Gridding NCDC daily meteorological data

Version 2.0

Greg O'Donnell, Bernt V. Matheussen, Alan Hamlet, Ed Maurer

OVERVIEW

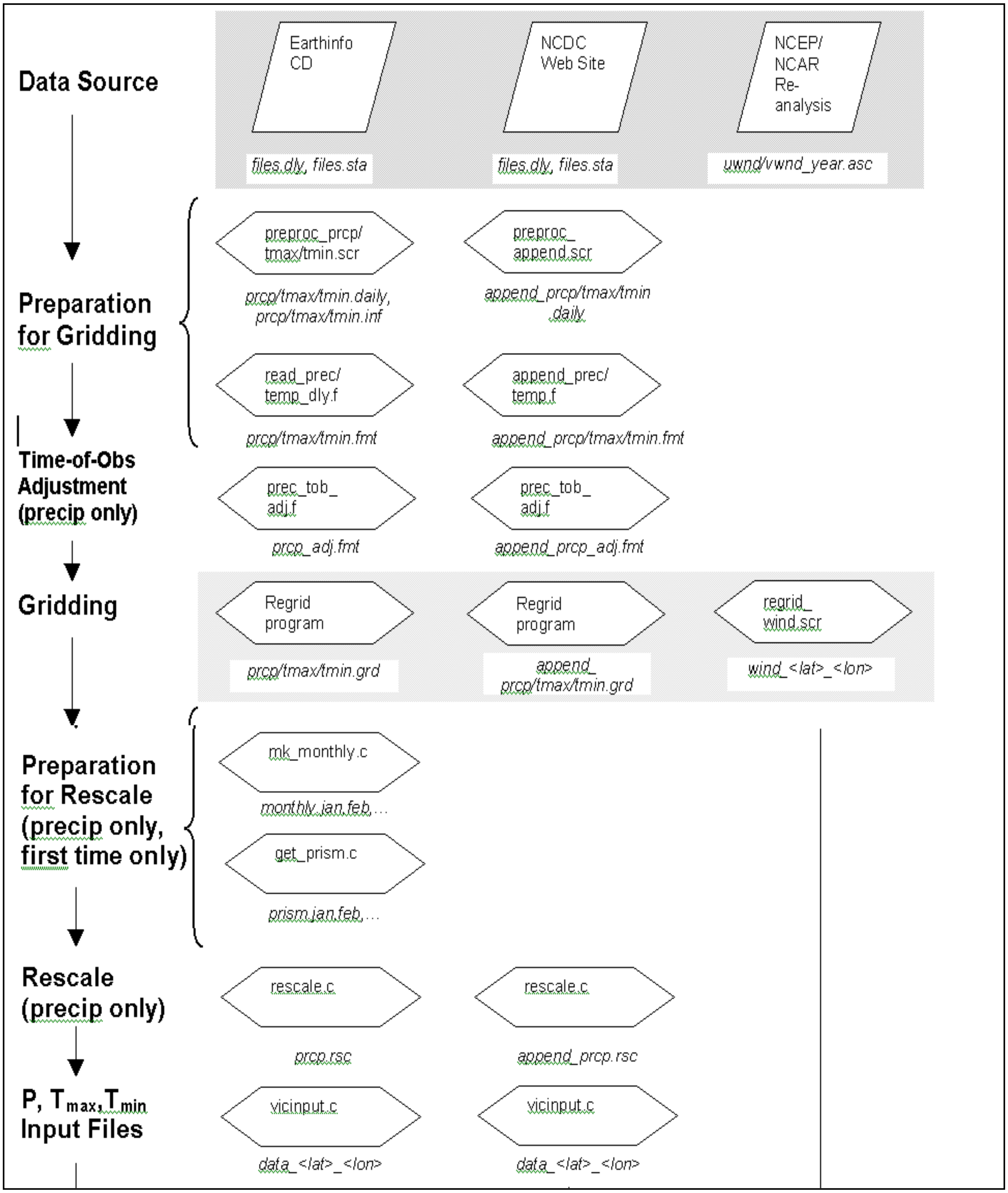
Following is a summary of the steps involved in creating a set of forcing files for the VIC model.

