

Advanced Features of the WRF Preprocessing System

Michael Duda



The WRF Users' Basic Tutorial 27 – 31 July 2015, Boulder, CO

*NCAR is sponsored by the National Science Foundation

Outline

- The GEOGRID.TBL file
 - What is the GEOGRID.TBL file?
 - Ingesting new static fields
 - Examples: Using high-resolution land use and topography data
- The METGRID.TBL file
 - What is the METGRID.TBL file?
 - Example: Defining interpolation options for a new field
 - Example: Using the METGRID.TBL file for a real-time system
- Utility programs example: fixing "hot lakes"



The GEOGRID.TBL File

- GEOGRID.TBL is the file that determines which fields are interpolated by geogrid *at runtime*
 - Each entry in GEOGRID.TBL corresponds to one data source
 - When new data sources are involved, or when the default treatment of fields is inadequate, user may want/need to edit GEOGRID.TBL
 - However, default GEOGRID.TBL is sufficient to initialize a WRF simulation



The GEOGRID.TBL File

- Format of GEOGRID.TBL file is simple text, with specifications of the form keyword=value
- Example entry for a 30" landuse data set:

name=LANDUSEF # Houston, TX urban data
 priority = 1
 dest_type = categorical
 z_dim_name = land_cat
 interp_option = 30s:nearest_neighbor
 abs_path = 30s:/users/duda/Houston/



For a complete list of possible keywords See p. 3-46

The GEOGRID.TBL File

- Using the GEOGRID.TBL, we can
 - Change the method(s) used to interpolate a field
 - Apply smoothing filters to continuous fields
 - Derive fields from others
 - E.g., dominant category or slope fields
 - Add new data for geogrid to interpolate



New Fields in GEOGRID.TBL

There are three basic types of new data to be added through the GEOGRID.TBL file:

- 1) Completely new fields
 - fields that were previously not processed by geogrid
- 2) Different resolution data sets for an existing field
 - Such sources *do not need to be supplemented* by existing data
 - E.g., Adding a 90-meter resolution topography data set
- 3) Alternative sources for a field that *must be used in addition to an existing source*
 - E.g., A new soil category data set exists, but covers only South Korea



1) Completely new fields



name = MY NEW FIELD NAME

dest type = continuous

interp option = four pt

priority = 1

abs path

For a new field, simply add an entry in GEOGRID.TBL for that field.



Priority of this data source compared with other sources for same field

How to interpolate this field

Where on disk to find the data for this field



= /data/duda/mydata/

2) Different resolution data set

Different resolution data sets for an existing field :

Specify the path to the new data set and which interpolation methods should be used for the new resolution in the <u>existing entry for that field</u>.

```
name = HGT_M
priority = 1
dest_type = continuous
smooth_option = smth-desmth
interp_option = 30s:special(4.0)+four_pt
interp_option = my_res:four_pt
interp_option = default:four_pt
rel_path= 30s:topo_30s/
rel_path= my_res:new_topo_directory/
rel_path= default:topo_2m/
```



3) Alternative data sources

Alternative sources for a field that must be used in addition to an existing source :

Add a new entry for the field that has the same name as the field's existing entry, but make priority of new entry <u>higher</u>.

```
name = HGT_M
    priority = 2
    dest_type = continuous
    interp_option = default:four_pt
    rel_path = default:some_path/

name = HGT_M
    priority = 1
    dest_type = continuous
    interp_option = default:four_pt
    rel_path = default:four_pt
```



Preparing new geogrid data sets

To add a new data source, we need to

- 1) Write the data in the proper binary format
 - See Chapter 3: "Writing Static Data to the Geogrid Binary Format"
 - Can make use of read_geogrid.c and write_geogrid.c
- 2) Create an "index" metadata file for the data set
 - This tells geogrid about the projection, coverage, resolution, type, and storage representation of the data set
- 3) Add/edit entry for the data in the GEOGRID.TBL file
 - The change to GEOGRID.TBL will follow one of the three cases mentioned before



The geogrid format is a simple binary raster

- Elements of a rectangular array of data are written, row by row, to a file
- No record markers or any type of metadata are written to this file

8 rows	С	С	2	Э	С	2	
	a 43	a 44	a 45	a 46	a 47	a 48	
	a 37	a 38	a 39	$\mathbf{a}_{_{40}}$	$\mathbf{a}_{_{41}}$	a 42	See p. 3-37
	a ₃₁	a ₃₂	a 33	a 34	a 35	a 36	start of file
	a ₂₅	a ₂₆	a ₂₇	a ₂₈	a ₂₉	a 30	$a_1 a_2 a_3 (a_{47} a_{48})$
	a 19	a_{20}	a ₂₁	a ₂₂	a ₂₃	a 24	
	a_{13}	\mathbf{a}_{14}	a 15	a_{16}	a 17	a ₁₈	end of file
	a_7	a_{s}	a,	$\mathbf{a}_{\scriptscriptstyle 10}$	\mathbf{a}_{11}	a 12	
	a,	a 2	a 3	a ₄	a₅	a 6	A file containing a N×M array, with

6 columns

A file containing a N×M array, with each element represented using K bytes, should have size exactly N*M*K bytes!





