Namelist Guide

A. Entries in fv_core_nml

A.1. Required options:

layout Integer(2): Processor layout on each tile. The number of PEs assigned to a domain must equal layout(1)*layout(2)*ntiles. Must be set.

npx Integer: Number of grid *corners* in the x-direction on one tile of the domain; so one more than the number of grid cells across a tile. On the cubed sphere this is *one more than* the number of cells across a cube face. Must be set.

npy Integer: Number of grid *corners* in the y-direction on one tile of the domain. This value should be identical to npx on a cubed-sphere grid; doubly periodic or nested grids do not have this restriction. Must be set.

npz Integer: Number of vertical levels. Each choice of npz comes with a pre-defined set of hybrid sigma-pressure levels and model top (see fv_eta.F90). Must be set.

ntiles Integer: Number of tiles on the domain. For the cubed sphere, this should be 6, one tile for each face of the cubed sphere; normally for most other domains (including nested grids) this should be set to 1. Must be set.

A.2. Initialization options:

add_noise Real: amplitude of random thermal noise (in K) to add upon startup. Useful for perturbing initial conditions. -1 by default; disabled if 0 or negative.

adjust_dry_mass Logical: whether to adjust the global dry-air mass to the value set by dry_mass. This is only done in an initialization step, particularly when using an initial condition from an external dataset, interpolated from another resolution (either horizontal or vertical), or when changing the topography, so that the global mass of the atmosphere matches some estimate of observed value. False by default. It is recommended to only set this to True when initializing the model.

breed_vortex_inline Logical: whether to bogus tropical cyclones into the model, which are specified by an external file. Options are set in fv_nwp_nudge_nml. False by default.

dry_mass Real: if adjust_dry_mass is true, sets the global dry air mass, measured in the globally-averaged surface pressure (Pascals) by adding or removing mass from the lowest layer of the atmosphere as needed. 98290. (Pa) by default.

external_ic Logical: Whether to initialize the models state using the data in an externally specified file, given in res_latlon_dynamics. By default this file is assumed to be a legacy lat-lon FV core restart file; set either ncep_ic or fv_diag_ic to true override this behavior..false. by default. Note that external_ic = true will cause the model to re-initialize the dynamical fields from the input dataset regardless of whether warm_start is set.

full_zs_filter Logical: whether to apply the on-line topography filter during initialization. Only active if get_nggps_ic = .true. This is so topography filtering can be performed on the initial conditions output by the pre-processing tools, which currently do not support topography filtering for some configurations (such as the nested grid); this also allows the user to easily test changes to the topography filtering on the simulation. Note that for all other initialization methods (if external_ic = .true.) the on-line topography filter will be applied automatically during the initialization of the topography. .false. by default.

mountain Logical: takes topography into account when initializing the model. Set this to true to apply the terrain filter (if n_zs_filter = 2 or 4) upon startup; also set to True when cold starting so that the topography can be initialized. Only set this to false if you wish to cold-start without any topography; this value is ignored for the aquaplanet test_case = 14. True by default. It is highly recommended to not alter this value unless you know what you are doing.

na_init Integer: Number of forward-backward dynamics steps used to initialize adiabatic solver. This is useful for spinning up the nonhydrostatic state from the hydrostatic GFS analyses. 0 by default. Recommended to set this to a non-zero value (1 or 2 is typically sufficient) when initializing from GFS or ECMWF analyses.

ncep_ic Logical: If external_ic =.true., this variable says whether the file in res_latlon_dynamics is an NCEP analysis or reanalysis file. This option zeros out all tracer fields except specific humidity..false. by default.

nggps_ic Logical: If external_ic =.true., reads initial conditions from horizontally-interpolated output from chgres. False by default. Additional options are available through external_ic_nml.

ecmwf_ic Logical: If external_ic =.true., reads initial conditions from ECMWF analyses. .false. by default.

external_eta Logical: If .true., reads the interface coefficients a_k and b_k from either the restart file (if restarting) or from the external initial condition file (if nggps_ic or ecwmf_ic are .true.). This overrides the hard-coded levels in fv_eta. .false. by default.

nord_zs_filter Integer: order of the topography filter applied to n_zs_filter. Set to 2 to get a second-order filter, or 4 to get a fourth-order filter; other values do no filtering. 0 by default. This should not be set to a non-zero value on multiple successive simulations; the filter is applied every time the model restarts. This option is useful for testing the terrain filter, and should not be used for regular runs.

npz_rst Integer: If using a restart file with a different number of vertical levels, set npz_rst to be the number of levels in your restart file. The model will then remap the restart file data to the vertical coordinates specified by npz. 0 by default; if 0 or equal to npz no remapping is done.

nudge Logical: whether to use the nudging towards the state in some externally-supplied file (such as from reanalysis or another simulation). Further nudging options are set in fv_nwp_nudge_nml. False by default.