- Generate a DEM grid mask for the basin of interest in Arc/Info ascii format. Elevations should be in meters, and the projection should be geographic (decimal degrees)
- Retrieve the raw data for precipitation, and maximum and minimum temperature from the Earthinfo CDs (through 12/31/1997 is available).
- Preprocess each set of raw data with preproc_<param>.scr and reformat each parameter with read_<param>.f
- Download monthly data from 1/1998 through the most recent date available from the NCDC Climate Data Online page.
- Preprocess this NCDC data with preproc_append.scr and append.<param>.f (preprocessed files can be concatenated or gridded separately).
- Make a time-of-observation adjustment to precipitation using prec_tob_adj.f if desired
- Regrid each of the parameters with the regridding programs
- Calculate the monthly means of the gridded precipitation data (usually for 1961-1990) with mk_monthly.c, and extract the PRISM monthly means for the same period (get_prism.c). Rescale the precipitation to PRISM monthly means with rescale.c
- Combine the gridded rescaled precipitation and gridded temperature data into preliminary VIC input files with vicinput.c
- Generate daily wind files (from NCEP/NCAR reanalysis data) with regrid_wind.c
- Combine these into the VIC files with combine_wind.c
- Once this is done, the files can be updated with the monthly data from NCDC Climate Data Online as it becomes available, and from the CDC site for Reanalysis wind. There appears to be roughly a 10 week delay in posting the data by NCDC. The steps for appending this data use the above steps exactly as before, but setting APPEND_FLAG in the combine_wind.c program appropriately to indicate the new data is to be appended to existing files.

This process is illustrated in the flowchart below.

1 Introduction

This document provides a brief explanation of the methodology involved in developing a gridded meteorological data set (daily precipitation, maximum and minimum temperatures, and daily average wind speed) for the hydrology VIC model. The process includes two general steps, based on two data sources: 1) taking raw data from the EarthInfo National Climate Data Center (NCDC) CDs and converting them into VIC input files, and 2) appending more recent data downloaded from the NCDC On-Line Web Site. The core of the gridding process is the interpolation routine called SYMAP (Shepard, D.S., Computer Mapping: the SYMAP Interpolation Algorithm, In: Spatial Statistics and Models, Gaile and Willmott, eds., 1984). For precipitation, all the interpolated data are scaled to match long-term monthly means from the PRISM monthly precipitation dataset. For 10 meter daily wind data, gridded data are obtained from the NCEP/NCAR Reanalysis, and linearly interpolated to the VIC grid resolution.



The preprocessing steps are performed using UNIX shell scripts and programs written both in C and in Fortran 77. A basic knowledge of UNIX, C and FORTRAN is presumed, although the programs should not require alteration. The idea behind this work is to reduce the time almost all hydrologists use to prepare input files for their models. Hopefully this methodology can save time for all of us. The University of Washington takes no responsibility for any damage or errors that these programs contain or may produce. If you discover any errors, have any questions or comments PLEASE email Ed Maurer (edm@hydro.washington.edu) or Alan Hamlet (hamleaf@u.washington.edu).

1.1 Basin Definition

Before any preprocessing is possible the basin boundary at the gridded resolution must be identified. The corner coordinates, *xllcorner* and *yllcorner*, refer to the location of the bottom left of the grid - the coordinates are not cell centered, i.e. they refer to the location of the corner of the grid. The resolution should be in decimal degrees (e.g. 1.0, 0.25, 0.125). Basin boundaries may be delineated using the *GCIP Reference Data Set (GREDS)* CD-ROM, which contains the hydrological units (HUCs) for the US. The file should be imported into Arc/Info and the necessary HUCs extracted. The vector representation should then be converted to a grid and output into ASCII format, providing the necessary basin definition file (although the header format may require modification). A second source for basin delineation is the USGS site at: http://edcwww/cr.usgs.gov/landdaac/gtopo30/hydro/index.html, which has global basin and sub-basin definitions in raster data format, though it should be noted that instead of HUC codes it uses Pfafstetter codes to identify the basins. Alternatively the basin may be constructed by hand with the aid of maps.

Obtain the elevations of each of the grid cells. A global DEM data is available at a 30 arc-second or 1-kilometer resolution at: http://edcwww.cr.usgs.gov/landdaac/gtopo30/gtopo30.html. The format of the input elevation file should be identical to the mask file (described in section 2.6.3 Mask File), only rather than having integers in each cell for computational and non-computational cells, there should be elevations. Keep the finer resolution elevation file for your basin, since it will be needed in preparing the routing network.

1.2 EarthInfo NCDC Summary of the Day CD-ROMS

The NCDC Summary of the Day information is published on a set of CD-ROMS by EarthInfo, Inc. To extract the information a DOS program is available in the EarthInfo package called SD.BAT. The data on the CD-ROMS is held by state, hence the location of the basin relative to the state boundaries is helpful in extracting the information. (GMT may be used to plot the basin mask on a map with the state boundaries.) For each CD including states within the basin the following sequence of events must be performed;

- Select the states that lie at least partially within the basin and “step to” the next level, selecting “station.” A list of stations and associated data is displayed.
- Select F2 (or *ctrl*-f), filter. A list of filter options is then displayed on the lower half of the screen. First select parameter, then enter code:
  1 for Tmax, 2 for Tmin, or 12 for Precipitation.

