Nodal Attribute File Builder Version 8.3

Taylor Asher

1 Introduction

This document covers version 8.3 of the tool[1]. The program, f13builder.bash, acts to interface with a series of Fortran utilities for generating individual nodal attributes. It then assembles these individual attributes, along with the requisite information, to make a single fort.13 file. The program has been developed on a Linux platform with the Bash shell language, however it may work with other shells. This has not been tested. A utility to convert individual attributes in the fort.13 file to maxele.63-style files is also supplied in the "check" directory.

2 Attributes

Currently, the following attributes can be generated:

- mannings_n_at_sea_floor (abbreviated mnasf) via the mannings_n_finder_v11auto.f program
- primitive_weighting_in_continuity_equation (abbreviated pwice) via the tau0_genauto.f program
- sea_surface_height_above_geoid (abbreviated sshag) via the sshagiderv01.f90 program
- surface_canopy_coefficient (abbreviated scc) via the surface_canopy_v9auto.f program
- surface_directional_effective_roughness_length (abbreviated sderl) via the surface_roughness_calc_v23auto.f program
- surface_submergence_state (abbreviated sss) via the sssiderv02.f90 program
- wave_refraction_in_swan (abbreviated wris) via the wrisoutabox.f90 program

This program is available on the ADCIRC website. The pre-existing Fortran programs upon which this code is based were modified for interface with f13builder.bash. Because the program is based on the existing ADCIRC utilities, the inputs match those already in use in the respective programs. The operation of these programs, therefore, is not explained here.

3 Land-Cover Dependent Programs

The programs which are based on land cover datasets such as the NLCD and CCAP (mannings_n_at_sea_floor, surface_directional_effective_roughness_length, and surface_canopy_coefficient), have all been generalized to accept an arbitrary land cover dataset. The user must provide a conversion table between land cover values and nodal attribute values. The program package comes with conversion tables for NLCD and CCAP data. The NLCD conversions come from those already present in the utilities on the ADCIRC website. The CCAP conversion tables are provided through work done under multiple studies. The structure of the conversion tables is provided in Section 7. **IMPORTANT**: The CCAP land cover algorithm has changed, which notably affects land cover classifications. *The mappings provided with this code may no longer be as accurate*. More notes on this in Section 12.

[1] Version 8 is not strictly backwards-compatible with previous versions due to a flag added for wris. Previous input files therefore may not work.

4 Custom Fortran Programs

The wrisoutabox.f90, sssiderv02.f90, and sshagiderv01.f90 programs were created for this purpose, and so are explained here.

The wrisoutabox.f90 program operates by being supplied the lower-left and upper-right corners of a box. It then finds all the nodes not within this box (including those lying on the edges) and provides them to be listed in the fort.13 file.

The sssiderv02.f90 program operates by being supplied one or more "seed" locations (x/y coordinates) of areas that should be considered "wet" (deeper than a specified value) when the simulation starts, as well as the elevation below which should be considered wet. An example of this would be a point in the Atlantic Ocean. The program then determines all nodes connected to the seed location(s) that should also be considered "wet", all nodes connected to those nodes, etc., until the basin is fully identified. This is done for each seed location so that multiple separate basins can be used. Once this is complete, all locations that would be considered "wet" at simulation start time but are not connected to one of the identified basins, i.e. isolated wet nodes, are output as the list of nodes that need to be set to start dry.

The sshagiderv01.f90 program operates in a manner very similar to the sssiderv02.f90 program, except that instead of finding all isolated "wet" nodes, this program is designed to all nodes that should be set to a specified starting wet elevation using a seed x-y coordinate pair and a water elevation. Multiple seeds may be specified, so that basins that are not hydraulically connected (such as an ocean and land-locked lake) can be specified separately with their own elevations. The program proceeds by starting with the highest elevation, therefore if any seeds are found to be hydraulically connected, the highest elevation is specified.

5 Program Usage

To run the program, the user must first compile all of the Fortran programs in the "src" directory, placing the output executables in the "bin" directory. The "compile.sh" file in the "src" directory should perform this automatically, it only requires the name of the Fortran compiler to be used as an input on the command line. Naming conventions in the "bin" directory must be maintained for f13builder.bash to work properly. For reference, here are the source (left) and executable (right) file names:

mannings_n_finder_v11auto.f	mnasf
sshagiderv01.f90	sshag
sssiderv02.f90	sss
surface_canopy_v9auto.f	scc
surface_roughness_calc_v23auto.f	sderl
tau0_genauto.f	pwice
wrisoutabox.f90	wris

It is recommended to copy the entire directory into the project directory. Then, from within the directory, make (or modify the existing) input file (default name f13builder.in) and execute the program with the statement

./f13builder.bash f13builder.in

It is necessary that the user be in the build directory because the main program needs to know the location of its associated executables.

