Installing and Running the MAPSS model (Mapped Atmosphere-Plant-Soil System)

Contact: Ray Drapek
PNW Research Station
3200 SW Jefferson Way
Corvallis, Oregon, USA
97331
email: drapek@fsl.orst.edu
phone: (541) 750-7483

Introduction

NetCDF Files

All climate and ancillary data files are in NetCDF format. For more information on NetCDF look up http://www.unidata.ucar.edu/packages/netcdf. From this website: "NetCDF (network Common Data Form) is an interface for array-oriented data access and a library that provides an implementation of the interface. The NetCDF library also defines a machine-independent format for representing scientific data. Together, the interface, library, and format support the creation, access, and sharing of scientific data." Version 3.6.1 of NetCDF was used.

Sparse Versus Full Grid Format

As a space saving device, many of the files have been stored in sparse grid format. The full gridded region of any dataset includes many mapcells that are not modelled (example: ocean mapcells). Rather than store many no-value code numbers for all these empty mapcells, a sparse grid file only stores the data for those mapcells that are modelled. Data are stored in the order they appear by row (top to bottom and left to right). An additional file is needed to translate these sparse grid data to full grid when needed. This file has data for all rows and columns. Off-grid mapcells are indicated with zeros and on-grid mapcells are indicated with positive integers. For MAPSS the file "gridpnt.nc" fills this purpose. It is described further in the "Ancillary Files" section.

This distribution contains
1) MAPSS code files:

at_flag.c
bfuncs.c
c3c4_functions.c
classification.c
distrib.c
drain.c
fire.c
grasses_equil.c
io_general.c
lifeform_rules.c
mapss_1t.c
mapss_alarms.c
mapss_equil.c
mapss_main.c
misc_init.c
nc_input.c
nc_output.c
one_step.c
pet_calc.c
pm.c
qdips.c
read_site_parm.c
satvp.c
station_out.c
swp.c
transpire.c
trbxfr.c
wbcalc.c
woody_equil.c
at_flag.h
bfuncs.h
c3c4_functions.h
c3c4_functions_static.h
category.h
classification.h
classification_static.h
distrib.h
drain.h
drain_static.h
exit_codes.h
file_paths.h
fire.h
fire_static.h
global_extern.h
global_vars.h
grasses.h
grasses_static.h
help_text.h
io_general.h
lifeform_rules.h
lifeform_rules_static.h
mapss_alarms.h
mapss_bin.h
mapss_common.h
mapss_defines.h
mapss_equil.h
mapss_funcs.h
mapss_main.h
mapss_out_month_macros.h
mapss_out_year_macros.h
mapss_switches.h
mapss_types.h
mapss_vars.h
misc_init.h
nc_input.h
nc_output.h
one_step.h
output_var_info.h
pet_calc.h
pet_calc_static.h
physdefs.h
physmacs.h
qdips.h
read_site_parm.h
satvp.h
station_out.h
swp.h
swp_static.h
transpire.h
transpire_static.h
trbxfr.h
trbxfr_static.h
wbcalc.h
wbcalc_static.h
woody.h
woody_static.h

2) The makefile: Makefile.

3) Example parameter files: site, parameters, and fire_param.dat.

4) Example ancillary data files (input files for running MAPSS over the conterminous 48 states of the USA at a ½ degree latitude by ½ degree longitude gridcell resolution): elev.nc, soils_scs.nc, and gridpnt.nc.

5) Example climate data files (input files): ppt.nc, vpr.nc, tmp.nc, and wnd.nc.

6) Example filter file: out_filter.new.

7) Example output files: mapss_out_month.nc, mapss_out.nc.

Command Line MAPSS Call

MAPSS can be run command line. An example command line is given at the end of this document.

Command line options (these can be seen by typing "mapss -h"):

```
 mapss [a:c:C:D:e:f:hI:o:p:P:r:s:Suvz]
      filename          If only argument, read command switches from file
      -a<scenario>        Alternate data (climate scenario)
      -c col1,col2       Begin processing with col1, end with col2
      -C<m|p>            Calculate PET (Marks, Penman-Monteith)
      -D path           Path to input climate data root directory
      -e filename        Error log file name
      -f filename       Name of output filter file
      -h               Help
      -I path           Directory to search for parameter files
      -o filename        Output file name
      -p filename        Parameter file name
      -r row1,row2        Begin processing with row1, end with row2
      -s filename       Site file name
      -S               Set Southern hemisphere <read gridpnt.nc file>
      -P dataset         Use netCDF input <dataset> files
      -u               Print output variables available
      -v               Version info and column headings
      -z               Use the grass pet when grass alone
```

Input Files Needed to Run MAPSS

Types of files are needed to run MAPSS:
1) Parameter files.
2) Ancillary data files.
3) Climate files.
4) Filter file.

Examples of all these files are included with this distribution.

Parameter Files

There are three parameter files used: "parameters", "site", and "fire_param.dat". These are ASCII text files.