nudge_dz Logical: during the adiabatic initialization (na_init ¿ 0), if set to .true. delz is nudged back to the value specified in the initial conditions, instead of nudging the temperature back to the initial value. Nudging delz is simpler (faster), doesn't require consideration of the virtual temperature effect, and may be more stable. .false. by default.

nudge_ic Logical: same as nudge, but works in adiabatic solo_core simulations to nudge the field to a single external analysis file. False by default.

nudge_qv Logical: during the adiabatic initialization (na_init ¿ 0), if set to .true., the water vapor profile is nudged to an analytic fit to the HALOE climatology. This is to improve the water vapor concentrations in GFS initial conditions, especially in the stratosphere, where values can be several times higher than observed. This nudging is unnecessary for other ICs, especially the ECMWF initial conditions. .false. by default.

n_zs_filter Integer: number of times to apply a diffusive filter to the topography upon startup, if mountain is True and the model is not being cold-started. This is applied every time the model is warm-started, so if you want to smooth the topography make sure this is set to 0 after the first simulation. 0 by default. If initializing the model from cold-start the topography is already being filtered by an amount appropriate for the model resolution.

res_latlon_dynamics character(len=128) If external_ic =.true. gives the filename of the input IC file. INPUT/fv_rst.res.nc by default.

res_latlon_tracers character(len=128) If external_ic =.true. and both ncep_ic and fv_diag_ic are.false., this variable gives the filename of the initial conditions for the tracers, assumed to be a legacy lat-lon FV core restart file. INPUT/atmos_tracers.res.nc by default.

warm_start Logical; whether to start from restart files, instead of cold-starting the model. True by default; if this is set to true and restart files cannot be found the model will stop.

A.3. I/O and diagnostic options:

agrid_vel_rst Logical: whether to write the unstaggered latitude-longitude winds (u_a and v_a) to the restart files. This is useful for data assimilation cycling systems which do not handle staggered winds. .false. by default.

check_negative Logical: whether to print the most negative global value of microphysical tracers.

fv_debug Logical: whether to turn on additional diagnostics in fv_dynamics..false. by default.

fv_land Logical: whether to create terrain deviation and land fraction for output to mg_drag restart files, for use in mg_drag and in the land model..false. by default;.true. is recommended when, and only when, initializing the model, since the mg_drag files created provide a much more accurate terrain representation for the mountain gravity wave drag parameterization and for the land surface roughness than either computes internally. This has no effect on the representation of the terrain in the dynamics.

io_layout Integer(2): Layout of output files on each tile. 1,1 by default, which combines all restart and history files on a tile into one file. For 0,0, every process writes out its own restart and history files. If not equal to 1,1, you will have to use mppnccombine to combine these output files prior to post-processing, or if you want to change the number of PEs. Both entries must divide the respective value in layout.

nf_omega Integer: number of times to apply second-order smoothing to the diagnosed omega. When 0 the filter is disabled. 1 by default.

print_freq Integer: number of hours between print out of max/min and air/tracer mass diagnostics to standard output. 0 by default, which never prints out any output; set to -1 to see output after every dt_atmos. Computing these diagnostics requires some computational overhead.

range_warn Logical: checks whether the values of the prognostic variables are within a reasonable range at the end of a dynamics time step, and prints a warning if not. False by default; adds computational overhead, so we only recommend using this when debugging.

write_3d_diags Logical: whether to write out three-dimensional dynamical diagnostic fields (those defined in fv_diagnostics.F90). This is useful for runs with multiple grids if you only want very large 3D diagnostics written out for (say) a nested grid, and not for the global grid. False by default.

A.4. Options controlling tracers and interactions with physics

adiabatic Logical: whether to skip any physics. If true, the physics is not called at all and there is no virtual temperature effect. False by default; this option has no effect if not running solo_core.

do_Held_Suarez Logical: whether to use Held-Suarez forcing. Requires adiabatic to be false. False by default; this option has no effect if not running solo_core.

do_uni_zfull Logical: whether to compute z_full (the height of each model layer, as opposed to z_half, the height of each model interface) as the midpoint of the layer, as is done for the nonhydrostatic solver, instead of the height of the location where $p = \bar{p}$ the mean pressure in the layer. This option is not available for fvGFS or the solo_core. .false. by default.

dnats Integer: The number of tracers which are not to be advected by the dynamical core, but still passed into the dynamical core; the last dnats+pnats tracers in field_table are not advected. 0 by default.

dwind_2d Logical: whether to use a simpler & faster algorithm for interpolating the A-grid (cell-centered) wind tendencies computed from the physics to the D-grid. Typically, the A-grid wind tendencies are first converted in 3D cartesian coordinates and then interpolated before converting back to 2D local coordinates. When this option enabled, a much simpler but less accurate 2D interpolation is used. False by default.

fill Logical: Fills in negative tracer values by taking positive tracers from the cells above and below. This option is useful when the physical parameterizations produced negatives. False by default.

