Notes 2025-02-10

Carrie Hashimoto

Table of contents

LSM

- geo_em.d0x.nc
- wrfinput_d0x.nc
- GENPARM.TBL
- MPTABLE.TBL
- SOILPARM.TBL (not used?)
- soil_properties.nc

geo_em.d0x.nc

User guide

Namelist variables

- To create, define model domain:
- Temporal: 2019-10-27 00:00 to 2019-11-05 00:00

Extent set-up

Lat 29.5, 30.5 Lon -96.1, -95.1

Choosing a projection:

“The polar stereographic projection is best suited for high-latitude WRF domains, the Lambert conformal projection is well-suited for mid-latitude domains, and the Mercator projection is good for low-latitude domains or domains with predominantly west-east extent.”

e_we = 97 east-west grid edges, number of velocity points

Reasoning for selection:

- Length at top of domain is 95.8 km; at bottom is 96.8 km
- dx is 1000 m
- Start with 97 grid edges

e_sn = 112 south-north grid edges

- Similar reasoning: length of side of domain is 111.2 km
- dy is 1000 m
- Start with 112 grid edges

Extent setup

ref_lat = 30 lat of center point

ref_lon = -95.6 lon of center point

dx = 1000 (default) grid distance in x

- The grid distance is in meters for the ‘polar’, ‘lambert’, and ‘mercator’ projection, and in degrees longitude for the ‘lat-lon’ projection.

dy = 1000 grid distance in y (examples use 1000 and 30000 - choosing first, but might need to coarsen if model is too large)

Extent setup

map_proj = 'lambert'

- Projection; from 'lambert', 'polar', 'mercator', and 'lat-lon'.
- Lambert best for mid-latitudes; mercator for low latitudes - in between

truelat1 = 30

- first true latitude for the Lambert conformal conic projection, or the only true latitude for the Mercator and polar stereographic projections.

truelat2 = 60

- second true latitude for the Lambert conformal conic projection.

stand_lon = -97

- A real value specifying, the longitude that is parallel with the y-axis in the Lambert conformal and polar stereographic projections. For the regular latitude-longitude projection, this value gives the rotation about the earth's geographic poles. No default value.

geog_data_res =

- In their example, they use 'nlcd2011_30m+gtopo_30s+default', but the static data they use are somewhat different than the default - ex instead of nlcd2011_30m land use, the land use in the default static data is modis_landuse_20class_30s_with_lakes, so I'm not sure if I should input that as the default resolution since it's 30 second instead of 30 minute
- A character string specifying a corresponding resolution or list of resolutions separated by + symbols of source data to be used when interpolating static terrestrial data to the grid. This string should contain a resolution matching a string preceding a colon in a rel_path or abs_path specification (see the description of GEOGRID.TBL options) in the GEOGRID.TBL file for each field. If a resolution in the string does not match any such string in a rel_path or abs_path specification for a field in GEOGRID.TBL, a default resolution of data for that field, if one is specified, will be used. If multiple resolutions match, the first resolution to match a string in a rel_path or abs_path specification in the GEOGRID.TBL file will be used. Default value is 'default'.
- First, geogrid looks for resolution used on nlcd2011_30m, and moves to gtopo_30s if it can't find it, before using the default if needed

geog_data_path = '/home/docker/WRF_WPS/geog_high_res_mandatory'

- Path to geographic data; I downloaded the CONUS static data from WRF-Hydro and added to the Docker since they were missing
- Note: the tutorial uses a reduced dataset called "geog_conus", but they warn that the data in this version of the static data aren't supported for non-training purposes

sources for WRF-Hydro default static data

wrfinput_d0x.nc

GEMPARM.TBL

MPTABLE.TBL

SOILPARM.TBL

Spatial soil parameters

- create_SoilProperties.R is used to create gridded soil properties

Terrain Routing

- Fulldom_hires.nc
- hydro2dtbl.nc or HYDRO.TBL

Fulldon_hires.nc

hydro2tbl.nc

Channel and Reservoir Routing

- Fulldom_hires.nc
- CHANPARM.TBL
- Route_Link.nc
- nudgingParams.nc
- *.usgsTimeSlice.ncdf
- LAKEPARM.nc
- GWBASINS.nc
- GWBUCKPARM.nc
- spatialweights.nc

Making my own Docker

- miniconda3 folder
  - Not certain what I'll need in here, but start by using one of their base dockerfiles
- WRF_WPS folder: contains utilities, WPS, and WRFV3
- Since I'm using WRF-Hydro, not WRF, try just including WRF-Hydro 5.3.0 instead
- And use most recent version of WPS, 4.6.0
- Also, include v4 static data instead of v3, which is used in the training
- Also include training lessons for both main WRF-Hydro tutorial and WPS
- Build from Dockerfile and bind-mount these folders to the Docker
- Software I know the Docker needs:
- gfort