Configuring time-steps and generating stable initial conditions for new grids in CAM-SE

Adam R. Herrington

National Center for Atmospheric Research, Boulder, Colorado, USA

December 9, 2020

Introduction

This document contains instructions for setting the appropriate time-steps and generating stable initial conditions for a new grid in CAM-SE. The first part of this document provides recommendations for the many time-steps in CAM. The second part explains how to generate stable initial conditions for use with the recommended time-steps. Throughout this document, *red fonts* refer to xml variables, *blue fonts* refer to namelist variables.

Time-stepping

CAM has lots of time-steps. The logical place to start is in defining the physics time-step. The physics time-step is the longest time-step in the model[1], and all other time-steps are defined through splitting the physics time-step by some integer.

Physics time-step

The physics time-step is controlled by the xml variable *ATM_NCPL*. This variable defines the number of times per model day to call the physics package,

$$\Delta t_{phys} = \frac{secpday}{ATM\_NCPL}$$

In CESM, *ATM_NCPL* also sets the frequency of coupling with the other model components.

There are no rigorous arguments for what the physics time-step should be in GCMs. Because the physics is expensive, the guiding philosophy seems to prioritize minimizing its cost, favoring large physics time-steps. GCMs usually set it to 1800 s (30 min) or thereabouts for standard 1° grids, and this is the default physics time-step in CAM.

[1]Technically, radiation is the longest time-step in CAM. Radiation tendencies are computed every hour, but are applied each physics time-step.

Table 1: Recommended physics, macrophysics/microphysics & CLUBB time-steps for different grids.

Grid Resolution	ATM_NCPL	Δt_{phys}(s)	*cld_macmic_num_steps*	Δt_{macmic}(s)	*nadv*	Δt_{clubb}(s)
$ne30x2$	96	900	3	300	1	300
$ne30x4$	192	450	2	225	1	225
$ne30x8$	384	225	1	225	1	225
$ne30x16$	384	225	1	225	1	225

Time-step convergence studies (Williamson and Olson, 2003; Herrington and Reed, 2018; Wan et al., 2020) indicate that a 1800 s time-step has large absolute time-truncation errors. What this means is that if you shorten the physics time-step, the solutions will change in a significant way. In particular, the model becomes less cloudy, which reduces SWCF, and increases stratiform precipitation rates and reduces convective precipitation rates. If this new model state impedes the science, then the user should consider re-tuning the model.

Herrington and Reed (2020) provide a physical justification for scaling the physics time-step in proportion to the grid spacing. This scaling is designed to avoid increasing time-truncation errors with resolution that damp the resolved vertical velocity field. Table 1 provides the recommendations from that study. The $ne30x16$ refinement recommendation departs from the scaling with resolution. This grid resolution is in the grey zone, and so it may be advantageous to invite some damping of the vertical velocities from time-truncation errors, which would act to compensate for the lack of non-hydrostatic pressure gradients in a hydrostatic model.

Macrophysics/microphysics & CLUBB time-steps

The physics package contains a number of subcycles. The user should keep track of the macrophysics/microphysics and CLUBB time-steps, as both can contribute substantially to the model cost. The macrophysics/microphysics time-step is found through splitting the physics time-step by the namelist variable *cld_macmic_num_steps*,

$$\Delta t_{macmic} = \frac{\Delta t_{phys}}{cld\_macmic\_num\_steps}.$$

The default value of *cld_macmic_num_steps* = 3, and it is recommended that this value be lowered for most refined grids (Table 1). The MG microphysics can be ran with a smaller time-step through dividing the macrophysics/microphysics time-step by the integer *micro_mg_num_steps*. By default this value is set to 1, and no further subcycling of the microphysics occurs.

CLUBB is advanced as a subcylcle in the macrophysics/microphysics loop. If the macrophysics/microphysics time-step is set properly, the CLUBB time-step is computed internally to the recommended value and no additional namelist variables need to be specified (Table 1).

