ATSC 5014 (Dynamic Meteorology)
GEMPAK/IDV II
September 17, 2015

Model Output and GEMPAK/IDV

Today we will use GEMPAK to examine the geostrophic wind field as output from the National Centers for Environmental Prediction (NCEP) North American Model (NAM). The NAM is the key operational model in the suite of numerical products offered by NCEP. It has at its core the Weather Research and Forecasting model (WRF) that is run at a horizontal resolution of 12 km four times daily for an 84-hour period. We will first create maps using GEMPAK for the 1200 UTC NAM simulation and use the analyses to compute the geostrophic wind at various locations.

Step 1: Creating soft links to the grid files

The first task is to log onto bat. It is again recommended that you make a new directory under atsc5014 (again, the 'mkdir' command in Unix) for the gridded model output, naming the new directory something like 'grids' or 'model'. Then 'cd' into this directory. The files that we are working with are found in the directory: /net/weather/data/gempak/model/nam. If you do an 'ls –l' of that directory, you will see that a number of model grids exist for each time period. You will find files such as 2015091712_nam211.gem, 2015091712_nam212.gem, etc. The files are all taken from the same NAM simulation but represent different output file resolutions. The '211', '212', '215' and '237' output is on grids with a resolution of 81-, 40-, 20- and 32-km, respectively. The full 12-km output is found on the '218' grids and can be seen in the nam12km directory that resides parallel to the nam directory on weather. The files of interest here will be the '212' grids. You can view all the nam model files in the directory by using the 'ls –l' command followed by the path and file name. For our file name, we want all files that have 'nam' in them. One way to do that is to use the wildcard notation in Unix. If we specify files as '*nam2??.gem', it is understood that we refer to all files that contain 'nam2??.gem' somewhere in the file name. The asterisk * is the wildcard symbol and refers to any set of characters. The question mark symbol refers to any single character. Thus if we type 'ls –l /net/weather/data/gempak/model/ *nam???.gem', we are requesting a long listing for files contained in the path '/net/weather/…' that contain 'nam' in the file name with three characters following (such as '211'). You should see a number of nam files, all in the form yyyymmddhh_namxxx.gem, where yyyy is the year, mm the month, dd the date, hh the time of the start of the NAM forecast and xxx pertains to the model grids. Each file contains NAM output at three-hour increments.

To start, let's again create soft links to the NAM files for today. We could simply type the soft link command 'ln –s /net/…/fname fname' as we have done before. Today we will work with a Unix script to provide experience playing with a nice tool that saves on repetitive typing. Briefly, a script is just a series of executable commands. I have put a sample Unix script called 'softlink' at '/netdata/R1/data/parish/class/atsc5014/2015' that you can modify to create soft links. You will need to do a couple things. First, copy the file 'softlink' to your working directory on bat ('cp /netdata/R1/data/parish/class/atsc5014/2015/softlink .'). Once you have 'softlink' in the

appropriate directory on bat, you can look at it using the text editor **gedit** (my favorite) or any other text editor of your choice. Open the file and look at it. It is your task to read through the script and get an understanding of what is going on. Keep this script if you wish to do something similar in the future.

Before you can run the softlink script, you need to make certain it is 'executable'. You can establish whether the file is executable by the 'ls –l' command. If it is not executable, we can change file permissions at the Unix command line using the 'chmod' command. This 'chmod' command (abbreviated from 'change mode') is a shell command in Unix that allows the file owner to change file system modes such as permissions. If you type 'ls –l softlink', you will see something at the beginning of the descriptor that looks like: '-rw-r--r-- 1 yourname general 185 Month Day Time softlink ...'. This tells us something about the file 'softlink'. In particular, the first 11 characters tell us something about the type of file and permissions for this file. There are three categories regarding ownership: 'user', 'group' and 'others'. For the 11 characters, the first character pertains to the file (i.e., 'd' for directory, 'l' for link, '-' for regular file), the next 3 characters pertain to the 'owner', next 3 to the 'group' and last 3 to 'others'. The 'user' generally owns the file and can change permissions for the 'group' and 'others' classes. Briefly, there are permissions regarding 'reading' indicated by 'r', 'writing' indicated by 'w', 'executing' indicated by 'x' that are of interest to us. For the example above, the 11 character notation '-rw-r--r—' indicates that 'softlink' is a regular file and the 'owner' can read and write to the file and the 'group' and 'others' can only read the file. We want to make the file 'softlink' executable. To do this, simply type 'chmod +x softlink'. If we now type 'ls –l softlink', we will see that it now looks like: -rwxr-xr-x 1 yourname general 185 Month Day Time softlink. If you don't want anyone else to be able to access your file, type 'chmod go = softlink' where the 'g' and 'o' refers to 'group' and 'others' and the blank character following 'go' clears all permissions in the 'group' and 'others' categories. Check to see that it is executable and then just type 'softlink' (or ./softlink) and enter and you should see that soft links have been created to all the relevant NAM files. If your softlink has been successful, you should see three files in a light blue or teal color. Typing ls -l *.gem will give complete information about the path of the original file.