NOTE: the precipitation should be output as **inches** and the temperature as **Fahrenheit** ((These are the defaults for the EarthInfo CDs. Use *ctrl*-u to swap between English and Metric units if necessary). The

preprocessing programs described below convert these units to those required in the VIC model.

Now displayed on the screen is a list of all stations within the given state for which the required data are available. It is advised to reduce disk requirements that those stations with very short records, which are at distance from the basin, or whose recording period does not coincide with the required period are filtered out. For example; if the basin falls within a small area of a state then restrictions on latitude and longitude may be imposed to reject stations at a great distance from the basin boundary. It should also be noted that filtering is cumulative; successive filters are applied on the already filtered set of stations.

After filtering, a list of stations meeting all the criteria should be displayed on the screen. Select all stations using *ctrl*-A, then F4, export, ASCII to save this data (list of station names, locations, and other information) to file. This station description file is usually named with a .sta extension (see example in Appendix 1.2).

To export the actual daily time-series data to a file press *return*. From the menu select *daily*. To write to a file select F4, export (prompting ‘Yes’ to ‘Write All Records’). This daily data file is typically named with a .dly extension (see example in Appendix 1.1). Data for the entire period of interest can be processed at this point – at later steps the period of a particular run can be cut back.

A general set of filtering criteria that should be considered is:

Parameter (selected as above)

Latitude and Longitude (conforming to the basin boundaries, which also screens out stations with unknown location information)

Elevation (any values below the minimum and above the maximum in the basin should be selected, to screen out stations with unknown elevation)

% Coverage (a recommended minimum value of 50%, maximum 100%)

Record Years (minimum 20 years of record for each station, with a maximum of 150)

Date (20 years after the beginning of the period of interest through the end of the CD records. This is to ensure that the minimum number of record years is actually also a minimum number of years of overlap with the period of interest. If one select a lower threshold for Record Years, End date should begin at an earlier date to reflect this.)

There should now be a *daily data file* mt_prcp.dly or_prcp.dly wa_prcp.dly
and a *station description file* mt_prcp.sta or_prcp.sta wa_prcp.sta
for each state or set of states (if more than one are on a CD) that the basin covers.
The data and the *mask* files should now be ftp-ed to a UNIX workstation for pre-processing, then gridding, and then final processing into input files in the format required for the VIC model.

2 PRECIPITATION

2.1 Transferring data to workstation with FTP

Transfer the files to the UNIX system as ASCII files not BIN files.

2.2 Pre-processing Data Files (Earthinfo CD data)

2.2.1 GZIP daily (.dly) data files*

In the new version of the gridding programs we have tried to save some disk space throughout the pre-processing steps. We then have to compress the daily data files using GZIP, for example:

```
gzip mt_prcp.dly  [or_prcp.dly  wa_prcp.dly]
```

Don’t gzip the station information files. For your own control you can word-count the station information files with wc *.sta and get the total number of stations.

Make sure that for each state (or set of states) the station information and corresponding station data files have the same prefix (e.g. for Washington use wa.sta and wa.dly), and the suffixes must be kept constant (e.g. always use sta and dly).

2.2.2 preproc_precip.scr

This script preprocesses the raw NCDC to a more suitable format. This step processes both the station information and data files.

On line 40 and 41 in the preproc_precip.scr script change the name of the output files. These will be the two files that will be produced by this preprocessing step for the next steps in gridding.

```
40        set INFO = ./ar_prcp.inf
41        set DAT = ./ar_prcp.daily
```

Also, check the extensions set in the script for the input files to make sure they match the extensions on your input files. These are set in the script on lines 43 and 44 (set EXT_DAT = dly; set EXT_INF = sta)

Before running the preproc_precip.scr you have to create a new file, "filenames", with the name of the input files. Do not include the filename extensions (or the .gz), as these are set in the script as mentioned above:

```
filenames.txt:
        /.mt_prcp
        /.or_prcp
        /.wa_prcp
```

Now run the script $preproc_precip.scr filenames.txt
The output is the station information file (see Appendix 1.4) and the daily data file (Appendix 1.3).

2.3 *read_prec_dly.f*

This program has been rewritten to reduce the binary executable size (it will typically be 1/50th the size), although it will run slightly slower.