inline_q Logical: whether to compute tracer transport in-line with the rest of the dynamics instead of sub-cycling, so that tracer transport is done at the same time and on the same time step as is p and potential temperature. False by default; if true, q_split and z_tracer are ignored.

ncnst Integer: Number of tracer species advected by fv_tracer in the dynamical core. Typically this is set automatically by reading in values from field_table, but ncnst can be set to a smaller value so only the first ncnst tracers listed in field_table are not advected. 0 by default, which will use the value from field_table.

nwat Integer: Number of water species to be included in condensate and water vapor loading. The masses of the first nwat tracer species will be added to the dry air mass, so that p is the mass of dry air, water vapor, and the included condensate species. The value used depends on the microphysics in the physics package you are using. For GFS physics with only a single condensate species, set to 2. For schemes with prognostic cloud water and cloud ice, such as GFDL AM2/AM3/AM4 Rotsteyn-Klein or Morrison-Gettlean microphysics, set to 3. For warm-rain (Kessler) microphysics set to 4 (with an inactive ice tracer), which only handles three species but uses 4 to avoid interference with the R-K physics. For schemes such as WSM5 or Ferrier that have prognostic rain and snow but not hail, set to 5 (not yet implemented). For six-category schemes that also have prognostic hail or graupel, such as the GFDL, Thompson, or WSM6 microphysics, set to 6. A value of 0 turns off condensate loading. 3 by default.

phys_hydrostatic Logical: Option to enable hydrostatic application of heating from the physics in a nonhydrostatic simulation: heating is applied in hydrostatic balance, causing the entire atmospheric column to expand instantaneously. If false, heating from the physics is applied simply as a temperature tendency. True by default; ignored if hydrostatic =.true.

pnats Integer: The number of tracers not to advect by the dynamical core. Unlike dnats, these tracers are not seen by the dynamical core. The last pnats entries in field_table are not advected. 0 by default.

tau_h2o Real: time-scale (days) for simple methane chemistry to act as a source of water in the stratosphere. Can be useful if your stratosphere dries out too quickly; consider a value between 60 and 120 days if this is the case. 0. by default, which disables the methane chemistry. Values less than zero apply the chemistry above 100 mb; else applied above 30 mb. Requires adiabatic to be false.

use_hydro_pressure Logical: whether to compute hydrostatic pressure for input to the physics. Currently only enabled for the fvGFS model. Ignored in hydrostatic simulations. False by default.

z_tracer Logical: whether to transport sub-cycled tracers layer-by-layer, each with its own computed sub-cycling time step (if q_split = 0). This may improve efficiency for very large numbers of tracers. False by default; currently not implemented.

A.5. Timestep options

k_split Integer: number of vertical remappings per dt_atmos (physics time step). 1 by default.

n_split Integer: number of small dynamics (acoustic) time steps between vertical remapping. 0 by default, in which case the model produces a good first guess by examining the resolution, dt_atmos, and k_split.

umax Real: for the doubly-periodic grid (grid_type = 4) an estimate of the maximum wave speed (m/s), used to determine the value of n_split when n_split = 0. 350 by default.

q_split Integer: number of time steps for sub-cycled tracer advection. 0 by default (recommended), in which case the model determines the number of time steps from the global maximum wind speed at each call to the tracer advection.

A.6. Grid options

deglat Real: Latitude (in degrees) used to compute the uniform f-plane Coriolis parameter for doubly-periodic simulations (grid_type = 4). 15. by default.

do_schmidt Logical: Whether to enable grid stretching and rotation using stretch_fac, target_lat, and target_lon..false. by default.

dx_const Real: on a doubly-periodic grid (grid_type = 4) specifies the (uniform) grid-cell-width in the x-direction, in meters. 1000 by default.

dy_const Real: on a doubly-periodic grid (grid_type = 4) specifies the (uniform) grid-cell-width in the y-direction, in meters. 1000 by default.

grid_type Integer: which type of grid to use. If 0, the equidistant gnomonic cubed-sphere will be used. If 4, a doubly-periodic f-plane cartesian grid will be used. If -1, the grid is read from INPUT/grid_spec.nc. Values 2, 3, 5, 6, and 7 are not supported and will likely not run. 0 by default.

hybrid_z Logical: whether to use a hybrid-height coordinate, instead of the usual sigma-p coordinate. False by default. (Not currently maintained.)

make_hybrid_z Logical: Converts the vertical coordinate to a hybrid-height coordinate, instead of the usual sigma-p coordinate. Requires hybrid_z = True. False by default.

p_ref Real: surface pressure used to construct a horizontally-uniform reference vertical pressure profile, used in some simple physics packages in the solo_core and in the Rayleigh damping. *This should not be confused with the actual, horizontally-varying pressure levels used for all other dynamical calculations.* 1.e5 by default. *Changing this value is strongly discouraged.*

shift_fac Real: westward zonal rotation (or shift) of cubed-sphere grid from its natural orientation with cube face centers at 0, 90, 180, and 270 degrees longitude. The shift, in degrees, is 180/shift_fac. This shift does not move the poles. By default this is set to 18, shifting the grid westward 180/18=10 degrees, so that the edges of the cube do not run through the mountains of Japan; all standard CM2.x, AM3, CM3, and HiRAM simulations use this orientation of the grid. Requires do_schmidt =.false.