The CLUBB time-step is defined by dividing the macrophyiscs/microphysics time-step by the integer *nadv*,

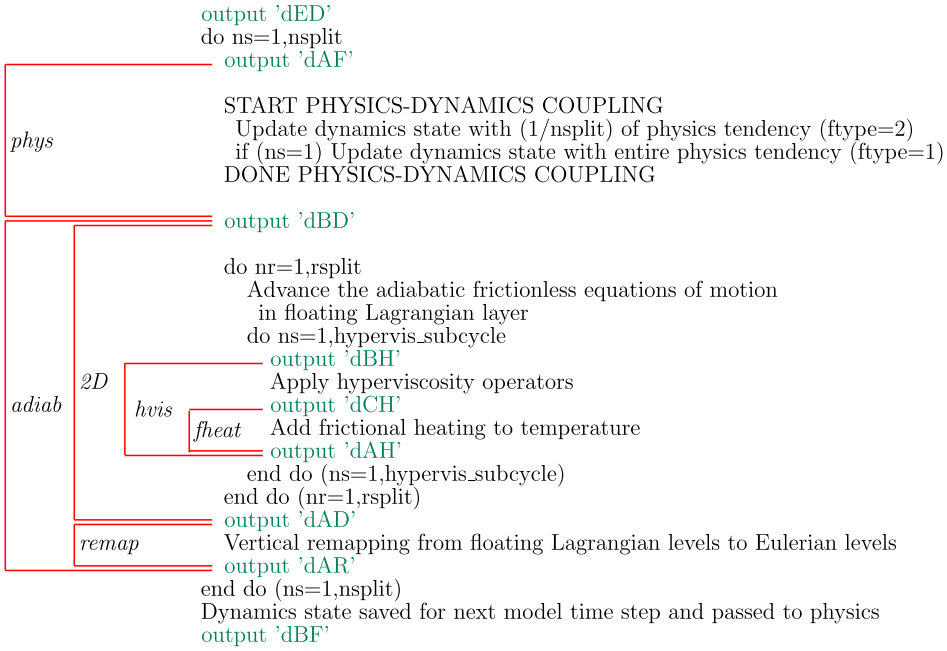

Figure 1: Schematic of the spectral-element time-step loops after Lauritzen and Williamson (2019).

$$\Delta t_{clubb} = \frac{\Delta t_{macmic}}{nadv}.$$

The namelist variable *clubb_timestep* defaults to 300 s (5 min). This does not reflect the actual CLUBB time-step, but rather provides an upper bound. Internally, the CLUBB time-step is set to the smallest value of $nadv$ such $\Delta t_{clubb} \leq$ *clubb_timestep*. At present, the actual clubb time-step is not written to any of the logs. An issue was recently opened on the CAM trunk for it to write the clubb time-step to the log files (`https://github.com/ESCOMP/CAM/issues/278`).

Spectral-element dynamical core time-steps

The spectral-element dynamical core has many time-steps. The structure of its nested loops are depicted schematically in Figure 1 after Lauritzen and Williamson (2019). All spectral-element specific namelist variables begin with the three character modifier "*se_*" which has been omitted from the variables names in this section for brevity.

The outer-loop divides the physics time-step by *nsplit*. At the very top of this loop the physics tendencies are applied, and at the very end of this loop vertical remapping is performed. The *rsplit* loop is nested within the *nsplit* loop and advances the frictionless adiabatic equations of motion in 2D floating Lagrangian layers. Tracer advection is also computed in this loop, but could optionally be subcycled further *qsplit* number of times. Within the *rsplit* loop, the hyper-viscosity operators are sub-cycled *hypervis_subcycle* number of times, which applies hyper-viscous damping of the state, and then converts the associated dissipation of kinetic energy to frictional heating. A

Table 2: Recommended dycore time-steps for different grids. Entries with an asterisk denote default values used in the model.