Before compilation check that the variable nstat (total number of stations) is of sufficient size.
Also set variable maxprec (maximum daily precipitation) to the appropriate size. After making any necessary changes to the code, compile the program:

F77 –o read_prec_dly read_prec_dly.f
or on FreeBSD machines:
G77 –o read_prec_dly read_prec_dly.f

Run the executable program, which will prompt for the input and output files:

$> read_prec_dly

Input file:	**cmb_prcp.daily**	(output daily data file from preproc_precip.scr)
Station information file:	**cmb.inf**	(info file produced by preproc.scr)
Output file for prec time-series:	**cmb_prcp.fmt**	(formatted for regridding - will be used further)
Start and End years:	**1950 1995**	

Or the *read_precip.input* file can be editted to contain these lines, and the file can be run with :
$> *read_prec_dly < read_precip.input*
An example ,fmt output file is shown in Appendix 1.5.

2.4 *Append CD data with NCDC Climate Data Online data*

The .fmt file created in the step above can be appended with any amount of data (not limited to complete years) downloaded from the NCDC Climate Data Online web site at http://cdo.ncdc.noaa.gov/plclimprod/plsql/poemain.poe. Select "Surface Data, Daily" and then select "Continue with Advanced Options." Select the state of interest, and specify "Entire State." Then select the output option of "File." Continue onto the next page, where you have to *unselect* "All parameters" and then select PRCP, TMAX, and TMIN. Select the period of record from the menu, and leave everything else as default ("Delimited-no station names" and choose a delimiter of "Comma, with data flags"). Write down the code number on the web page after you submit the request, since the e-mail they send when the data can be picked up only references this number.

When you retrieve the data, they have a list of four files, and only the first one, with the daily data, is needed. Now this data can be processed to be in the same format as the processed CD data, and it uses the same station information file (to take advantage of the filtering capability of the CD software, which is not available on the web site). Save the data files with the extension .dly and use gzip to compress them.

2.5 *Preprocess the NCDC Climate Data Online data*

Set up a file with the names of all the downloaded data files, removing the .dly.gz from the names (similar to the use in preproc_precip.scr). For three files mo.dly.gz, ar.dly.gz, up.dly.gz, the "filename" file would look like:

```
mo
ar
up
```

Edit *preproc_append.scr* to include the correct output filename and input file extension and parameter, where indicated :

```
# INPUT FOR THIS SCRIPT###############################################
# parameter: either "prcp", "tmax", or "tmin" (use lowercase)
set PARAM = prcp
# input filename extensions
set EXT_DAT = dly
# output filename
set DAT =  ./append_tmin.daily
```

2.5.1 Reformat the NCDC Climate Data Online data

To prepare the data for regridding, compile *append_prec.f* and *append_temp.f* and edit the scripts to run them: *run_append_prec.scr* and *run_append_temp.scr*. These scripts access the station files created from the Earthinfo CD data (e.g. "basin_prcp.inf"), and the program searches through the new data for stations that match the ID numbers in the CD station info file. No new stations are added or lost in this process. The output files should be named something like "basin_append_prcp.fmt" and "basin_append_tmax.fmt". The output is identical in format to the .fmt file created from the CD data.

2.5.2 Combine the NCDC Climate Data Online with the CD formatted data

This step is only done when first creating met files for a basin. The reformatted NCDC Climate Data Online files can be gridded, rescaled, put into VIC input format independently, and appended to files already created as well. However, for initial preparation, it is simpler to just cat the files together:

cat basin_prcp.fmt basin_append_prcp.fmt >! basin_total_prcp.fmt

2.6 Precipitation Time-of-Observation Adjustment

Whether using formatted CD precipitation data, formatted data from the web site, or the concatenated files, they can be adjusted with the same program *prec_tob_adj.f.* This program accesses the library of NCDC Co-op station histories of times of observations in the file *coop_tob.his*, and adjusts the data to fall in proportionately in the current day and the previous day according to the observation time. A threshold can be specified, below which the entire day's precipitation is moved as a block to whichever day has the greater proportion of hours since the last observation. *prec_tob_adj.input* can be editted to contain all input, to look like (without the descriptive text, of course):

append_prcp.fmt	formatted data from CDs or NCDC web site
east_prcp.inf	station information file
tob/coop_tob.his	station time-of-observation history file
0	threshold below which no adjustment
append_prcp_adj.fmt	output file name
2000 1	year and month of start of data
2000 2	year and month of end of data

and then the program can be run with *prec_tob_adj < prec_tob_adj.input* . The output file has the same format as the input file, and can be input to the regridding program.