stretch_fac Real: stretching factor for the Schmidt transformation. This is the factor by which tile 6 of the cubed sphere will be shrunk, with the grid size shrinking accordingly. 1 by default, which performs no grid stretching. Requires do_schmidt =.true. The model will crash if stretch_fac is set to zero. Values of up to 40 have been found useful and stable for short-term cloud-scale integrations.

target_lat Real: latitude (in degrees) to which the center of tile 6 will be rotated; if stretching is done with stretch_fac the center of the high-resolution part of the grid will be at this latitude. -90 by default, which does no grid rotation (the Schmidt transformation rotates the south pole to the appropriate target). Requires do_schmidt =.true.

target_lon Real: longitude to which the center of tile 6 will be rotated. 0 by default. Requires do_schmidt =.true.

nested Logical: whether this is a nested grid. False by default.

twowaynest Logical: whether to use two-way nesting, the process by which the nested-grid solution can feed back onto the coarse-grid solution. False by default.

parent_grid_num Integer: Number of the parent to this nested grid. The coarsest grid in a simulation is numbered 1; further nested grids are numbered sequentially. Required to be a positive value if nested = True. Unless you are nesting inside of nested grids or running multiple (coarse) grids that themselves do not interact, this should be set to 1. -1 by default, indicating that this grid does not have a parent grid.

parent_tile Integer: number of the tile (ie. face) in which this nested grid is found in its parent. Required to be a positive value if nested = true. If the parent grid is not a cubed sphere, or itself is a nested grid, this should be set to 1. If the parent grid has been rotated (using do_schmidt) with the intent of centering the nested grid at target_lat and target_lon, then parent_tile should be set to 6. 1 by default.

refinement Integer: refinement ratio of the nested grid. This is the number of times that each coarse-grid cell face will be divided into smaller segments on the nested grid. Required to be a positive integer if nested = true. Nested grids are aligned with the coarse grid, so non-integer refinements are not permitted. 3 by default.

nestupdate Integer: type of nested-grid update to use; details are given in model/fv_nesting.F90. 0 by default.

A.7. Solver options

a2b_ord Integer: order of interpolation used by the pressure gradient force to interpolate cell-centered (A-grid) values to the grid corners. 4 by default (recommended), which uses fourth-order interpolation; otherwise second-order interpolation is used.

beta Real: Parameter specifying fraction of time-off-centering for backwards evaluation of the pressure gradient force. 0.0 by default, which produces a fully backwards evaluationthe pressure gradient force is entirely evaluated using the updated (time n+1) dynamical fields. A value of 0.5 will equally weight the PGF determined at times n and n+1, but may not be stable; values larger than 0.45 are not recommended. A value of 0.4 is recommended for most hydrostatic simulations, which allows an improved representation of inertia-gravity waves in the tropics. In non-hydrostatic simulations using the semi-implicit solver (a_imp > 0.5) the values of a_imp and beta should add to 1, so that the time-centering is consistent between the PGF and the nonhydrostatic solver. Proper range is 0 to 0.45.

c2l_ord Integer: order of interpolation from the solvers native D-grid winds to latitude-longitude A-grid winds, which are used as input to the physics routines and for writing to history files.

4 by default (recommended); fourth-order interpolation is used unless c2l_ord = 2.

consv_am Logical: whether to enable Angular momentum fixer. False by default.

consv_te Real: fraction of total energy lost during the adiabatic integration between calls of the physics, to be added back globally as heat; essentially the strength of the energy fixer in the physics. Note that this is a global energy fixer and cannot add back energy locally. The default algorithm increments the potential temperature so the pressure gradients are unchanged. 0 by default. Proper range is 0 to 1. 1 will restore the energy completely to its original value before entering the physics; a value of 0.7 *roughly* causes the energy fixer to compensate for the amount of energy changed by the physics in GFDL HiRAM or AM3.

convert_ke Logical: If true, adds energy dissipated through mechanical damping to heat throughout the *entire* depth of the domain; if false (default) this is only done in the sponge layer at the top of the domain. This option is only enabled if d_con > 1.e-5.

d_con Real: Fraction of kinetic energy lost to explicit damping to be converted to heat. Acts as a dissipative heating mechanism in the dynamical core. 0. by default. Proper range is 0 to 1. Note that this is a local, physically correct, energy fixer.

delt_max Real: maximum allowed magnitude of the dissipative heating rate, K s^{-1}; larger magnitudes are clipped to this amount. This can help avoid instability that can occur due to strong heating when d_con > 0. A value of 0.008 (a rate equivalent to about 800 K/day) is sufficient to stabilize the model at 3-km resolution. Set to 1 by default, which effectively disables this limitation.

