Unofficial WEC-Sim Process Manual

Chelsea Kimball, University of New Hampshire

1. Read through the instructions on GitHub pages for the WEC-Sim installation. It is also recommended that the user reads through the entire User Manual before getting started.

https://wec-sim.github.io/WEC-Sim/master/user/getting_started.html

2. To download Wec-Sim from GitHub, it is recommended that you install GitHub's Command Line Interface (CLI), which is located here:

https://cli.github.com/

3. Then, once on the GitHub Wec-Sim repository, located here: https://github.com/WEC-Sim/WEC-Sim, find the green 'Code' button, and copy the GitHub CLI line

gh repo clone WEC-Sim/WEC-Sim

4. Use this line in the GitHub CLI to download Wec-Sim. Continue with Steps 1-4 listed on the 'Getting Started' page.

5. Before getting started on Wec-Sim, you will need a geometry file of your device. It is recommended that you use a simplified .stl file. Cut down on complicated features, and use the largest, most important features in the design file. This will help prevent long run times.
    - When saving a .stl file from SolidWorks there are several important items that must be completed:
        - Orientation – ensure that the Z direction in SW is pointing upwards and corresponds to your device's vertical axis in the water column.
        - Origin – you may end up needing two files with origins located in different places:
            - The WEC-Sim .stl file requires that the origin be placed at the device's center of gravity.
                - Note: If you are using a multiple body system, ensure that each individual body you are running references its individual center of gravity.
            - Capytaine and Nemoh require that the .stl file's origin corresponds to the waterplane location
                - Capytaine and Nemoh may also require that you apply a 'Convert to Mesh Body' feature once you have completed your design of your part file
                - The Mesh Body allows control of the mesh characteristics, it's better to have a coarser mesh than a finer one
        - Save the .stl file as an ASCII file, and ensure you've chosen meters for units.

6. You will need to install the software 'meshmagick' to your computer to transform your .stl file into a mesh data file.
    a. Download and install Anaconda to your computer: https://www.anaconda.com/products/individual

This is required for installing meshmagick, but also for running Capytaine later.

    b. Restart your computer after download is complete.
    c. Meshmagick is available for free on GitHub, and can be downloaded using the GitHub CLI or GitBash; https://github.com/LHEEA/meshmagick
    d. Open the Anaconda Prompt and navigate to the file folder location (root repository) where meshmagick was downloaded. Use one of the install methods below. Either should work, but you may need to try one or the other:
        - pip install -e .
        - conda install -c frongere meshmagick

7. Transform your .stl file into a .nemoh file
    a. Save a copy of your .stl file to your meshmagick root folder
    b. Open an Anaconda Prompt
    c. Navigate to the meshmagick root folder
    d. Use the command: meshmagick yourfilename.stl -ofmt nemoh
    e. Examine the meshmagick root folder to determine whether your new file has saved in the .nemoh format
    f. Record the number of points and quadrilaterals within each .nemoh file on a separate piece of paper, these will be entered into Capytaine later. The total # of points appear first in file, where each point is represented by x, y, z coordinates. The number of quadrilaterals must be calculated by subtracting the first Line Number after the break of five zeros from the last Line Number in the .nemoh file.

8. Verify that your .nemoh file is in the correct orientation with waterline at the expected location, and that there are no other errors with your mesh. You can do this using meshmagick's viewer function.
    a. Open an Anaconda prompt and navigate to the meshmagick folder
    b. View your recently created .nemoh file(s) using the following command:
        meshmagick yourfilename.nemoh --show
        Reference:
        https://lheea.github.io/meshmagick/command_line/cli_usage.html#mesh-file-visualization
    c. In the viewer, use your mouse to examine the mesh file. Is the Z axis pointing upwards along the vertical axis that corresponds with the water depth?
    d. Press the letter 'h' on your keyboard, does the waterline plane show in the expected location?
    e. Does the mesh look continuous without breaks or errors? Recommend fixing mesh if needed, reference:

https://lheea.github.io/meshmagick/command_line/cli_usage.html#mesh-healing

9. Download & Install Capytaine with Conda
    a. Follow instructions available here: https://ancell.in/capytaine/latest/user_manual/installation.html#with-conda
    b. Open an Anaconda prompt and submit the following:
    conda install -c conda-forge capytaine

10. Running BEM Software – Capytaine
    a. Open Anaconda Navigator from start menu
    b. Install and then Open the Spyder IDE Environment from the Anaconda Navigator
    c. Locate the file folder within Wec-Sim that contains the Capytaine example for RM3 (or other relevant example to you):
        … WEC-Sim\examples\BEMIO\CAPYTAINE\rm3
    d. Create another folder under the CAPTYAINE folder within Wec-Sim, and name it for your project.
    e. Take this time to review the Capytaine User Manual for background information: https://ancell.in/capytaine/latest/user_manual/index.html
    f. Copy the files from the relevant Wec-Sim example for your project into your new file folder. Delete any dat/gdf/Nemoh files and replace with the .nemoh textfiles you created in step 7.
    g. Delete any KH or Hydrostatics files (these will be recreated when you run the code for your own geometry)
    h. Delete any H5 or NC files (these are results corresponding to the example, and will be re-created for your own device)
    i. You may need to edit the gbm_dofs.py file to make it relevant for your application, however if you choose a Wec-Sim example with identical degrees of freedom to your own application, then you can leave it unchanged
    j. You will need to modify the __example.py file for your application, so renaming it to something specific to your project will help prevent confusion. In Spyder IDE, navigate to your project folder in the upper right and view Files.
    k. Open the .py file that you just renamed for your project, it should appear in the viewer on the left in Spyder. You will need to edit the following items within the .py file before running Capytaine:
        - "bem_file" → rename the mesh file(s) to the .nemoh files you created in Step 7
        - "bem_cg" → alter to reflect the location of your device's center of gravity. Remember: Your .nemoh file is relative to the waterline plane, so these centers of gravity are relative to the waterline plane.
        - "bem_name" → rename as appropriate for your project
        - Alter the wave frequencies, headings, and water depth as necessary
        - Rename the file name for the .NC file which will be produced

- Change any other parameters you see fit (ie density)
- Save the file & Click Run File (top right, green arrow)
- Code may take a significant amount of time to run (several hours). Check after a few minutes to see whether there are errors generated, but otherwise be patient. Once code has finished, check file folder to see whether the .NC file was generated.

11. If you have completed Step 4, and have installed Wec-Sim completely, open Matlab on your computer. It's now time to use Wec-Sim. The Wec-Sim team has created a BEM transformation code, called Bemio that works within Matlab.
    a. Navigate to your project's file folder from step 9 in Matlab
    b. Open the bemio.m code and change the filename.nc to match your created .nc folder name
    c. Run the bemio code and check to see whether the .h5 file has been generated

12. Set up Project folder in WecSim, emulating one of their examples
13. Modifications needed & how to run
14.
15. PTO Modification
    a. https://energy.sandia.gov/wp-content/uploads/2014/06/SAND2015-2069C.pdf
    b.

Additional Resources:

Capytaine https://github.com/mancellin/capytaine

Wec-Sim

Youtube https://www.youtube.com/channel/UCAzFtxxapywCN4ler_bN4lw Has 5 training videos on how to use Wec-Sim