2.7 regrid

This program was written by Greg O'Donnell and Alan Hamlet, and later modified bt Andy Wood.
The program *regrid* is used to take data at irregular spacing (e.g. met. station data at the locations available) or regular gridded data at a different resolution than the grid you want (e.g. raw data comes on an 80km square grid, and you want ¼ degree latitude longitude grid). The program reads two information files, and then regrids the raw input data to a latitude longitude grid at a resolution chosen by the user (e.g. ¼ degree,

½ degree, 1 degree, etc.). The interpolation routine used is the Symap algorithm. Any number of nearest neighbors can be specified for the interpolation routine. The program searches out the nearest neighbors from the station list and compensates automatically for missing data identified by a void marker which is set at -99.0 in the input files by using the earlier preprocessing routines. Remember to keep track of what is used as void (e.g. -99.0). This is set in the **grd.f** file. Note that this program uses dynamic memory allocation, and therefore the current (as of May 1999) fortran compiler on Linux machines cannot make the program.

By using *gtar -xvf* GRID.TAR, there should be a *regrid* subdirectory, into which the files required for regridding are copied. The list of files should include:

makefile
argtest.f
chgtemp.f
grd.f
indexx.f
isaleap.f
iso_weight.f
latlong.f
near_neigh.f
regrid.runfile
regrid.trace
symap.f

It might be wise to remove any old object files before recompiling. **rm *.o** in the regrid directory. Then use the **make** command to compile the list of individual programs into one executable program, which as defined in the Makefile will be named **regrd**. The program has been developed for use on HP-UX UNIX machines, and will need modification for other operating systems (depending on the fortran compiler). It may be prudent to scan the input data for junk values prior to processing. (awk is very useful for this.)

Finally, edit the file regrid.runfile to contain the appropriate filenames and options as outlined below. The final regrid.runfile should look something like:

```
'p'                   to set p/t (precipitation/temperature) switch
4 50                  minimum number of nearest neighbors (recommended: 4) and maximum (50)
"./cmb.inf"           input station info file
"./elev_mask.cmb"     elevation mask file
"./cmb_prcp.fmt"      formatted daily value data file
"cmb_prcp.grd"        output file
0                     flag for stop for no data while running: 0=no, 1=yes
1                     output format: 0=ascii, 1=4-byte binary, 2=2-byte integer, 3=NetCDF
40                    data multiplier – used for 2-byte binary data only
```

Usage:

```
$> regrd
```

which is hardwired to look for regrid.runfile in the same directory as the executable for input variables. The NetCDF and 4-byte binary formats have not yet been developed for this program. If output format in 2-byte binary is selected, values are multiplied by the input multiplier and then written as (signed) short integers. The other programs for gridding, rescaling, and creating vicinput files also have flags to accommodate the binary output.

2.7.1 P/T Switch

The first argument is a switching variable that tells the program whether or not to lapse temperature data to the output grid elevations. When interpolating precipitation data, or if for some reason you do not want to lapse the temperature data to the new grid elevations, use "p" for the first argument. Setting the first argument to "t" adjusts the station temperature data by -0.0065 (C per meter increase of elevation. For example, if the station used for interpolation was 1000 meters higher than the destination grid cell, then 6.5 degrees would be added to the temperature at that station data before interpolation took place.

2.7.2 Nearest Neighbors

The number of nearest neighbors can be varied for the gridding program, though the number of nearest neighbors cannot be larger than the number of stations. If you attempt to specify this, the program will give an error message and stop. The program searches for the input number of minimum nearest neighbors with data from the station list and compensates for missing data, interpolating using an inverse square distance routine as described for the Symap algorithm. The original Symap publication uses 4 as a minimum number of nearest neighbors, and that is recommended here. The maximum is set only to limit array sizes, but using 50 or 60 is not unreasonable.

If too few nearest neighbors are specified, the program will return NODATA values to certain grids cells, which cannot be used in the VIC program. The goal is to specify the fewest number possible while avoiding any occurrences of NODATA in the output grid.

2.7.3 Info File

This is the same station info file produced by preproc_precip.scr. The info file contains the latitude, longitude and elevation (m) of the stations with data, as well as station names and other attributes. The stations are listed in the same sequence they are encountered by a program reading the data from the gridded input files.

2.7.4 Elevation Mask File (DEM)

The mask file contains the required information about the destination grid, and is of the Arc/Info ascii export format. The program uses this information to assign the latitude, longitude, and elevation for each of the destination cells. If the 'p' option is specified above, the gridding program will only screen for NODATA values (since actual elevations will not be used in these computations – though they will be required later for the VIC model), and therefore any grid in the proper format that delineates the basin with NODATA values outside and some value inside can function as a mask file. The example below is for the Columbia River Basin at 1-degree resolution. The E's in the example below represent the elevation values (m) of each destination grid cell, although any cell containing a non-NODATA_value would be computed.

```
ncols          13
nrows          14
xllcorner    -123.0
yllcorner      39.0
cellsize        1.0
NODATA_value    0
0 0 0 0 E 0 0 0 0 0 0 0 0
0 0 0 0 E E E 0 0 0 0 0 0
0 0 0 0 0 E E E 0 0 0 0 0
0 0 E E E E E E 0 0 0 0 0
0 0 E E E E E E E E 0 0 0
0 0 E E E E E E E E 0 0 0
0 0 E E E E E E E E E 0 0
0 0 E E E E E E E 0 0 0 0
0 E E E E E E E E E 0 E 0
0 E 0 0 E E E E E 0 E E E
0 0 0 0 0 E E E E E E 0 0
0 0 0 0 0 0 E 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0
```