fill_dp Logical: like fill except for p, the hydrostatic pressure thickness. When the filling occurs a diagnostic message is printed out, which is helpful for diagnosing where the problem may be occurring. Typically if the pressure filling is needed a crash is inevitable, and thus this option is often better for debugging than as a safety valve. False by default.

fv_sg_adj Integer: timescale (in seconds) at which to remove two-delta-z instability when the local (between two adjacent levels) Richardson number is less than 1. This is achieved by local mixing, which conserves mass, momentum, and total energy. Values of 0 or smaller disable this feature. If n_sponge < 0 then the mixing is applied only to the top n_sponge layers of the domain. Set to -1 (inactive) by default. Proper range is 0 to 3600.

halo_update_type Integer: which scheme to use for halo updates in multiprocessor simulations. If set to 1 non-blocking updates are used, which can improve simulation efficiency on some systems. Otherwise, blocking halo updates are performed. 1 by default.

hord_mt Integer: horizontal advection scheme for momentum fluxes. A complete list of kord options is given in the table below. 9 by default, which uses the third-order piecewise-parabolic method with the monotonicity constraint of Huynh, which is less diffusive than other constraints. For hydrostatic simulation, 8 (the L04 monotonicity constraint) is recommended;

for nonhydrostatic simulation, the completely unlimited ("linear" or non-monotone) PPM scheme is recommended. If no monotonicity constraint is applied, enabling the flux damping (do_vort_damp = .true.) is highly recommended to control grid-scale noise. It is also recommended that hord_mt, hord_vt, hord_tm, and hord_dp use the same value, to ensure consistent transport of all dynamical fields, unless a positivity constraint on mass advection (hord_dp) is desired.

hord_vt Integer: horizontal advection scheme for absolute vorticity and for vertical velocity in nonhydrostatic simulations. 9 by default.

hord_tm Integer: horizontal advection scheme for potential temperature and layer thickness in nonhydrostatic simulations. 9 by default.

hord_dp Integer: horizontal advection scheme for mass. A positivity constraint may be warranted for hord_dp but not strictly necessary. 9 by default.

hord_tr Integer: horizontal advection scheme for tracers. 12 by default. This value can differ from the other hord options since tracers are sub-cycled (if inline_q == False) and require positive-definite advection to control the appearance of non-physical negative masses. 8 (fastest) or 10 (least diffusive) are typically recommended.

hord	Horizontal Advection Method
5	Fastest unlimited fifth-order scheme with built-in $2\Delta x$ filter; not recommended for hord_tr. This is also the most accurate and least diffusive FV scheme available within FV^3 if monotonicity preservation is not a high priority.
6	*Developmental* PPM scheme with an intermediate-strength monotonicity constraint. More diffusive than 5.
7	6, applying a $2\Delta x$ filter and a positivity constraint
8	PPM with the constraint of Lin 2004
9	PPM with the Hunyh constraint
10	9, with a $2\Delta x$ filter, and the Huynh constraint applied only if a certain condition is met; otherwise unlimited

kord_mt Integer: vertical remapping scheme for the winds. 8 by default; 9 is recommended as the safest option, although 10, and 11 can also be useful. See table below for a complete list of kord options.

kord_tm Integer: vertical remapping scheme for temperature. If positive (not recommended), then vertical remapping is performed on total energy instead of temperature (see remap_t below). -8 by default.

kord_tr Integer: vertical remapping scheme for tracers. 8 by default. 9 or 11 recommended. It is often recommended to use the same value for kord_tr as for kord_tm.

kord_wz Integer: vertical remapping scheme for vertical velocity in nonhydrostatic simulations. 8 by default; 9 recommended. It is also recommended to use the same value for kord_wz as for kord_mt.

kord	Vertical remapping reconstruction method
4	Monotone PPM
6	PPM
7	PPM with Hyunh's second constraint (see L04)
9	Monotonic cubic spline with $2\Delta z$ oscillations removed
10	Selectively monotonic cubic spline, where local extrema are retained, with $2\Delta z$ oscillations removed
11	Non-monotonic (linear) cubic spline with $2\Delta z$ oscillations removed; if an invalid value for kord is given, this scheme is used
13	Monotonic cubic spline with $2\Delta z$ oscillations removed
16	Non-monotonic cubic spline with a strong $2\Delta z$ filter (similar to hord = 6).

no_dycore Logical: disables execution of the dynamical core, only running the initialization, diagnostic, and I/O routines, and any physics that may be enabled. Essentially turns the model into a column physics model. False by default.

remap_t Logical: whether the vertical remapping is performed on (virtual) temperature instead of (virtual) potential temperature. Since typically potential temperature increases exponentially from layer to layer near the top boundary, the cubic-spline interpolation in the vertical remapping will have difficulty with the exponential profile. Temperature does not have this problem and will often yield a more accurate result. True by default.

reproduce_sum Logical: uses an exactly-reproducible global sum operation performed when computing the global energy for consv_te. This is used because the FMS routine mpp_sum() is not bit-wise reproducible due to its handling of floating-point arithmetic, and so can return different answers for (say) different processor layouts. True by default.