- Since the contents of the file contain <u>only</u> the values from the array, *care must be taken if using Fortran to write the array*
 - Fortran unformatted writes add *record markers* to the beginning and end of each record
 - So, rather than $X_1X_2X_3...X_{n-1}X_n$ we get $RX_1X_2X_3...X_{n-1}X_nR$, where R is a record marker

Instead of Fortran, the C routines read_geogrid.c and write_geogrid.c may be used to read and write binary files



- these may be called from either Fortran or C

The filenames of geogrid binary files should have the form:

xxxxx-XXXXX.yyyyy-YYYYY

where

XXXXX	is the starting x-index
XXXXX	is the ending x-index
ууууу	is the starting y-index
YYYYY	is the ending y-index

E.g., For a binary file containing an array with 500 columns and 750 rows, the file name would be 00001-00500.00001-00750



- If the data are not available in a single tile (array), multiple files may be used to store the data
- All tiles must have the same x-dimension
- All tiles must have the same y-dimension
- If necessary, a tile can be "padded" with missing values to expand it to the same size as other tiles in the data set





- If the data do not cover a rectangular region, areas with no data are simply filled with a missing value so that the overall data set is rectangular
- The particular missing value used in the data set is specified in the index metadata file for the data set







- Given dataset for new Houston urban land use categories
 - Regular lat/lon projection, 30" resolution; categories 31, 32 & 33



Area of Houston data tile in relation to model domain – white=missing data and blue=valid data



The WRF Users' Basic Tutorial 27 – 31 July 2015, Boulder, CO

To make use of the new data, we do the following:

1) Write the data to the binary format used by geogrid

2) Create an index file for the data

```
Data set has categories
type=categorical
                                           31 through 33
category min=31; category max=33
projection=regular 11
                                             30 arc second resolution
dx=0.00833333; dy=0.00833333
               known y=1.0
known x=1.0;
                                           Geographic location of
known lat=29.3375
known lon=-95.9958333
                                           data set
wordsize=1
tile x=157; tile y=143; tile z=1
missing value = 0.
units="category"
                                     Treat 0 as "no data"
description="3-category urban LU"
                                                       See p. 3-49
```



3) Define an entry for the data in GEOGRID.TBL





4) Run geogrid.exe

Any gridpoints covered by Houston data will use it; otherwise default USGS data will be used







The WRF Users' Basic Tutorial 27 – 31 July 2015, Boulder, CO

Example: South Korea

Shuttle Radar Topography Mission (SRTM) 3 arc second topography data

- We would like to use the SRTM data, especially for domains 2 and 3.
- Follow steps for adding a new resolution for an existing data set (case 2)





Example: Seoul

To use the SRTM topography data, we

- 1) Write data to geogrid binary format
- 2) Create an index file for the data set
- 3) Modify the GEOGRID.TBL entries for HGT_M, HGT_U, and HGT_V

```
name = HGT_M
priority = 1
dest_type = continuous
interp_option = 30s:special(4.0)+four_pt
interp_option = SRTM:four_pt
rel_path = 30s:topo_30s/
rel_path = SRTM:SRTM/
```

4) Specify that we should interpolate from SRTM in namelist by setting geog_data_res = '30s','SRTM+30s','SRTM+30s'



Example: Seoul



Domain 3 (DX=111m) using default 30" USGS topography

Domain 3 (DX=111m) using 3" SRTM topography



The WRF Users' Basic Tutorial 27 – 31 July 2015, Boulder, CO

Another Example: Los Angeles

For Los Angeles, we have a 30-meter resolution, 3 urban land use category data set



The WRF Users' Basic Tutorial 27 – 31 July 2015, Boulder, CO

Outline

- The GEOGRID.TBL file
 - What is the GEOGRID.TBL file?
 - Ingesting new static fields
 - Examples: Using high-resolution land use and topography data
- The METGRID.TBL file
 - What is the METGRID.TBL file?
 - Example: Defining interpolation options for a new field
 - Example: Using the METGRID.TBL file for a real-time system
- Utility programs example: fixing "hot lakes"