Step 2: Working with GDPLOT2 in GEMPAK

We are going to use the program **gdplot2** to create isobaric maps. All GEMPAK files that begin with 'gd' refer to gridded output sets such as from numerical models. The program **gdplot2** will be the main program we will use for analysis in this class as well as in your Synoptic Lab class next semester. You can find out what grids are in a particular GEMPAK grid file by using the program **gdinfo**. Again, the tools you learned from creating upper level maps will help out here so refer to the *.nts files that you have saved when you have questions regarding the appropriate set of parameters.

When you type **gdplot2** in your new directory, you will see something that looks like the following:

GDFILE	**Grid file**	**$GEMDATA/HRCBOB.GRD**
GDATTIM	**Grid date/time**	**LAST**
GLEVEL	**Grid level**	**500**

GVCORD	**Grid vertical coordinate**	**PRES**
PANEL	**Panel loc/color/dash/width/regn**	**0**
SKIP	**Skip_cntr/skip_plt_x;skip_plt_y**	**0**
SCALE	**Scalar scale / vector scale**	**999**
GDPFUN	**Scalar grid or vector grid funct**	**TMPC**
TYPE	**GDPLOT2 function processing type**	**C**
CONTUR	**Subbox/smooth**	**0**
CINT	**Contour interval/min/max**	**0**
LINE	**Color/type/width/label/smth/fltr**	**3**
FINT	**Fill interval/min/max**	**0**
FLINE	**Fill colors/fill types**	**10-20**
HILO	**Color/symbol/rng/rad/cnt/intp**	
HLSYM	**HILO txt size/posn/font/wdth/hw**	
CLRBAR	**Color/ornt/anch/x;y/ln;wd/freq|t**	
WIND	**Wind symbol/siz/wdth/typ/hdsz**	**BM1**
REFVEC	**Mag;x;y;txtsiz/font/wdth/HW;labl**	
TITLE	**Title color/line/title**	**1**
TEXT	**Size/fnt/wdth/brdr/N-rot/just/hw**	**1**
CLEAR	**Clear screen flag**	**YES**
GAREA	**Graphics area**	**WV**
IJSKIP	**Iskp;Istrt;Istp/Jskp;Jstrt;Jstp**	
PROJ	**Map projection/angles/margins|dr**	**MER**
MAP	**Map color/dash/width/filter flag**	**1**
MSCALE	**fgc;bgc;mask/units/lat;hide/valu**	**0**
LATLON	**Line color/dash/width/freq/inc/l**	
DEVICE	**Device|name|x size;y size|color**	**XW**
STNPLT	**Txtc/txt attr|marker attr|stnfil**	
SATFIL	**Satellite image filename(s)**	
RADFIL	**Radar image filename(s)**	
IMCBAR	**Color/ornt/anch/x;y/ln;wd/freq**	
LUTFIL	**Enhancement lookup table filenam**	
STREAM	**lines/arrows/stop/slow/scale**	
POSN	**Position / Text format**	**0**
COLORS	**Color list**	**1**
MARKER	**Marker color/type/size/width/hw**	**0**
GRDLBL	**Grid point label color**	**0**
FILTER	**Filter data factor**	**YES**

Your actual settings may look somewhat different depending if you have run GEMPAK routines from your current working directory previously. Again you will note that not all parameters will need to be set. Many of the parameters will look familiar from previous exercises so jump right in. To start, you will need to set GDFILE to the input file 2015091700_nam212.gem, which is the model run starting at 0000 UTC 17 September 2015. GDAT is the time of interest and we will set that to f12 indicating the 12-hour forecast. Remember that GEMPAK help is only a few keystrokes away by typing, for example, 'help GDAT' to get help with the expected format. We will start with the 300 mb map so set 'GLEVEL=300'. GVCORD should be set for PRES as shown above.