6 Input file structure

Text in *italics* are informational, text in **bold** are parameters in the file.
General:

- **filfort13** – fort.13 (nodal attribute) output file name
- **filfort14** – fort.14 (mesh) file name
- **fillc** – land cover file name (can be a dummy name if no land cover is used)
- **nattr** – number of nodal attributes
- *for cnt1=1..nattr*
    - **curnamattr** – name of current attribute
    - *parameters of current attribute (may be multiple lines)*

For specific attributes:

- mannings_n_at_sea_floor
    - **fillc2mnasf** – file containing the table converting land use codes to mnasf values
    - **diflcmnasf difdefmnasf** – whether a different land cover dataset should be used rather than **fillc** (y=yes n=no), and whether a different default value should be used (y=yes n=no, default is 0)
    - *if diflcmnasf=y*
        - **fillcmnasf** – land cover file name for mnasf
    - *if difdefmnasf*=y
        - **defmnasf** – default value for mnasf
- primitive_weighting_in_continuity_equation
    - *none*
- surface_canopy_coefficient
    - **fillc2scc** – file containing the table converting land use codes to scc values
    - **diflcscc** – whether a different land cover dataset should be used rather than **fillc**, y=yes n=no
    - *if diflcscc=y*
        - **fillcscc** – land cover file name for scc
- surface_directional_effective_roughness_length
    - **sderlrmax** – max distance to find points (line 3 in v16 surf_rough.in)
    - **sderlrw** – weighting distance (line 4 in v16 surf_rough.in)
    - **sderlcalcmode** – whether to do sector (1) or linear (2) calculation of sderl coefficients
    - **fillc2sderl** – file containing the table converting land use codes to sderl values
    - **diflcsderl** – whether a different land cover dataset should be used rather than **fillc**, y=yes n=no
    - *if diflcsderl=y*
        - **fillcsderl** – land cover file name for sderl
- sea_surface_height_above_geoid
    - **nseedsshag** – number of seed coordinates for sshag (set to 0 to just supply a default value)
    - *if nseedsshag>0, for cnt2=1..nseedsshag*

        - **seedxsshag seedysshag wetelelimsshag** – x-y coordinates of seed location and wet elevation value to assign
    - **difdefsshag** – whether to use a different default value for sshag (0 is the default), y=yes n=no
    - *if difdefsshag=y*
        - **defsshag** – new default water elevation offset for sshag; note that this is an elevation, NOT a depth
- surface_submergence_state
    - **nseedsss** – number of seed coordinates for sss
    - *for cnt2=1..nseedsss*
        - **seedxsss seedysss wetelelimsss** – x-y coordinates of seed location and wet elevation value to use for marching
    - **difdefwetelelimsss** – whether a different default wet/dry cutoff value is specified and how it should be handled.
      y=a wet/dry cutoff is specified in this file (on the next line)
      n=a wet/dry cutoff is not specified and the default value of 0 is used
      sshagdef=the default value of sshag should be used as the cutoff (requires specification of sshag)
    - *if difwetelelimsss=y*
        - **defwetelelimsss** – elevation above which nodes are considered dry (nodes equal to this elevation are wet); note that this is an elevation, NOT a depth
- wave_radiation_in_swan
    - **nwrisvert** – flag for vertex count; a value of -1 indicates the classic lower-left and upper-right corners of a box are to be used, no other option is accepted at this time, other values not accepted at this time
    - *if nwrisvert=-1*
        - **wrisx1 wrisy1 wrisx2 wrisy2** – lower-left (x1,y1) and top-right (x2,y2) coordinates of box bounding the area outside which wave refraction is disabled

7 Conversion table file structure

Text in *italics* are informational, text in **bold** are parameters in the file.

- **nlc** – Number of entries in the conversion table
- *for cnt1=1..nlc*
    - **lcval attrval** – land cover value and corresponding nodal attribute value

8 Variables

8.1 f13builder.bash

- cur* = “current” (in a programming loop sense) value of a given variable
- def* = default value(s) to use; attributes include mnas, pwice, scc, sderl, sshag, sss, wris
- do* = whether to do specified nodal attributes (0=no); attributes include mnas, pwice, scc, sderl, sshag, sss, wris
- fidfort14 = file identifier used for fort.14 (mesh) file
- fidin = file identifier used for main input file

- filfort13 = name of fort.13 (nodal attribute) file (read from main input file)
- filfort14 = name of fort.14 (mesh) file (read from main input file)
- filin = name of main input file to program, as supplied at the command line
- fillc = name of the land cover dataset file
- fillc* = name of the land cover dataset file for the specified nodal attribute; attributes include mnas, scc, sderl
- fillc2* = name of the file containing a table to convert the land cover data to the appropriate attribute; attributes include mnasf, scc, sderl
- fort14file = name of fort.14 (mesh) file
- nam* = name of nodal attributes
- nattr = number of attributes, specified in input file
- ne = number of elements in mesh (read from mesh file)
- nn = number of nodes in mesh (read from mesh file)
- nnd* = number of nodes not matching their default value for a given attribute; attributes include mnasf, pwice, scc,sderl, sshag, sss, wris
- vpn* = number of values per node; attributes include mnas, pwice, scc, sderl, sshag, sss, wris
- wrisx1 = x-coordinate of bounding box's lower-left corner for wris
- wrisy1 = y-coordinate of bounding box's lower-left corner for wris
- wrisx2 = x-coordinate of bounding box's top-right corner for wris
- wrisy2 = y-coordinate of bounding box's top-right corner for wris