The METGRID.TBL File

The METGRID.TBL file controls how meteorological fields are interpolated

- Unlike GEOGRID.TBL, METGRID.TBL *does not determine which fields will be processed*, only
 how to process them if they are encountered
- Every field in intermediate files will be interpolated
 - If no entry in METGRID.TBL for a field, a default interpolation scheme (<u>nearest neighbor</u>) will be used
 - It is possible to specify in METGRID.TBL that a field should be discarded





The METGRID.TBL File

- Suitable entries in METGRID.TBL are provided for common fields
 - Thus, many users will rarely need to edit METGRID.TBL
- When necessary, different interpolation methods (and other options) can be set in METGRID.TBL
 - Interpolation options can depend on the source of a field



The METGRID.TBL File

 Example METGRID.TBL entry (for "soil moisture 0–10 cm")

```
name=SM000010
interp_option=sixteen_pt+four_pt+average_4pt
masked=water
interp_mask=LANDSEA(0)
fill_missing=1.
flag_in_output=FLAG_SM000010
```



- Suppose we have a 1000×1000 domain over Houston (dx=500 m)
 - This is the same domain as in the urban land use example
- Meteorological data come from 1-degree GFS
 - Note that we will be interpolating 1-degree data onto a 500-m grid!
- We want to create an entry for a new soil moisture field, SM000010



• Initially, we run metgrid.exe and get the message:

INFORM: Entry in METGRID.TBL not found for field SM000010. Default options will be used for this field!

- The resulting SM000010 field looks very coarse
- We need to create a METGRID.TBL entry so metgrid will know how to interpolate this field!





• We add an initial entry in METGRID.TBL for SM000010:



27 – 31 July 2015, Boulder, CO

 Now, after running metgrid.exe again, the SM000010 field looks like





The WRF Users' Basic Tutorial 27 – 31 July 2015, Boulder, CO

But, the interpolated field still looks bad near the coastline



27 – 31 July 2015, Boulder, CO

• Update the METGRID.TBL entry for SM000010

```
name = SM000010
masked = water
interp_mask = LANDSEA(0)
interp_option = sixteen_pt + four_pt + average_4pt
fill_missing = 0.
```

- If 16-pt doesn't work, then try 4-pt before reverting to a 4-point average
 - Note that 4-point average will work anywhere nearest_neighbor would (missing/masked values not counted in the average)



• The resulting field, below-left:



Interpolated SM000010 field (sixteen_pt + four_pt + average_4pt)



Which interpolator was used at each model grid point



 By using wt_average_4pt instead of average_4pt:



sixteen_pt + four_pt + average_4pt



sixteen_pt + four_pt + wt_average_4pt



The WRF Users' Basic Tutorial 27 – 31 July 2015, Boulder, CO

- Suppose we have a real-time system that:
 - Uses GFS for initial and boundary conditions
 - When possible (i.e., if the files are available soon enough) uses soil moisture and soil temperature fields from AGRMET
- In our system, it may occasionally happen that the AGRMET files are not ready when we want to start our WRF run
 - Because system is real-time, we want to proceed using just the GFS land surface fields!



• We already know how to run ungrib on multiple sources of data to get

GFS:YYYY-MM-DD_HH

and

```
AGRMET:YYYY-MM-DD_HH
```

intermediate files, and specify

```
fg_name = 'GFS', 'AGRMET',
```

in the &metgrid namelist record to use both
sources

See p. 3-24



Without further changes, what happens if:

Only GFS data are available when we run metgrid

Metgrid runs and warns that no AGRMET data files were found:

```
Processing 2012-04-01_00

GFS

AGRMET

WARNING: Couldn't open file AGRMET:2012-04-01_00 for

input.
```

Metgrid will finish, but will only use GFS data!



And the 0–10 cm soil moisture field (SM000010) looks like:





The WRF Users' Basic Tutorial 27 – 31 July 2015, Boulder, CO

However, what happens if:

Both GFS and AGRMET files are available when we run metgrid?

Our SM000010 field looks like:



We get unreasonable values with magnitude ~1E30 near land-water boundaries!

The WRF Users' Basic Tutorial 27 – 31 July 2015, Boulder, CO

Why are there bad values near coastlines? What went wrong?

- In both Vtable.GFS and Vtable.AGRMET, the land-sea mask field is named LANDSEA
 - In METGRID.TBL, our entry for SM000010 says:

```
name=SM000010
interp_option=sixteen_pt+four_pt+wt_average_4pt+search
masked=water
interp_mask=LANDSEA(0)
fill_missing=1.
flag_in_output=FLAG_SM000010
```



```
name=SM000010
interp_option=sixteen_pt+four_pt+wt_average_4pt+search
masked=water
interp_mask=LANDSEA(0)
fill_missing=1.
flag_in_output=FLAG_SM000010
```

After metgrid reads in LANDSEA from GFS file *to use as an interpolation mask*, it ignored the LANDSEA field from AGRMET *for use as a mask*.