2.7.5 Input File

This is the data file output from the script *read_prec_dly.scr* for a single variable from which you wish to interpolate new values for the grid specified in the mask file. This file should contain an integer number of timesteps, each containing data for each station in sequence as outlined in the section on preprocessing above.

2.7.6 Output File

This is the output file and will contain the same number of timesteps as the input file, but will contain only data for the grid cells identified in the mask file. At 1-degree resolution for the Columbia Basin, for example, the output file would contain a record of 71 fields for each timestep, with each record followed by an end of line character. Note that the units are identical to those in the input file. After this program has run there should be no void markers in the cmb_prcp.grd file if there was good data for at least one nearest neighbor. Check this using grep. If there are still some lines containing –9.0 or –99.0 run the regridding program with more stations to interpolate from. (see Appendix 1.6 for sample grid in ascii format)

2.7.7 Binary Option

Before regridding all data is in ascii format. Due to the large file sizes, 2-byte binary format has been used to be more economical with disk space. To prevent loss of precision when using the 2-byte binary option, the data is multiplied by the specified multiplier at this point before being truncated to a 2-byte (short, signed) integer.

2.8 mk_monthly

To scale the data to PRISM monthly precipitation averages the monthly means from the time series data is needed. This program reads the output from the regridding program and generates monthly values for the whole period. Only one file for each month is generated (monthly.jan, monthly.feb,…). The source is in mk_monthly.c and run_ mk_monthly.scr is the script that runs it. The elevation maskfile is needed. The script run_ mk_monthly.scr will look like:

```
# This script is used to run 'mk_monthly'
# The output directory for 'jan.monthly, feb.monthly, etc
# must be made before program execution.

set maskfile =           elev_mask.cmb
set BIG_inputfile =      cmb_prcp.grd     (output from A.H. regridding program)
set output_directory =   mnth/            (monthly.jan, monthly.feb, ..)
set start_dat =          1950             (start date of data )
set start_prd =          1961             (begin date for which means will be calculated)
set end_prd =            1990             (end date for which means will be calculated)
set bin_flag =           1                (flag=1 for 2-byte signed integer binary data)
set mult =               40               (multiplier used for the precipitation data)

mk_monthly $maskfile $BIG_inputfile $output_directory $start_dat $start_prd $end_prd
```

For the date *start_prd*, the user can choose for which period the monthly values should be calculated. If data from 1948 to 1995 are to be used for the VIC model, the period of 1961-1990 should be used since this matches the period for which prism data exists. (see Appendix 1.7 for a sample portion of monthly.jan)

2.9 Get_prism

Now we need the mask file and the PRISM raw data files as input to generate monthly prism precipitation files for comparison with the data for the basin we are gridding. These prism raw data files (can_us_prism.jan, feb, etc.) contain prism monthly mean data for both the US and Canada for 1961-1990. Data are originally from Daly et al., 1994. For more details about the datasets go to the web page:
http://www.ocs.orst.edu/prism/overview.html#application
If other raw data files are used the source code must be changed accordingly. In the source file there is an explanation of how to do this.

The script for running the program, run.get_prism, must be checked to make sure the correct mask file name is included, as well as source data filenames and output files. The script is set up to place output of monthly prism values for the basin delineated by the mask file into files prism.jan, prism.feb, …

The program starts by reading the prism data for the same area as in the mask file. All PRISM raw data files are in resolution of 2,5 minutes (i.e. 24 gridcells in one linear degree). In this program, the prism data are aggregated to the resolution specified in the maskfile by using the mean value of the PRISM grid cells in each mask grid cell. For example, the average of 36 PRISM cells are used in order to write one 0,25 degree resolution grid cell. If the elevation mask covers an area for which there are no prism data (e.g., parts of Canada) the program will write a negative value to the output file. The rescaling program will ignore this data point and does not scale the data in the grid cell where prism data are lacking. The source data should be compiled, and then run using the script:

```
$ gcc –o get_prism get_prism.c
$ run.get_prism
```