Grid Resolution	Δt_{phys}(s)	*nsplit*	Δt_{remap}(s)	*rsplit*	Δt_{tracer}(s)	*qsplit*	Δt_{dyn}(s)
$ne30x2$	900	2*	450	3*	150	1*	150
$ne30x4$	450	2*	225	3*	75	1*	75
$ne30x8$	225	2*	112.5	3*	37.5	1*	37.5
$ne30x16$	225	4	56.25	3*	18.75	1*	18.75

separate loop within the *rsplit* loop applies Laplacian damping of the sponge layer, subcycling *hypervis_subcycle_sponge* number of times. The sponge layer is defined as the top three levels of the model in standard low top configurations of CAM.

The various dycore time-steps are defined as,

$$\Delta t_{remap} = \frac{\Delta t_{phys}}{nsplit}$$

$$\Delta t_{tracer} = \frac{\Delta t_{phys}}{rsplit \times nsplit}$$

$$\Delta t_{dyn} = \frac{\Delta t_{phys}}{qsplit \times rsplit \times nsplit}$$

$$\Delta t_{hypervis} = \frac{\Delta t_{phys}}{hypervis\_subcycle \times qsplit \times rsplit \times nsplit}$$

$$\Delta t_{hypervis\_q} = \frac{\Delta t_{phys}}{hypervis\_subcycle\_q \times qsplit \times rsplit \times nsplit}$$

$$\Delta t_{hypervis\_sponge} = \frac{\Delta t_{phys}}{hypervis\_subcycle\_sponge \times qsplit \times rsplit \times nsplit}$$

The user is reminded that the dycore time-steps are a function of the physics time-step, and so changing *ATM_NCPL* impacts all other time-steps. Since the recommendation is to scale the physics time-steps in proportion to grid spacing, and the dynamical core time-steps need to scale with the grid-spacing for stability, then the default dycore *splits* gives stable time-steps for most grid resolutions. The recommended values for the dycore time-steps and namelists variables are provided in Table 2. Entries with an asterisk denote default values and hence don't need to be specified in user_nl_cam.

Users would be wise to check the *atm.log.** from a run with the new grid as it contains information on the stability of time steps. Search for "dt" in the log to find a list of CFL calculations for the different dycore time-steps, which is shown in Figure 2 for a $ne30x8$ grid. Even though the *split* variables are those recommended in Table 2, the CFL calculations indicate that $\Delta t_{dyn} = \Delta t_{tracer} = 37.5$ s is unstable (green circles in Figure 2). It is the experience of the author that the CFL calculations for these two time-steps are too strict. As long as the model time-steps aren't too different (e.g., within a factor of 1.5), users may ignore these CFL warnings as long as they adhere to the time-steps recommended in Table 2. That being said, if the CFL limited time-step is larger by an order of magnitude, the user should

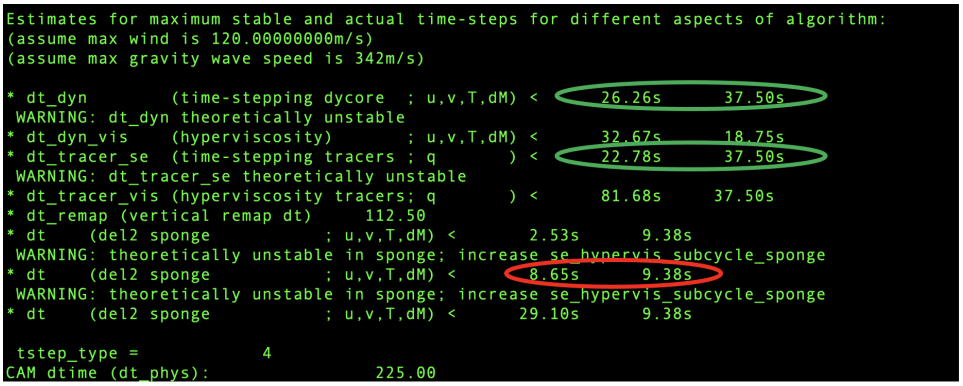

Figure 2: Snapshot of the portion of the *atm.log.** containing the CFL stability calculations.