- So, metgrid used the GFS LANDSEA mask even when interpolating AGRMET data!

When metgrid interpolated SM000010, it used the GFS landmask for a field masked by the AGRMET landmask!

GFS LANDSEA field

AGRMET LANDSEA field

Note the disagreement between the two data sources near coastlines.

Solution:

- Rename LANDSEA to AGR_LAND in Vtable.AGRMET
- Rename LANDSEA to *GFS_LAND* in Vtable.GFS
- Create separate entries in METGRID.TBL one for GFS SM000010 field another for AGRMET SM000010 field


```
name=SM000010; from_input=GFS
interp_option=sixteen_pt+four_pt+wt_average_4pt+search
masked=water
interp_mask=GFS_LAND(0)
fill_missing=1.
flag_in_output=FLAG_SM000010
```


With modified Vtables and METGRID.TBL:

The SM000010 field when only GFS files are available

The SM000010 field when both GFS and AGRMET files are available

Outline

- The GEOGRID.TBL file
 - What is the GEOGRID.TBL file?
 - Ingesting new static fields
 - Examples: Using high-resolution land use and topography data
- The METGRID.TBL file
 - What is the METGRID.TBL file?
 - Example: Defining interpolation options for a new field
 - Example: Using the METGRID.TBL file for a real-time system
- Utility programs example: fixing "hot lakes"

Motivating Problem

The "Hot Lake" problem: Inland water bodies that are not resolved by SST data sets often receive extrapolated values from nearby oceans or other resolved water bodies.

Above left: Skin temperature field (TSK) for Iceland and surrounding ocean on 26 January 2011 1200 UTC from NCEP GFS and RTG SST data.

Above right: TSK in the Mývatn region. SST for Mývatn Lake is ~277 K!

Approach

In WRF v3.3 and later, let the *real* preprocessor know which water points are inland water bodies, and provide it a more accurate estimate of SST to be used only over these water bodies.

1) Identify inland water bodies in the land cover data set

- 2) Provide a suitable proxy for SST field over inland water bodies
 - E.g., Average surface air temperature for X days prior, 273 K for frozen lakes, etc.
- 3) Modify the SST field in the WRF input file
 - Use new capability in v3.3 real.exe program

Identifying Lakes

Some data sets already identify lakes with separate categories

MODIS, CORINE

For others, we need a way to do this

- Should be automated
 - don't want to spend long hours clicking on pixels for each data set
- Should be tunable
 - what constitutes a lake will naturally depend on what our SST data set is able to resolve
- Ideally, would not require auxiliary data

In namelist.wps, set:

- geog_data_res = "usgs_lakes+30s" for USGS land use (16=ocean, 28=lake)
- geog_data_res = "modis_lakes+30s" for MODIS land use (17=ocean, 21=lake)

Creating a Proxy SST Field

- The *avg_tsfc.exe* utility program may be used to compute the average 2-m air temperature field for any number of full diurnal cycles
- Number of cycles determined by available intermediate files and date range in namelist
- The resulting TAVGSFC intermediate file may be provided to the metgrid program

Test case: Lake Mývatn

To confirm that everything is working as expected, try correcting the temperature for Lake Mývatn in the winter

Grid ID	Resolutio n	Size
1	16 km	99x99
2	4 km	208x172
3	1 km	136x128
4	250 m	160x160

Ics + BCs from NCEP GFS

Sea surface temperatures from RTG SST Initial time: 26 January 2011, 1200 UTC

The WRF Users' Basic Tutorial 27 – 31 July 2015, Boulder, CO

Test case: Lake Mývatn

26 January 2011, 12 UTC

extrapolated SST values

Initial skin temperature field using previous 5-day average of 2-m air temperature for lake SST

Test case: Lake Mývatn

Time series of sensible heat flux in the center of the lake show a significant decrease when using a more realistic SST (TAVGSFC)

Latent heat flux time series from simulation using TAVGSFC for SST also shows a decrease from RTG SST time series as well

Summary

- In this lecture, we've seen
 - What the GEOGRID.TBL and METGRID.TBL files do
 - How to use new geographical data sources in the WPS
 - High-resolution land use and topography data
 - How to use the METGRID.TBL file to correct two types of interpolation-related problems
 - How utility programs can be used to improve simulations
- For other features of the WPS, see Chapter 3 of the User's Guide
- For more information about using high-resolution topography data or urban land use data (over the U.S.), see http://www2.mmm.ucar.edu/people/duda/files/
 how_to_hires.html

Questions?

