Geophysical Research Letters

Supplementary material for: Bathymetric influences on Antarctic ice-shelf melt rates

Authors: D. N. Goldberg, T. A. Smith, S. H. K. Narayanan, and M. Morlighem

Contents

1 Modifications to the MITgcm adjoint

The MITgcm, and in particular a configuration using the `SHELFICE` physics package for an Antarctic ice shelf, has been differentiated algorithmically[4], and so no additional modifications were required for applications to ice sheet-ocean interactions. However, there are technical issues in using bathymetry as a control variable. For instance, fluid fractions at grid cell faces (see Section ?? of main text) are based on the minimum fraction of adjacent cells, leading to potential non-differentiability. We adopt the approach of[5] of "smoothing" the min/max functions, but we note that this feature has not been used outside of bathymetric sensitivity studies.

Another computational challenge in treating bathymetry as a control variable lies with the implicit solve for the free surface at each time step[6]. The model solves the linear system $\mathbf{A}\eta = \mathbf{b}$ for η, where η is the free surface at the next time step, and $\mathbf{b}$ is a field arising from the baroclinic step of the model. $\mathbf{A}$ is a linear, self-adjoint operator on η and the propagation of sensitivity from η to b can be calculated analytically:

$$\delta^*\mathbf{b} = \mathbf{A}^{-1}\delta^*\eta, \tag{1}$$

where $\delta^*\eta$ is the *adjoint sensitivity* of η and likewise for $\mathbf{b}$. This formulation is standard in the MITgcm for adjoint based sensitivity analyses of any control variable except for fluid depth. However, the operator $\mathbf{A}$ depends on ocean column depth, which in the present study is a control variable, and therefore the backward-propagation of sensitivities from η to $\mathbf{A}$ must be considered as well.[5] dealt with this issue by allowing the AD tool to differentiate the linear solver code; however, as it is an iterative solver, this approach requires storing intermediate variables at each solver iteration during every time step of the forward model, which hinders performance and does not scale well to high dimensional problems.[5] recommend, but do not implement, using the approach of[2], which augments Eqn. (1) with

$$\delta^* \mathbf{A} = -\delta^* \mathbf{b}\ \eta^T. \tag{2}$$

In this work we implement this approach, obviating the need for the AD tool to differentiate the implicit solver.

2 Resilient Adjoints

Simulation of large models requires the use of high performance computing (HPC), generally with defined job time limits. For instance, standard batches on the ARCHER supercomputer have a walltime limit of 24 hours (there is a special queue for jobs that take up to 48 hours, but there are fewer resources available and generally longer wait times for this queue). Additionally, imposed time limits aside, longer computational jobs increase the risk of network or server errors leading to crashes. The MITgcm has a restart capability allowing to circumvent these limits: the "state" of the model is periodically saved to file, and new jobs can begin from this time stamp by reading the saved state. To restart the adjoint model, simulations must save both the forward and adjoint states – a capability referred to as *resilient adjoints*. A similar capability was previously implemented with TAF as *the Divided Adjoint* (DIVA).

Here we provide an overview of resilient adjoints, a strategy that enhances the default checkpointing scheme used by OpenAD. Checkpointing approaches store the state of the primal (forward) computation and reduce the amount of memory that is required to compute adjoints. By default, OpenAD uses binomial checkpointing for the time-stepping loop[3]. Consider a computation consisting of l timesteps, with c the number of checkpoints that can be stored. Figure S1 (top) illustrates binomial checkpointing for $l = 10$ and $c = 3$.

A two-level checkpointing approach can build upon this approach by converting the time stepping loop into a loop nest containing l_2 outer iterations and l_1 inner iterations where $l = l_2 \times l_1$[1]. The inner loop uses binomial checkpointing as before; the outer loop uses periodic checkpointing. The left part of Figure S1 (bottom) illustrates two level checkpointing for $l_2 = 5$, $l_1 = 10$ and $c_1 = 3$. The resilient adjoints capability enhances two level checkpointing by storing to disk the adjoint state computed at the end of each outer level iteration. To restart a computation at the granularity of an l_2 timestep then, only the stored l_2 state checkpoints and the last adjoint checkpoint, if any, are required.

References

[1] Aupy, G., J. Herrmann, P. Hovland, and Y. Robert, Optimal multistage algorithm for adjoint computation, *SIAM Journal on Scientific Computing*, *38*, doi:10.1137/15M1019222, 2014.

[2] Giles, M. B., G. Corliss, C. Faure, A. Griewank, L. Hascoët, and U. Naumann, *On the Iterative Solution of Adjoint Equations*, pp. 145–151, Springer New York, New York, NY, doi:10.1007/978-1-4613-0075-5-16, 2002.

[3] Griewank, A., and A. Walther, Algorithm 799: Revolve: An implementation of checkpointing for the reverse or adjoint mode of computational differentiation, *ACM Trans. Math. Softw.*, *26*(1), 19–45, doi:10.1145/347837.347846, 2000.

[4] Heimbach, P., and M. Losch, Adjoint sensitivities of sub-ice shelf melt rates to ocean circulation under Pine Island Ice Shelf, West Antarctica, *Annals of Glaciol.*, *54*, 59–69, doi:10.3189/2012/AoG60A025, 2012.

[5] Losch, M., and P. Heimbach, Adjoint sensitivity of an ocean general circulation model to bottom topography, *Journal of Physical Oceanography*, *37*(2), 377–393, doi:10.1175/JPO3017.1, 2007.

[6] Marshall, J., C. Hill, L. Perelman, and A. Adcroft, Hydrostatic, quasi-hydrostatic, and nonhydrostatic ocean modeling, *Journal of Geophysical Research: Oceans*, *102*(C3), 5733–5752, doi: 10.1029/96JC02776, 1997.