A.8. Nonhydrostatic options

a_imp Real: Controls behavior of the non-hydrostatic solver. Values > 0.5 enable the semi-implicit solver, in which the value of a_imp controls the time-off-centering: use a_imp = 1.0 for a fully backward time-stepping. For consistency, the sum of beta and a_imp should be 1 when the semi-implicit solver is used. The semi-implicit algorithm is substantially more efficient except at very high (km-scale) resolutions with an acoustic time step of a few seconds or less. 0.75 by default. Proper values are 0, or between 0.5 and 1 Only used if hydrostatic =.false.

hydrostatic Logical: whether to use the hydrostatic or nonhydrostatic solver. True by default.

make_nh Logical: Whether to re-initialize the nonhydrostatic state, by re-computing dz from hydrostatic balance and setting w to 0. False by default.

p_fac Real: Safety factor for minimum nonhydrostatic pressures, which will be limited so the full pressure is no less than p_fac times the hydrostatic pressure. This is only of concern in mid-top or high-top models with very low pressures near the model top, and has no effect in most simulations. The pressure limiting activates only when model is in danger of blowup due to unphysical negative total pressures. 0.05 by default. Only used if hydrostatic =.false. and the semi-implicit solver is used. Proper range is 0 to 0.25.

use_logp Logical: Enables a variant of the Lin pressure-gradient force algorithm, which uses the logarithm of pressure instead of the Exner function (as in Lin 1997). This yields more accurate results for regions that are nearly isothermal. Ignored if hydrostatic = true. False by default.

A.9. Damping options

d2_bg Real: coefficient for background second-order divergence damping. This option remains active even if nord is nonzero. 0.0 by default. Proper range is 0 to 0.02.

d2_bg_k1 Real: strength of second-order diffusion in the top sponge layer. 0.16 by default. This value, and d2_bg_k2, will be changed appropriately in the model (depending on the height of model top), so the actual damping may be very reduced. See atmos_cubed_sphere/model/dyn_core.F90 for details. Recommended range is 0. to 0.2. Note that since diffusion is converted to heat if d_con > 0 larger amounts of sponge-layer diffusion may be *less* stable.

d2_bg_k2 Real: strength of second-order diffusion in the second sponge layer from the model top. 0.02 by default. This value should be lower than d2_bg_k1.

d4_bg Real: Dimensionless coefficient for background higher-order divergence damping. 0.0 by default. If no second-order divergence damping is used, then values between 0.1 and 0.16 are recommended. Requires nord > 0. Note that the scaling for d4_bg differs from that of d2_bg; nord >= 1 and d4_bg = 0.16 will be less diffusive than nord = 0 and d2_bg = 0.02.

dddmp Real: Dimensionless coefficient for the second-order Smagorinsky-type divergence damping. 0.0 by default. 0.2 (the Smagorinsky constant) is recommended if ICs are noisy.

d_ext Real: coefficient for external (barotropic) mode damping. 0.02 by default. Proper range is 0 to 0.02. A value of 0.01 or 0.02 may help improve the models maximum stable time step in low-resolution (2-degree or poorer) simulations; otherwise a value of 0 is recommended.

do_vort_damp Logical: whether to apply flux damping (of strength governed by vtdm4) to the fluxes of vorticity, air mass, and nonhydrostatic vertical velocity (there is no dynamically correct way to add explicit diffusion to the tracer fluxes). The form is the same as is used for the divergence damping, including the same order (from nord) damping, unless nord = 0, in which case this damping is fourth-order, or if nord = 3, in which case this damping is sixth-order (instead of eighth-order). We recommend enabling this damping when the linear or non-monotonic horizontal advection schemes are enabled, but is unnecessary and not recommended when using monotonic advection. False by default.

n_sponge Integer: controls the number of layers at the upper boundary on which the $2\Delta x$ filter is applied. *This does not control the sponge layer.* 0 by default.

nord Integer: order of divergence damping: 0 for second-order; 1 for fourth-order (default); 2 for sixth-order; 3 for eighth-order. Sixth-order may yield a better solution for low resolutions (one degree or coarser) by virtue of it being more scale-selective and will not damp moderately-well-resolved disturbances as much as does lower-order damping.

nord_tr Integer: Order of tracer damping; values mean the same as for nord. 0 by default.

rf_cutoff Real: pressure below which no Rayleigh damping is applied if tau > 0.

rf_fast Logical: option controlling whether to apply Rayleigh damping (for tau > 0) on the dynamic/acoustic timestep rather than on the physics timestep. This can help stabilize the model by applying the damping more weakly more frequently, so the instantaneous amount of damping (and thereby heat added) is reduced. .false. by default, which applies the Rayleigh drag every physics timestep.

tau Real: time scale (in days) for Rayleigh friction applied to horizontal and vertical winds; lost kinetic energy is converted to heat, except on nested grids. 0.0 by default, which disables damping. Larger values yield less damping. For models with tops at 1 mb or lower values between 10 and 30 are useful for preventing overly-strong polar night jets; for higher-top hydrostatic models values between 5 and 15 should be considered; and for non-hydrostatic models values of 10 or less should be considered, with smaller values for higher-resolution.

vtdm4 Real: coefficient for background other-variable damping. The value of vtdm4 should be less than that of d4_bg. A good first guess for vtdm4 is about one-third the value of d4_bg. 0.0 by default. Requires do_vort_damp to be .true. Disabled for values less than 1.e-3. Other-variable damping should not be used if a monotonic horizontal advection scheme is used.

