WRF-Hydro® MODELING SYSTEM

WRF-Hydro V5.1.1 Test Case User Guide

Creation Date: January2020
WRF-Hydro Development Team

Purpose

This example test case is meant to orient you to running the WRF-Hydro modelling system using prepared geographical inputs, and prepared forcing data for a specific region (domain). We provide baseline instructions to test that your WRF-Hydro executable was compiled correctly and is able to run on your system. Please see the Readme.txt file provided with the test case for specific file descriptions and a regional description.

This test case provides example namelists and prepared geographical input files for three routing configurations. While the step-by-step walkthrough instructions here guide you through running the Gridded configuration we suggest that you explore the other routing configurations and related namelists for reference.

For a detailed technical description of WRF-Hydro and instructions on how to run WRF-Hydro in standalone mode see the *WRF-Hydro V5 Technical Description* document and *How to Run WRF-Hydro V5 in Standalone Mode* user guide available from https://ral.ucar.edu/projects/wrf_hydro.

For further information on the WRF-Hydro modeling system and possible training opportunities, please visit (https://ral.ucar.edu/projects/wrf_hydro).

Requirements

Linux operating system or Docker

- WRF-Hydro source code > v5.1.1
  https://ral.ucar.edu/projects/wrf_hydro/model-code
- An official WRF-Hydro example test case > v5.1.1
  https://ral.ucar.edu/projects/wrf_hydro/testcases
- All system libraries needed by the WRF-Hydro modelling system can be found in the *How To Build & Run WRF-Hydro V5.1.1 in Standalone Mode* user guide and the FAQ web page located at https://ral.ucar.edu/projects/wrf_hydro

Step by step walkthrough

Directory structure setup

We will organize all files and folders under a common top-level directory to simplify commands in this walkthrough. All paths mentioned in this walkthrough will be relative to this top-level directory. For example, /home/user/project_directory/example_case/ will be referred to as example_case/

Linux operating system or Docker

1. Open a terminal window
2. Create a top-level directory that will hold all subdirectories and files used for this walkthrough. Hereafter referred to as the 'project directory'.
3. Copy the uncompressed WRF-Hydro source code into the project directory created in step 2.
4. Copy the example case directory into the project directory as well.
5. An example of your project directory structure using the Croton, NY example test case would look like the following:

```
-project_directory
    |___wrf_hydro_nwm_public*/
        |___trunk/
            |___NDHMS/
    |___example_case/
        |___FORCING/
        |___Gridded_no_lakes/
        |___NWM/
        |___Reach/
        |___supplemental/
        |___Gridded/
            |___DOMAIN/
            |___RESTART
            |___referenceSim/
```

Compiling the model code

This section will walk you through compiling the WRF-Hydro model code for running the Gridded configuration.
1. Navigate to the WRF-Hydro source code directory at wrf_hydro_nwm_public*/trunk/NDHMS
2. Configure the model compilation environment. For this walkthrough we will assume you are using the Linux gfort compiler. If you are using a different compiler select the appropriate option for your system.
   `./configure 2`
3. Next we will compile the model using the NoahMP land surface model (LSM). Compile-time options are set using environment variables using a supplied file of variable definitions. An example file is included, wrf_hydro_nwm_public*/trunk/NDHMS/template/setEnvar.sh. For this walkthrough we will use default

values with the exception of SPATIAL_SOIL in order to enable specification of spatially distributed land surface model parameters.

4. Use vi or your text editor of choice to open the setEnvar.sh script and change SPATIAL_SOIL=0 to SPATIAL_SOIL=1.
   After saving your changes, your setEnvar.sh script should now look like:

```bash
#!/bin/bash

# WRF-Hydro compile time options

# This is a WRF environment variable. Always set to 1=On for compiling
WRF-Hydro.
export WRF_HYDRO=1

# Enhanced diagnostic output for debugging: 0=Off, 1=On.
export HYDRO_D=1

# Spatially distributed parameters for NoahMP: 0=Off, 1=On.
export SPATIAL_SOIL=1

# RAPID model: 0=Off, 1=On.
export WRF_HYDRO_RAPID=0

# Large netcdf file support: 0=Off, 1=On.
export WRFIO_NCD_LARGE_FILE_SUPPORT=1

# WCOSS file units: 0=Off, 1=On.
export NCEP_WCOSS=0

# Streamflow nudging: 0=Off, 1=On.
export WRF_HYDRO_NUDGING=0
```

5. Now we will compile the model by executing the wrf_hydro_nwm_public*/trunk/NDHMS/compile_offline_NoahMP.sh script. Navigate to the wrf_hydro_nwm_public*/trunk/NDHMS directory and execute the compile script, supplying our compile options setEnvar.sh file:
   `./compile_offline_NoahMP.sh template/setEnvar.sh`
   A number of messages and possibly warnings will be output to your stdout, any warnings about "…non-existent include directories…" can be ignored.