9 Check Utility

A utility, f13toattr63.bash, to convert individual attributes in the output fort.13 file (or any fort.13 file) is provided in the "check" directory. This program converts the specified nodal attribute to one or more maxele.63-formatted files, which are more easily reviewed. The program relies on the Fortran inflate executable, which the user must compile in the same directory. The program can then be called as follows:

./f13toattr63.bash **filin filoutpat nanam**

Where

- **filin** – name of input fort.13 file
- **filoutpat** – base name of output file (NOT a file path)
- **nanam** – nodal attribute name as it appears in the fort.13 file; using --all tells the program to extract all the nodal attributes to files, in which case the attributes' actual names are used as the base names instead of the **filoutpat** variable

10 Version History

10.1 V6 2012

First public release.

10.2 V7 2013-11

Version 7 is not strictly backwards-compatible with previous versions due to improvements in handling of sshag and sss. Provided enhanced capabilities to sss and sshag so they can use multiple starting elevations (disconnected basins).
Improvements to error catching and handling (primarily on inputs).

10.3 V8.X

10.3.1 V8.0 2014-04

Supplied sshag with the ability to just set a default value and nothing else.
Added framework for future capabilities in wris.
Added to the check utility the ability to automatically process all attributes in the fort.13 file.
Changed program so that it only checks for the default land cover file if a spatial attribute is supplied (previous versions claimed to have this ability, but didn't).

10.3.2 V8.2 2018-10

Fixed bugs in mannings_n_finder, surface_canopy, and surface_roughness_calc Fortran routines that could cause out-of-bounds array references.
Fixed bug in f13builder.bash that led to an error if there wasn't a land cover-dependent nodal attribute specified.

10.3.3 V8.3 2021-08

Added automated logging to compile.sh
Updated the algorithm used in generating surface_canopy_coefficient. The new algorithm was devised by John Ratcliff and ported to surface_canopy_v9auto.f by Shintaro Bunya. It trims the fringes of the extents of canopies. This update aims to avoid placing canopies over meteorological observation stations, which are often located in open spaces adjacent to forested areas (e.g., airports) and can be erroneously covered by canopies due to insufficient spatial resolution. The changes can be identified in the program source file by searching "sb20210624".

11 Potential Future Features

Make code gfortran compatible. NOT() expects an integer, and it seems that converting between logical and integer data types isn't easy. Probably need to rewrite code to address this. The internet indicates that, instead of doing .NOT.(a), to do a.eq.0, but I've not tested.

Allowance for generic nodal attribute. This would require that the attribute name, default value, number of values per node, and unit be specified in the main input file, along with the name of a file containing the number of nodes with the non-default value followed by these nodes listed.

Allow for mapping land cover codes to NLCD land cover codes (previously hard-wired into land cover-dependent attribute Fortran codes), thereby allowing the user to not need to specify multiple conversion tables for the different nodal attributes and simplifying the mapping decision process.

Let the user begin with an existing fort.13 file and only recreate/add/remove desired attributes.

Allow for ranges in land cover table (e.g. map land cover codes 1000-1999 to a single Manning's n).

12 Note on CCAP Land Cover Parameter Mappings

Taken from an email from Taylor Asher to the ADCIRC listserv December 10, 2015:

> Many of you might use NOAA's C-CAP land cover data to determine Manning's n, surface roughness, and other nodal attributes. I do, it's a great dataset. But be wary of a not-so-obvious change that recently caused me some headaches. In the most recent update, NOAA modified the algorithm, which affected how different land cover values were assessed. They also went back to the prior CCAP datasets (2006 at least) and re-evaluated those using the new algorithm. Quoting the metadata:
> https://coast.noaa.gov/dataservices/Metadata/TransformMetadata?u=https://coast.noaa.gov/data/Documents/Metadata/Imagery/harvest/CCAP_2006_Parent.xml&f=html
> "During the 2010 land cover mapping, significant changes were made to the Developed and Wetland classes, resulting in large changes and improvements in all dates of land cover."
> In other words, an area that appears completely the same could have a pretty significant change in what C-CAP classification it's given. And downloading the 2006 dataset now would give you a different dataset than what you would've gotten had you downloaded it a couple of years ago.
>
> For my current study area, that meant that there was an apparent **640%** increase in the amount of evergreen forest and a **140%** increase in the amount of developed open space simply due to the change in the algorithm. Meanwhile, the apparent amount of high- and medium-intensity developed areas reduced by **36%** and **28%**, respectively, again solely due to the algorithm change.
>
> Since mapping land cover values to Manning's n and other nodal attributes is typically done by reviewing the land cover values and how the correspond to imagery, that means that the mappings now need to be recalibrated to represent the current C-CAP classifications. That may also mean that the default parameter mappings that exist in current ADCIRC utilities are no longer appropriate, so caution is advised.