B. Entries in coupler_nml

months, days, hours, minutes, seconds Integer: length of the model integration in the corresponding units. All are 0 by default, which initializes the model and then immediately writes out the restart files and halts.

dt_atmos Integer: time step for the largest atmosphere model loop, corresponding to the frequency at which the top level routine in the dynamics is called, and the physics timestep. Must be set.

current_date Integer(6): initialization date (in the chosen calendar) for the model, in year, month, day, hour, minute, and second. (0,0,0,0,0,0) by default, a value that is useful for control integrations of coupled models.

calendar Character(17): calendar selection; JULIAN is typically recommended, although the other values (THIRTY_DAY_MONTHS, NOLEAP, NO_CALENDAR) have particular uses in idealized models. Must be set.

force_date_from_namelist Logical: if .true., will read the initialization date from the namelist variable current_date, rather than taking the value from a restart file. If the model is being cold-started (such as what is typically but not necessarily done if external_ic = .true.) then the initialization date must be specified in current_date, otherwise the model will stop. .false. by default.

atmos_nthreads Integer: number of threads for OpenMP multi-threading. 1 by default.

use_hyper_thread Logical: indicates that some of the threads in atmos_nthreads may be hyper-threads. .false. by default.

ncores_per_node Integer: number of processor codes per physical compute node. Used when setting up hyperthreading to determine number of virtual vs. hardware threads. 0 by default.

debug_affinity Logical: if .true. prints out a message describing cpu affinity characteristics while initializing OpenMP. .false. by default.

restart_days, restart_secs Integer: frequency at which to write out ”intermediate” restart files, which are useful for checkpointing in the middle of a long run, or to be able to diagnose problems during the model integration. Both are 0 by default, in which case intermediate restarts are not written out.

C. Entries in external_ic_nml

filtered_terrain Logical: whether to use the terrain filtered by the preprocessing tools rather than the raw terrain. .true. by default. Only active if nggps_ic = .true. or ecmwf_ic = .true.

levp Integer: number of levels in the input (remapped) initial conditions. 64 by default. Only active if nggps_ic = .true.

checker_tr Logical: whether to enable the “checkerboard” tracer test. .false. by default. Only active if nggps_ic = .true.

nt_checker Integer: number of tracers (at the end of the list of tracers defined in field_table) to initialize with an idealized "checkerboard" pattern, with values of either 0 or 1. This is intended to test the monotonicity or positivity constraint in the advection scheme. 0 by default. Only active if nggps_ic = .true.

D. Entries in surf_map_nml

surf_file Character(len=128): File containing topography data. This file must be in NetCDF format. INPUT/topo1min.nc by default. (Previous versions of the model have used 5 minute USGS data, which is not recommended.)

nlon Integer: Size of the longitude dimension in topography data; not used.

nlat Integer: Size of the latitude dimension in topography data; not used.

zero_ocean Logical: whether to prevent the smoothing from extending topography out into the ocean. True by default.

zs_filter Logical: whether to apply smoothing to the topography. True by default.

E. Entries in fv_grid_nml

grid_name Character(len=80): Name of the grid either being read in (if grid_spec = -1) or being created. This is only used for writing out a binary file in the directory from which the model is run. Gnomonic by default.

grid_file Character(len=120): If grid_type = -1 the name of the grid_spec file to read in. INPUT/grid_spec.nc by default; other values will not work.

F. Entries in test_case_nml

test_case Integer: number of the idealized test case to run. A number of nest cases are defined in tools/test_cases.F90, of which numbers 19 are intended for the shallow-water model. Requires warm_start =.false. 11 by default; this creates a resting atmosphere with a very basic thermodynamic profile, with topography. If you wish to initialize an Aquaplanet simulation (no topography) set to 14.

alpha Real: In certain shallow-water test cases specifies the angle (in fractions of a rotation, so 0.25 is a 45-degree rotation) through which the basic state is rotated. 0 by default.