6. There should now be a new directory called 'Run', wrf_hydro_nwm_public*/trunk/NDHMS/Run, containing the compiled binary, wrf_hydro.exe and required *.TBL files. These are the essential files that we will use with our example test case to run a WRF-Hydro simulation.
   Template namelist files are also copied over from the wrf_hydro_nwm_public*/trunk/NDHMS/template directory into the wrf_hydro_nwm_public*/trunk/NDHMS/Run directory during the compilation step. These namelist files are *templates only* and require substantial edits by the user. Your WRF-Hydro example test case will contain prepared namelists, thus these templates will not be needed and can be

deleted.

Running a WRF-Hydro simulation

In this section we will use our compiled WRF-Hydro model and an example test case to run a WRF-Hydro simulation. This walkthrough is using the Croton, NY example test case. Details on the domain and time period of the simulation are provided in the example_case/Readme.txt file.

1. First we need to copy the files in the wrf_hydro_nwm_public*/trunk/NDHMS/Run directory to the directory containing the domain and forcing files. For the WRF-Hydro Gridded configuration, these files are located in the directory example_case/Gridded.
   Copy the *.TBL files to the Gridded directory, for example:

   From your project directory:
   ```
   cp wrf_hydro_nwm_public*/trunk/NDHMS/Run/*.TBL example_case/Gridded
   ```
   Copy the wrf_hydro.exe file to the Gridded directory, for example:
   ```
   cp wrf_hydro_nwm_public*/trunk/NDHMS/Run/wrf_hydro.exe example_case/Gridded
   ```

NOTE: There are 2 other subfolders with the names Gridded and Reach. These folders contain prepared files for the Gridded and Reach configurations of WRF-Hydro. Information regarding physics options and routing configurations can be found in the *WRF-Hydro V5 Technical Description* located at https://ral.ucar.edu/projects/wrf_hydro. Also, note that there is only one FORCING directory. The same forcing data can be used for all three routing configurations.

2. Next we need to copy our forcing data to the Gridded directory.

   From the example_case/Gridded directory:`cp -r ../FORCING .`

Your domain/croton_NY/Gridded directory should now have the following directory structure and files:
```
-project_directory
        |___wrf_hydro_nwm_public*/

        |___example_case
           |___Gridded/
                  |___FORCING/
                  |___DOMAIN/
                  |___RESTART/
                  |___referenceSim/
                  |___CHANPARM.TBL
                  |___GENPARM.TBL
                  |___HYDRO.TBL
                  |___MPTABLE.TBL
                  |___SOILPARM.TBL
                  |___hydro.namelist
                  |___namelist.hrldas
                  |___wrf_hydro.exe
```
3. Now we will run the simulation. Note that there are many options and filepaths that need to be set in the two namelist files "hydro.namelist" and "namelist.hrldas". However, for this walkthrough these files have

been prepared for you.

Before running the model, ensure you are in the example_case/Gridded directory.

We will now run the model using mpirun with 2 cores. This command may differ depending on your system configuration, but here is an example of what this might look like:

```
mpirun -np 2 ./wrf_hydro.exe
```

4. If your simulation ran successfully, there should now be a large number of output files. Descriptions of the output files can be found in the *WRF-Hydro V5 Technical Description* at (https://ral.ucar.edu/projects/wrf_hydro). There are also two important files for determining the success or failure of the run, "diag_hydro.00000" and "diag_hydro.00001". These diag_hydro.* files contain logs and diagnostics on the simulation run, and one file is produced per core used in the run. Since we ran using 2 cores, we have 2 diag_hydro.* files.

   You can check that your simulation ran successfully by examining the last line of the diag files, which should read The model finished successfully........

```
cat diag_hydro.00000
```

   If this line is not present, the simulation did not finish successfully.

5. You can check the validity of your simulation results by comparing the restart files produced during your model run with the restart files included in the example_case/Gridded/referenceSim directory. The restart files contain all the model states and thus provide a simple means for testing if two simulations produced the same results.

Note: Our current example test case has only been run and tested with the Noah-MP land surface model. For information regarding running WRF-Hydro with Noah please see the WRF-Hydro V5 Technical Description.