The CFL calculations are perhaps more useful for informing the stability of the hyper-viscosity time-steps, at least as it relates to variable-resolution grids. When the namelist $se\_refined\_mesh = .true.$, which it needs to be for variable-resolution grids, the hyper-viscosity subcycles are automatically set to[2],

$$hypervis\_subcycle = 2$$
$$hypervis\_subcycle\_q = 1$$
$$hypervis\_subcycle\_sponge = 4$$

The CFL calculations indicates that $\Delta t_{hypervis}$ & $\Delta t_{hypervis\_q}$ are both stable, but that the hyper-viscosity time-step in the top two layers of the sponge-layer is unstable. In the experience of the author the top sponge layer CFL calculation is too strict and can be ignored. However, *it is necessary to satisfy the CFL criterion of the second sponge layer* (red circle in Figure 2). If one increases $hypervis\_subcycle\_sponge = 5$, then $\Delta t_{hypervis\_sponge} = 7.5$ s, which is less than the CFL limiting value of 8.65 s in that layer. The user needs to specify $hypervis\_subcycle\_sponge = 5$ in the namelists for this configuration to be stable.

Generating stable initial conditions

Generating initial conditions that are stable for the recommended time-steps can be a monotonous process. In the author's experience, this has been made easier through the implementation of the US standard atmosphere initial condition option.

First guess at initial conditions

The US standard atmosphere initial conditions are invoked through a combination of xml and namelist variables,

[2]These default hypervis_subcycle_xxx values are subject to change in the near future, and are only listed here as an example.

$$./xmlchange\ CAM\_CONFIG\_OPTS = \ '\ –analytic\_ic' – –append$$
$$analytic\_ic\_type = \ 'us\_standard\_atmosphere'$$
$$use\_topo\_file = .true.$$

This will set the initial conditions to the US standard atmosphere profile computed using the native topography file specified by the *bnd_topo* namelist.

Alternatively, one can interpolate from an existing initial condition file from a different grid, which is often unstable due to the differences in topography/surface pressure between the two grids. An interpolated file can be generated using the Community Mesh Generation Toolkit. Once generated, set the full path of the new initial condition file to the *ncdata* namelist. It should be noted that for CAM-chem simulations, an initial condition file needs to be specified even when using the US standard atmosphere analytical initial condition option, since CAM-chem needs more variables than are available from the US standard atmosphere.

Lastly, one can use Colin Zarzycki's Betacast software to generate initial conditions. Betacast includes a tool that generates a CAM initial condition file from a reanalysis snapshot. Specifically, it regrids the reanalysis to the target grid, and then performs a hydrostatic adjustment to the surface pressure field using the reanalysis and target grid topography. The author has not experimented with this approach, but the use of a hydrostatic correction is desirable in that it may alleviate the need for a spinup altogether.

However the user chooses to proceed with a first guess at initial conditions, a stable initial condition file needs to be generated. This requires dumping out new initial condition files during the spinup. To do this, set the namelist $inithist = \ 'DAILY'$ to generate new initial condition files at the end of each simulated day.

To determine if a spinup is required, run the new grid for 7 days using the recommended time-steps. If the model completes, run another 7 days and take the last **.cam.i.** file (day 14) and set it to the initial conditions via the *ncdata* namelist. If the model errors out, the initial conditions are likely unstable and a spinup is required.

The spinup procedure

A spinup in this context refers to an iterative process in which the user cranks up the hyper-viscosity coefficients, while also decreasing the time-step, and then *gradually* relaxes back to the recommended settings. I will reiterate my earlier statement, that the spinup can be a monotonous process. The user may be tempted to give up due to difficulties stabilizing the model. The key is to not shock the model by relaxing back to the recommended parameters too fast. The spinup can require an enormous amount of patience, and is not recommended for new users.

