (2020). Economy-wide effects of coastal flooding due to sea level rise: A multi-model simultaneous treatment of mitigation, adaptation, and residual impacts. *Environmental Research Communications*, *2*(1), 015002. https://doi.org/10.1088/2515-7620/ab6368

Utke, J., Naumann, U., Fagan, M., Tallent, N., Strout, M., Heimbach, P., Hill, C., & Wunsch, C. (2008). OpenAD/F: A modular open-source tool for automatic differentiation of Fortran codes. *ACM Transactions on Mathematical Software*, *34*(4), 18:1–18:36. https://doi.org/10.1145/1377596.1377598