- The particular parameter files to be used in any MAPSS run are indicated in the command line with the "-I" parameters.
- Following the –I should be a string showing the path to the directory where the "parameters" and "site" files can be found.
- The location of the "fire_param.dat" file is hard-coded into the MAPSS code and is set in the file_paths.h file.

Parameters found in all three files are numerical values. Wherever the first column is the start of a numerical value, MAPSS interprets that value as a parameter. Following the parameter value is a parameter name and often some comments. These are there for the benefit of MAPSS program users. MAPSS code ignores them. A "#" is used to indicate comment lines. Blank lines are ignored. For all three parameter files you must have the correct number of entries.

The "parameters" file. The "parameters" file holds parameters controlling a wide range of functions from transpiration calculations to vegetation classification. A "-p" can be used on the command line to indicate the parameters file separately from the site file. Most parameters are explained in Appendix 2 of Neilson (1995).

One parameter that bears special note is the "wue" parameter. Actually there are multiple wue parameters, each for different climate zones and vegetation types. The name wue stands for Water Use Efficiency. The reason these parameters bear special note is they are used in MAPSS for simulating an ecosystem response to enhanced carbon levels. MAPSS does not model photosynthesis and so we cannot directly model a CO2 response as an increase in carbon fixed. But increased CO2 levels are also known to bring about increases in water use efficiency and MAPSS does simulate water use. When running MAPSS for scenarios where CO2 levels were doubled over historical levels, the MAPSS research team set the wue parameters to 0.65. When running over historical CO2 levels, we set the wue parameters to 1.0.

The "site" file. The site file contains parameters affecting soil functions. A "-s" can be used on the command line to indicate the site file separately from the parameters file.

If you look at the "site" file, you will see that the parameter "WhichSoils" is set to "2", which means scs soils data. When it is set this way MAPSS looks for a soils file called soils_scs.nc, when it is set to "1" MAPSS looks for soils data in the file soils_fao.nc. Alternatively, setting "WhichSoils" to 0 means you don't have to include a soil data input file. MAPSS then will assume that all map-cells are a sandy loam.

The "fire_param.dat" file. This file contains parameters determining the way fire behaves in MAPSS. This file is not subject to change from one run to the next and can in effect be treated as part of the hard MAPSS code. Its location is hard wired for MAPSS in the file_paths.h file (constant FIRE_MODEL_PARMS) and there are no command line parameters for changing this location.

Ancillary Data Files

The ancillary files hold all the non-climate information needed to describe the region we are simulating over. These files are all in NetCDF format and may be either full grid or sparse grid. The names used in this section are what MAPSS is expecting for these files. Examples of each of these were included with this distribution.

The location of the ancillary files is indicated using the "-P" command line parameter. MAPSS is hard coded to look for data in a certain directory. You can see this in the file "file_paths.h" which is included in this distribution. The constant NETCDF_DATA_PATH is used to define this path. Users of this code will need to change this to something more appropriate to their own computer. The name on the command line after the "-P" will be a sub-directory under this base directory.

The example data are for the region bounded by 49N, 25N, 124.5W, and 67W. It is a 48 row by 115 column grid with ½ degree resolution in latitude and longitude. The ancillary files include:

elev.nc - Elevation is measured in meters. These are sparse grid files and are dimensioned as grid x band, where grid is the number of sparse-grid mapcells and band dimensioned as 1. The band dimension is superfluous, but is required by the MAPSS code.

gridpnt.nc - Full grid. Dimensioned as rows x columns. Off-land mapcells are indicated with zeros. On-land mapcells are positive integers sequentially numbered from one. This way every modeled gridcell is given a unique identifying number. This file is used to translate between full and sparse grids.

soils_fao.nc - Soil data.
Sparse grid data dimensioned as grid x band, where there are 10 "bands":

0=mineral_depth (measured in mm),
1=sand[surface](measured in %),
2=sand[intermediate](%),
3=sand[deep](%),
4=clay[surface](%),
5=clay[intermediate](%),
6=clay[deep](%),
7=rock_fragment_mineral[surface](%),
8=rock_fragment_mineral[intermediate] (%),
9=rock_fragment_mineral[deep](%).

The three soil depths referred to here are surface (0-0.5 meters), intermediate (0.5-1.5 meters), and deep (greater than 1.5 meters deep).

The name "soils_fao" refers to the fact that for our MAPSS run we stored data from the U.N. Food and Agriculture Organization in this file. If you use this file for your soil data you need to set the "WhichSoils" parameter in the "site" file to "1".

soils_scs.nc - Alternate soil data file.

Same format as soils_fao.nc. We used this format for soil data we picked up from the Soil Conservation Service. If you use this file for your soil data you need to set the "WhichSoils" parameter in the "site" file to "2".

Climate Files

All climate files are sparse grid NetCDF files dimensioned as grid x month. The number of months is always 12 since MAPSS runs on average climate (i.e. average climate for January, average climate for February, etc.).