2.10 Rescale

Rescale uses the output from mk_monthly, get_prism and regrid as input. The precipitation values are read from the gridded data file (*regrid* output) and scaled with the ratio prism.jan/monthly.jan in each gridcell. This makes the long term monthly means of the gridded data consistent with the prism monthly means. The script to run the rescale program, run_rescale.scr, looks like:

```
# This script runs the rescale program

set maskfile      = elevmask.txt
set inp_monthly   = mnth/monthly.        (output from mk_monthly.c )
set inp_prism     = out-prism/prism.     (output from get_prism.c)
set in_grd        = cmb_prcp.grd         (output from regrid)
set out_grd       = cmb_prcp.rsc         (outputfile)
set starty        = 1950
set start_month   = 1
set endyy         = 1990
set end_month     = 12
set bin_flag      = 1                    (flag=1 for 2-byte signed integer binary data)
set mult          = 40                   (multiplier used for the precipitation data)

rescale $maskfile $inp_monthly $inp_prism $in_grd $out_grd $starty $endyy

# Look in source code for more detailed info about the program
```

The monthly values of both the processed data and the prism data used in the rescaling should be saved, so that when additional months are added to the met files they can be rescaled by the same factors.

3 TEMPERATURE

After concatenating the tmax and tmin station information and daily records as described above the following steps should be taken. (Note that the temperatures output files may look different compared to the precipitation files, no appendix is made for temperature).

3.1 Compressing data with GZIP

The daily temperature files should be zipped into compressed:

```
gzip mt_tmax.dly
gzip wa_tmin.dly
```

Again, don't gzip the station information files. For your own control you can word-count the station information files with wc *.sta and get the total number of stations.

Make sure that for each state (or set of states) the station information and corresponding station data files have the same prefix (e.g. for Washington use wa.sta and wa.dly), and the suffixes must be kept constant (e.g. always use sta and dly).

3.2 Preprocessing of temperature data

There are two scripts, preproc_tmax.scr and preproc_tmin.scr, for the preprocessing of the tmax and tmin data files respectively. In general, these function almost identically as the preproc_precip.scr script. Accordingly, the documentation and steps for running these scripts are similar.

As with the precipitation preprocessing, lines 40 and 41 in the preproc_precip.scr script should be checked to ensure the correct names of the output files are used. These will be the files that will be produced by this preprocessing step for the next steps in gridding.

```
40        set INFO = ./ar_tmin.inf            [output file for station information]
41        set DAT = ./ar_tmin.daily           [output file for data]
```

Also, check the extensions set in the script for the input files to make sure they match the extensions on your input files. These are set in the script on lines 43 and 44 (set EXT_DAT = dly; set EXT_INF = sta)
As before, before running the script you have to create a new file, “filenames”, with the name of the input files. Do not include the filename extensions (or the .gz), as these are set in the script as mentioned above:

```
filenames.txt
                .mt_tmin
                or_tmin
                wa_tmin
```

Now run the scripts:

```
        $ preproc_tmax.scr filenames.txt
        $ preproc_tmin.scr filenames.txt
```

3.3 Reformatting Temperature Data Files

The daily time series (both tmax and tmin use the same program at this stage) need to be reformatted into the required format for the grid program using the program *read_temp_dly* (similar to that performed for precipitation). Be sure to check number of stations and void_nr in the source code. This programs runs OK on a HP workstation. The program should be compiled, and when run prompts for the required filenames:

```
        $ read_temp_dly
        Input file:
                cmb_tmax.daily              (output from preproc_tmax.scr and preproc_tmin.scr)
        Station file
                cmb_tmax.inf                (station info file produced in preprocessing)
        Output file for temperature data:
                cmb_tmax.fmt                (output file, will be used in next steps)
        Start and End years
                1950 1995
```

3.4 Appending data from NCDC Climate Data Online

As with precipitation data, the additional data available on the web can be prepared and appended to the formatted CD data, or regridded on its own. See section 2.4 above for details

3.5 regrid

As with the precipitation data processing, the data must now be regridded. The temperature data is reformatted using the same program as for precipitation. See the section above for documentation on how to compile the program, and check for desired input and output filenames. The program has as output regridded temperature files, e.g., cmb_tmax.grd, cmb_tmin.grd. If the binary option is selected for precipitation data, it must also be selected for temperature data. The multiplier does not have to be the same, however (recommended value of 100).