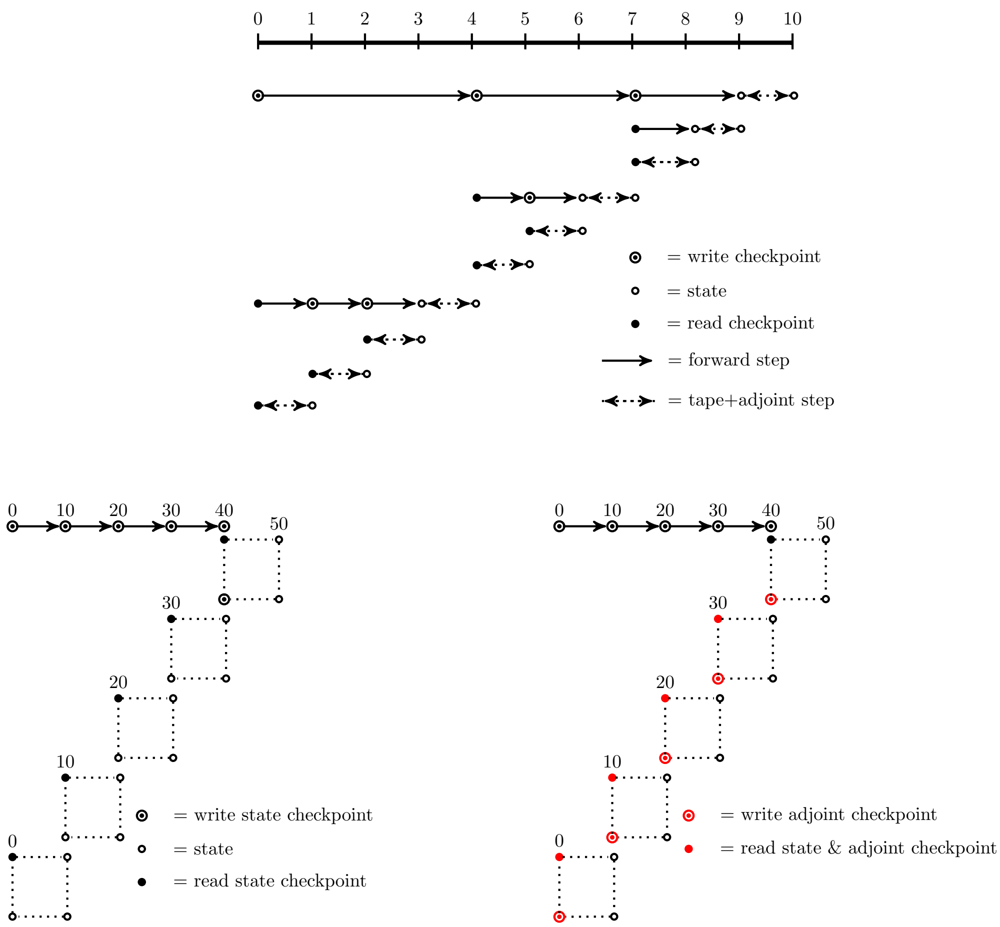

Figure S1: Top: Binomial checkpointing schedule for $l = 10$ time steps and $c = 3$ checkpoints. Bottom Left: Two level checkpointing schedule for $l = 50$ with ($l_2 = 5$) outer level iterations and ($l_1 = 10$) inner level iterations. Periodic checkpointing is used in the outer level and binomial checkpointing shown by the dashed box is used at the inner level. Bottom Right: Enhanced two level checkpointing schedule with support for resilient adjoints through the writing and reading of the adjoint state at the outer level.

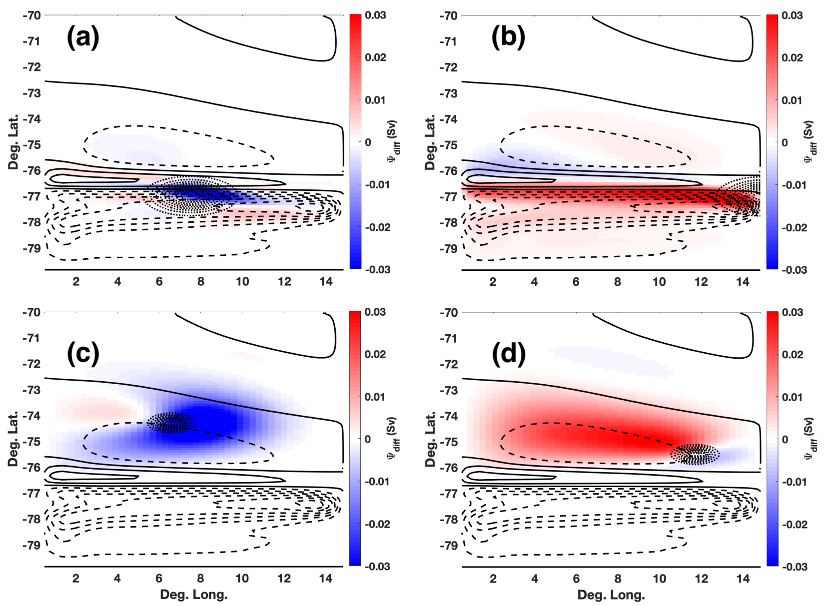

Figure S2: Perturbed beds (dotted contours) and corresponding perturbed barotropic stream functions (shading) in different regions of high sensitivity in Fig. 3 of the main text. (a) through (d) correspond to finite perturbations in locations (1) through (4) in Fig. 3(a) of the main text, respectively. Bathymetric peturbations plotted with δR=10 (Eqn. 3 of the main text) and 1m isolines. Isolines of unperturbed stream functions are also shown (solid where positive, dashed where negative; .05 Sv spacing).