The key parameter to set is GDPFUN (or abbreviated as GDPF). That sets all parameters that you want to contour. You can find the GEMPAK variable names that can be used by GDPF by typing 'help gparm'. In our case we will start out with an analysis of the height and temperature fields plus plotting some wind barbs. To contour multiple fields, simply separate variable names

with an exclamation mark. Thus, set ‘GDPF = hght ! tmpc ! wnd’. Again, ‘help GDPF’ will let you know what parameter names exist in GEMPAK. You will find that even if particular grids do not contain a desired variable, GEMPAK sometimes (but not always) can calculate and contour it. ‘HGHT’ and ‘TMPK’ are variables contained in the original GEMPAK grid file but not ‘TMPC’. GEMPAK, however, knows that all we need is the temperature in °C so there will be no problem. The next category is TYPE. This just specifies the type of plot. In our case, we have contour plots for height and temperature and a wind barb plot so we can set ‘type = c ! c ! b. You can type ‘help type’ for more information. Again, the exclamation point separates the variables listed in TYPE. Remember that GEMPAK is not case sensitive so ‘TYPE’and ‘type’ are understood to be the same. The next parameter is CONTUR and for now leave that blank. This parameter gives us some smoothing options that we should not need to explore at this time.

The next two parameters in the list are important. To start, CINT specifies the contour intervals. For the 300 mb map, we want to plot height contours at 60-m increments with a minimum height of 8400 m (to make certain that 9000 m is contoured) and the isotherms at 4°C increments starting at, say, -60°C (to make certain 0°C is contoured). Set CINT to get the contours as indicated (use ‘help cint’ for information). The next parameter ‘LINE’ is also critical. ‘LINE’ specifies the type of contour line (1 is solid, 12 is a nice dashed line), color of line (1 is white, 2 is red, etc.), line thickness, etc. In our case, contour the heights in white, solid lines and the isotherms in red, dashed lines (for xwindows). For each variable, label the contours for every other line. If you type ‘help colors’ you can see the coding for particular colors.

We have one other key parameter that we need to consider and that is wind. To display wind vectors on the map we need to do two things. Wind is set in GDPF as we have done above. Next we must set the WIND parameter to specify how the wind information is to be displayed. The default shown above where WIND is set to BM1 implies wind barbs in units of m/s in line color 1. Again, the ‘help’ command is useful to see what options are available. Since the NAM grids contain grid point information at a relatively high resolution, we need to be careful with how much wind information is plotted. We may need to play with the FILTER parameter to specify an appropriate number of wind barbs so our map contains sufficient information but is not too cluttered. To start, try setting FILTER = YES. Once you have a nice map, make sure to save it using the ‘save fname.nts’ command where ‘fname’ is whatever name you give. For this exercise, I typically save the file something like ‘300mb.nts’.

It is your task to create 300-mb maps (with heights, temperatures and wind barbs in meters per second) for the 12 hour forecast times for each of the three files for which you have soft links. You will create Postscript files and it is worth noting that line thickness can be enhanced to make a better final map. Here are some requirements:

300-mb height contours starting at 8400 m, contour interval 60 m (solid lines); isotherms starting at -60°C, contour interval 4°C (dashed lines)

As before, each map should have latitudes at 5° increments and longitudes at 15° increments in dashed lines with color 4, thickness 2, labeling each line. Your maps again should be in Lambert Conformal projection. Once you are happy with your map, set DEVICE to ps | mymap.ps to

create a Postscript file for your map. After you have created your Postscript files, print out the maps to help with the following:

> Calculate the geostrophic wind for each map at the following locations: KLBF and KGRB. Show all details of your calculations. Remember that the latitude lines can be used as a distance scale. A distance of 111 km corresponds to one degree of latitude. For maps of the entire country, it is often convenient to choose a distance scale of 2.5° or 5° of latitude in your geostrophic wind calculations.

Remember that when you end a GEMPAK session, always type ‘gpend’ to end. Again, you can see if you inadvertently ended a GEMPAK session without using ‘gpend’ by typing ‘ipcs’ (for inter-process communication status) in a terminal window. If you are not currently using GEMPAK, you shouldn’t see your username listed in the message queues section. If you do see yourself listed, use ‘cleanup’ to clear out your orphaned queues.