3.6 Rescaling Temperature to PRISM Data

Modifications to existing programs are ongoing to allow rescaling of the temperature data to PRISM data averages, as is currently done with precipitation data. The programs are not available yet, however

4 VIC INPUT

This program uses the rescaled precipitation file and the gridded temperature files and puts them into separate preliminary VIC input files. The output files are named, for example, data_41.83_-119.62 (1/4 degree), or data_45.4375_-97.4375 (1/8 degree) and are opened in write mode. Filenames describe cell centers. Currently, the vicinput.c program is set to 4 digit precision on filenames. In the function make_flist change the fprintf lines to use %.2f to create filenames with 2 decimals, for example. The program then reads 50 timesteps at a time from the input files (precipitation, tmin, tmax) and converts them into VIC input files. The number of files should be the same as number of gridcells in the elevation maskfile minus the number of void gridcells.

The source file is called vicinput.c and the script to run it is named run_vicinput.scr. The program is originally compiled and tested on meter.

```
# Runs the vicinput.c
set prcp=       /nfs/hydro4/usr3/tempbvm/qd-model/data/prcp/cmb_prcp.rsc
set tmax=       /nfs/hydro4/usr3/tempbvm/qd-model/data/tmax/cmb_tmax.grd
set tmin=       /nfs/hydro4/usr3/tempbvm/qd-model/data/tmin/cmb_tmin.grd
set e_msk=      elev_mask.txt
set out_dir= /nfs/meter/scratch/tempbvm/vicinput/
set binflag =  1             (again, binary flag of 1 means 2-byte format)

vicinput $prcp $tmax $tmin $e_msk $out_dir $binflag
```

5 Check the Data

The program *check.c* reads one of the output files (data_41.88_-119.62) and generates monthly means, max, min, SD, etc for precipitation as well as minimum temperature and maximum temperature.

Pick a station which is close to center of a gridcell and plot the monthly precip, tmin and tmax from the station against the gridded cell data. Remember that the gridcell might be on another elevation than the station. The temperature lapse rate used is 6.5 C/km. You should be able to see this on the plots.

Do this for a couple of stations. This is a good control of the whole methodology. GMT plots could also be useful.

6 NCEP/NCAR Reanalysis Wind Data

The VIC model can also use daily wind forcing data, which is easily added to the met files. The source of the wind data is the Reanalysis archive, which begins in 1948 and is updated regularly. You can read about this data source, and check available data at: http://www.cdc.noaa.gov/cdc/reanalysis/reanalysis.shtml
There is a two step process to creating the wind files – obtaining the data, and regridding it.

6.1 Getting the raw data from NCEP

Run the getwind.scr program (which must currently be run on HP-UX or FreeBSD to use ncks, see the flag in getwind.scr). The script must be editted to define a latitude-longitude box as large or larger than the mask file for the basin. Set the output directory, and the script will retrieve the raw data via ftp and clip it to the defined area, leaving one uwind_<year>.asc and one vwind_<year>.asc file for each year. If you are only interested in a couple of months, just retrieve the year(s) of interest at this point and the next step will sort it out.

6.2 Regridding wind data

The regrid_wind.c program is currently hardcoded to produce 2-byte binary data with a multiplier of 100. If other multipliers are desired or ascii format is used the program must be modified. Also, it is set to produce 4-digit filenames for the wind files.

The running script, run_regrid_wind.scr, takes as input the path to the raw data retrieved in the prior step, the location of the mask file, the path for the output files, and the same latitude and longitude limits defined in the previous step (they MUST be the same). The output is a set of files similar to the vicinput files, named something like wind_35.4375_-110.6875.

If the wind data is part of an update (of one or more months) to be appended to existing VIC forcing files, be sure to specify the same period of months for the wind data as is used for the precipitation and temperature. Since, at least up until now, Reanalysis data is available at least two months beyond what NCDC has in its on-line database, wind data should always be available through the end of the other met data.

7 Combine the wind data with precipitation and temperature data.

The program combine_wind.c is run with the script run_combine_wind.scr and reads the VIC files with precipitation and temperature and essentially adds the wind data as a fourth column. It also is currently set to transform the precipitation from a signed to an unsigned 2-byte integer. This program is only set to accept 2-byte binary input and output. This produces the final set of data_<lat>_<lon> files for running VIC.

7.1 Appending monthly updates to existing met files

The combine_wind.c program also can be used to append any number of months of data by specifying an append_flag of 1 along with the locations of the data_<lat>_<long> and wind_<lat>_<long> files containing the data for the month(s) to be appended, as well as the directory with the existing forcing files. If this append flag is not correctly set it can result in all of your original files being overwritten. It is wise to back up your original forcing files regularly!

8 References

Daly, C., R.P. Neilson, and D.L. Phillips. 1994. A statistical-topographic model for mapping climatological precipitation over mountainous terrain. Journal of Applied Meteorology, 33, 140-158.

Donald S Shepard
Computer mapping: The SYMAP interpolation algorithm
Spatial statistics and Models 133-145
1984 by Reidel Publishing Company.