Stage 1: Increased hyper-viscosity & decreased time-step

When $refined\_mesh = \ 'true'$, the model switches on the scale-aware tensor hyper-viscosity formulation of Guba et al. (2014). To ramp-up the viscosity coefficients, first grab the

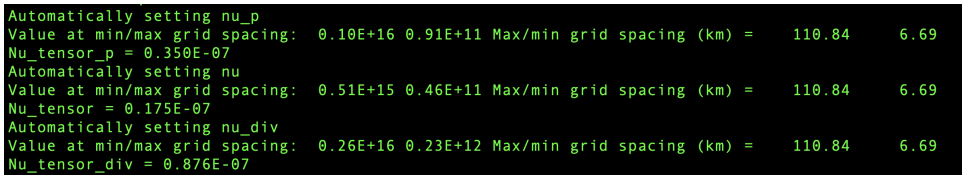

Figure 3: Snapshot of the portion of the *atm.log.** containing the default tensor viscosity coefficients.

default coefficients used in a failed run. In the *atm.log.** just prior to the time-step stability section, it will output a series of coefficients *Nu_tensor*, *Nu_tensor_p* & *Nu_tensor_div* (Figure 3). Increase these viscosity coefficients by a factor V by setting the namelists,

$$\begin{aligned} se\_nu &= V \times Nu\_tensor \\ se\_nu\_p &= V \times Nu\_tensor\_p \\ se\_nu\_div &= V \times Nu\_tensor\_div \end{aligned}$$

For extremely unstable conditions, set $V = 10$. This may be excessive depending on the circumstance, and so the user may want to begin with a smaller factor, e.g., $V = 5$.

To reduce the dynamics time-step, the user should crank up the vertical remapping *se_nsplit* namelist. Recall that *se_nsplit* $= 2$ by default, and the user needs to increase this by some factor N. Lets assume start with $N = 4$, although this may be excessive, and the user is free to experiment with a smaller value, e.g., $N = 2$.

Since the hyper-viscosity time-steps are a function of *se_nsplit*, the user should adjust the hyper-viscosity time-steps. These time-steps also need to be lowered due to the larger viscosity coefficients. Using the factors V & N defined above, the namelists should be adjusted as follows,

$$\begin{aligned} se\_hypervis\_subcycle &= V/N \times se\_hypervis\_subcycle \\ se\_hypervis\_subcycle\_q &= V/N \times se\_hypervis\_subcycle\_q \\ se\_hypervis\_subcycle\_sponge &= 1/N \times se\_hypervis\_subcycle\_sponge \end{aligned}$$

Note the number of subcycles are integers, and so round up the multiplicative factor as necessary. The user should run with these namelist settings for at least 7 days. It is possible that the hyper-viscosity time-steps are still not stable despite these changes in subcycling, and so the user should verify that the time-steps are stable by checking the *atm.log.**.

The user should familiarize themselves with the *cesm.log.**. If the run is not stable, then this log provides hints as to why the model is failing. In particular, when the dycore is unstable it will often print an error NEGATIVE LAYER THICKNESS in the *cesm.log.**. However, *this diagnostic does not print everytime the dycore is unstable*, and instead an obscure error may get triggered down-stream of the dycore instability that has nothing to do with the what is causing the model to error out in the first place. Therefore, the user needs to study the time-step diagnostics in the *atm.log.** to be certain that the dycore

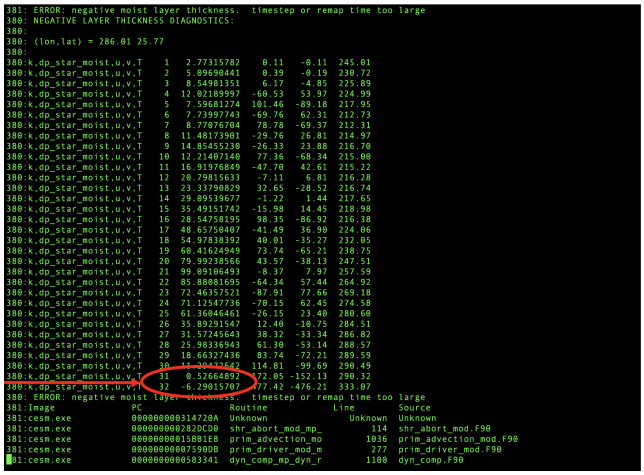

Figure 4: Snapshot of the portion of the *cesm.log.** containing the NEGATIVE LAYER THICKNESS diagnostics.

time-steps are stable.