G. Entries in nest_nml

ngrids Integer: number of grids in this simulation. 1 by default. (The variable ntiles has the same effect, but its use is not recommended as it can be confused with the six tiles of the cubed sphere, which in this case form only one grid.)

nest_pes Integer(100): array carrying number of PEs assigned to each grid, in order. Must be set if ngrids > 1.

p_split Integer: number of times to sub-cycle dynamics, performing nested-grid BC interpolation and (if twowaynest ==.true.) two-way updating at the end of each set of dynamics calls. If p_split $>$ 1 the user should decrease k_split appropriately so the remapping and dynamics time steps remain the same. 1 by default.

imfdeep These are options for a variety of convection schemes: for example IIRC imfdeep = 1 is old SAS, imfdeep = 2 is scaleaware SAS, 3 is either RAS or CS, etc. The best reference is to look for the options in GFS_physics_driver.F90

do_deep The correct options to disable the convection schemes are do_deep = .F. and shal_cnv = .F.

H. Entries in nggps_diag_nml

fdiag Real(1028): Array listing the diagnostic output times (in hours) for the GFS physics. This can either be a list of times after initialization, or an interval if only the first entry is nonzero. All 0 by default, which will result in no outputs.

I. Entries in atmos_model_nml (for fvGFS)

blocksize Integer: Number of columns in each “block” sent to the physics. OpenMP threading is done over the number of blocks. For best performance this number should divide the number of grid cells per processor ((npx-1)*(npy-1) / (layout_x)*(layout_y)) and be small enough so the data can best fit into cache?values around 32 appear to be optimal on Gaea. 1 by default

chksum_debug Logical: whether to compute checksums for all variables passed into the GFS physics, before and after each physics timestep. This is very useful for reproducibility checking. .false. by default.

dycore_only Logical: whether only the dynamical core (and not the GFS physics) is executed when running the model, essentially running the model as a solo dynamical core. .false. by default.

J. Entries in fms_nml

domains_stack_size Integer: size (in bytes) of memory array reserved for domains. For large grids or reduced processor counts this can be large (¿10 M); if it is not large enough the model will stop and print a recommended value of the stack size. Default is 0., reverting to the default set in MPP (which is probably not large enough for modern applications).

K. Important Relations

Cell-mean pressure:

$$p^* = \frac{\delta p^*}{\delta \log p^*} \qquad \text{(hydrostatic)} \qquad \text{(K1)}$$

$$p = \rho R_d T_v = -\frac{R_d}{g}\frac{\delta p^* T_v}{\delta z} \qquad \text{(nonhydrostatic)} \qquad \text{(K2)}$$

Conversion from temperature T to virtual potential temperature Θ_v:

$$T_v = T\left(1 + \epsilon q_v\right) \qquad \text{(K3)}$$

$$\Theta_v = T_v / p^\kappa, \qquad \text{(K4)}$$

where $\epsilon = 1 + R_v/R_d$ and $\kappa = R/c_p$. Note that we do *not* include the arbitrary constant factor p_0^κ in our definition of potential temperature; our form is the equivalent to setting $p_0 = 1$ Pa but simpler and more computationally efficient.

L. References

CW84 Collella, P, and P. R. Woodward, April 2004: The Piecewise Parabolic Method (PPM) for Gas-Dynamical Simulations. *Journal of Computational Physics*, **54**, 174–201.

L94 Lin, Shian-Jiann, et al., July 1994: A Class of the van Leer-type Transport Schemes and Its Application to the Moisture Transport in a General Circulation Model. *Monthly Weather Review*, **122:7**, 1575–1593.

LR96 Lin, Shian-Jiann, and R B Rood, September 1996: Multidimensional Flux-Form Semi-Lagrangian Transport Schemes. *Monthly Weather Review*, **124:9**, 2046–2070.

LR97 Lin, Shian-Jiann, and R B Rood, October 1997: An explicit flux-form semi-lagrangian shallow-water model on the sphere. *Quarterly Journal of the Royal Meteorological Society*, **123:544**, 2477–2498.

L97 Lin, Shian-Jiann, July 1997: A finite-volume integration method for computing pressure gradient force in general vertical coordinates. *Quarterly Journal of the Royal Meteorological Society*, **123:542**, 1749–1762.

L04 Lin, Shian-Jiann, 2004: A ”vertically Lagrangian” finite-volume dynamical core for global models. *Monthly Weather Review*, **132:10**, 2293–2307.

PL07 Putman, W M., and Shian-Jiann Lin, 2007: Finite-volume transport on various cubed-sphere grids. *Journal of Computational Physics*, **227:1**, 55–78.

HL13 Harris, Lucas M., and Shian-Jiann Lin, January 2013: A two-way nested global-regional dynamical core on the cubed-sphere grid. *Monthly Weather Review*, **141:1**.

HLT16 Harris, L.M., S.-J. Lin, and C.-Y. Tu, 2016: High-Resolution Climate Simulations Using GFDL HiRAM with a Stretched Global Grid. *Journal of Climate*, **29:11**, 4293–4314.

M. Revision History

v2.0beta.01 *16 Jan 2018* At ”Entries in nest_nml” section added two more items:

imfdeep

do_deep

v2.0beta *14 Nov 2017* Release candidate to EMC — lmh