The climate files include:

ppt.nc - Cumulative monthly precipitation in mm.

tmp.nc - Mean monthly temperature in degrees C.

vpr.nc - Mean monthly vapor pressure in pascals, and

wnd.nc - Mean wind speed in meters/second.

The variables accessed in these files also are called ppt, tmp, wnd, and vpr. MAPSS looks for an attribute for these variables called "scaled". The scaled attribute value is used to modify all data values within the file. So that if the scaled attribute for the temperature file is 100.0, then all values from that file are divided by 100 to yield the actual temperature.

The location of the climate files is indicated using the "-a" command line parameter. The name following the -a is a subdirectory that is located under the ancillary data directory. Since for any region over which we ran the MAPSS model there were often multiple climate scenarios, the data for these scenarios were stored in multiple sub-directories under the ancillary data directory.

Filter File

The filter file tells MAPSS what model variables to output. The filter file is simply an ASCII text listing of output variables, one per line. An example of a filter file included with this distribution is "out_filter.txt". Lines beginning with”#" are ignored by the MAPSS code and can be treated as comment lines. Blank lines are also ignored. The filter file is indicated in the MAPSS command line call with the "-f" parameter. Following the "-f" should be the filter file (full path included). The version of MAPSS being distributed here can potentially output 143 variables, but most users would only want a few. Type "mapss -u" once you have MAPSS properly compiled and it will give you a list of all the potential output variables.

Compiling MAPSS

The version of MAPSS included in this distribution was compiled and run on Sun Microsystems Ultrasparc sun4u machines running on the Solaris 8 operating system. The C compiler was the Sun Workshop Compiler C 5.0. A makefile is provided and MAPSS can be compiled by typing “make all”.

Steps to Getting MAPSS Operational

To get MAPSS functional you will need to:

1) Alter NETCDF_DATA_PATH in file_paths.h so that it points to the directory where you intend to store all the climate and ancillary data files.

2) Also in file_paths.h, alter FIRE_MODEL_PARMS so that it points to the directory where you store the fire parameters file (fire_param.dat).

3) Make sure all the /local/include/*.h files in the file depend.cc are available.

4) Create a subdirectory under your NETCDF_DATA_PATH where you will put your ancillary data files. This is the dataset directory. Since these generally describe a geographic region, the directory name generally reflects the region. Example: "usa_48_states".

5) create one (or more) subdirectories under the dataset directory where you will store the climate data. This is the scenario directory. These generally are named "historical" or are named for the climate scenario being simulated. Example: "hadcm3_sresa2".

6) Place your parameter and site file in an appropriate directory (user choice).

7) Create a filter file and place it in an appropriate directory (user choice).

8) Modify Makefile so that compiler location, the NetCDF library, etc. are correctly indicated.

9) Compile and run.

Example Command Line Run

An example run of MAPSS (as done on the MAPSS team computer):

> mapss -Cm -c 0,114 -r 0,47 -I /data/mapss/parameters -P usa_48_states -a historical -f

> /data/mapss/outputs/out_filter.txt -o /data/mapss/outputs/testrun.nc.

Breakdown:

"-Cm": This tells MAPSS to use equations derived by Danny Marks to calculate Potential Evapotranspiration. An alternative would be to type "-Cp", in which case MAPSS would use the equations of Penman-Monteith to make the PET calculation. This would require solar radiation (in 1000.0joule/m2/day) as an additional climate input. Marks PET calculations are suggested for MAPSS.

"-c 0,114": This tells MAPSS to run over columns 0 to 114.

"-r 0,47": This tells MAPSS to run over rows 0 to 47.

"-I /data/mapss/parameters": This tells MAPSS that the site and parameters files will be found in the directory /data/mapss/parameters.

"-P usa_48_states": This tells MAPSS that the ancillary data will be found in a directory called "usa_48_states", which is directly under the directory identified by the constant NETCDF_DATA_PATH in the MAPSS code.

"-a historical": This tells MAPSS that the climate data will be found in a directory called "historical" which will be right under the ancillary data directory.

"-f /data/mapss/outputs/out_filter.txt": This tells MAPSS that the filter file we will use is called /data/mapss/out_filter.txt (note: the full path is defined for this).

"-o /data/mapss/outputs/testrun.nc": This tells MAPSS to send output to the file called /data/mapss/outputs/testrun.nc (note: the full path is defined). This output file will be a NetCDF file and will include all of the variables defined in the filter file. If no “-o” parameter is defined in the command line, then MAPSS defaults to sending annual output to a file called “mapps_out.nc” and monthly output to a file called “mapps_out_monthly.nc”

CITATION

Neilson, R.P. (1995) A model for predicting continental scale vegetation distribution and water balance. Ecol. Appl. 5: 362-385.

(Neilson_Ecol_Apps_1995_A2.pdf)