Figure 4 shows the portion of the *cesm.log.** containing the NEGATIVE LAYER THICKNESS diagnostics. What is apparent from the log file is that the pressure thickness is negative at the lowest model level. If the negative pressure thickness had occurred in the sponge-layer, that would indicate the hyper-viscosity time-step in the sponge layer is unstable, and needs to be reduced via increasing *se_hypervis_subcycle_sponge*. Since the negative layer thickness is occurring at the lowest model level, this probably reflects unstable initial conditons, and so the user needs to either increase the hyper-viscosity coefficients or decrease the dynamics time-step via factors V & N, respectively.

Stage 2: Standard hyper-viscosity & decreased time-step

After successfully completing at least 7 simulated days with increased hyper-viscosity and reduced time-step, the user should begin relaxing the hyper-viscosity coefficients back to the recommended values. Begin by setting the *ncdata* namelist to the final **.cam.i.** file from the successful run in the prior step. Recall these initial condition files are generated so long as the user sets $inithist = 'DAILY'$. In roughly 7 day increments, slowly relax the hyper-viscosity coefficient factor V down to $V = 1$, subsequently updating *ncdata* to the last **.cam.i.** file from the prior run. Each time the user reduces the factor V, the values of *se_hypervis_subcycle* & *se_hypervis_subcycle_q* should be lowered as well.

Stage 3: Standard hyper-viscosity & standard time-step

Once the user is able to successfully run with hyper-viscosity factor $V = 1$, then the final step is to relax the dycore time-steps to their recommended values. Analogous to the prior step, slowly relax the dycore time-step factor N back to $N = 1$ in 7 day increments, subsequently updating the *ncdata* namelist to the final **cam.i.** of the prior run. As all three hyper-viscosity time-steps are a function of N, the corresponding hyper-viscosity subcycles should be adjusted accordingly. The final product is then the last **cam.i.** file from a stable $N = 1$ run. This file contains initial conditions that are in balance with the new grid, and should be cherished.

References

O. Guba, M. A. Taylor, P. A. Ullrich, J. R. Overfelt, and M. N. Levy. The spectral element method (sem) on variable-resolution grids: evaluating grid sensitivity and resolution-aware numerical viscosity. *Geosci. Model Dev.*, 7(6):2803–2816, 2014. doi: 10.5194/gmd-7-2803-2014.

A. R. Herrington and K. A. Reed. An idealized test of the response of the community atmosphere model to near-grid-scale forcing across hydrostatic resolutions. *J. Adv. Model. Earth Syst.*, 10(2):560–575, 2018. doi: 10.1002/2017MS001078.

A. R. Herrington and K. A. Reed. On resolution sensitivity in the community atmosphere model. *Quarterly Journal of the Royal Meteorological Society*, 2020. doi: 10.1002/qj.3873.

P. H. Lauritzen and D. L. Williamson. A total energy error analysis of dynamical cores and physics-dynamics coupling in the community atmosphere model (cam). *J. Adv. Model. Earth Syst.*, 2019. doi: 10.1029/2018MS001549.

H. Wan, C. S. Woodward, S. Zhang, C. J. Vogl, P. Stinis, D. J. Gardner, P. J. Rasch, X. Zeng, V. E. Larson, and B. Singh. Improving time step convergence in an atmosphere model with simplified physics: the impacts of closure assumption and process coupling. *Journal of Advances in Modeling Earth Systems*, 12(10):e2019MS001982, 2020.

D. L. Williamson and J. G. Olson. Dependence of aqua-planet simulations on time step. *Q. J. R. Meteorol. Soc.*, 129(591):2049–2064, 2003. doi: 10.1256/qj.02.62.