# To create hydrodynamic layers

1. Download the Pipeline zip file that includes the fvcomm toolbox, Matlab libraries and a Data folder. Extract the contents of the zip file.
2. Download your hydrodynamic data from wherever you find it and save this file in the folder called Data in the zip file. A note on file organisation for the Matlab script: save your various hydro dates in separate folders (i.e. April\_5, Sep\_5, etc.) because it will generate a lot of input files and it will get confusing if you have it all in one folder. I’ve put a dummy folder in the zip file so that you see what I mean. There should only be one .nc file in this folder because the Matlab script will just take the first one! If you want multiple files, make subfolders and then make the “month” variable (see Step 4) “April\_5/res\_200/”, for example.
3. Open the Matlab script
4. There are a few places you need to change values to adapt the script for new files. I have marked them in the code with %!!! Symbols. They are on lines:
   1. 10
   2. 21
   3. 34-46
   4. Dotted throughout whenever there is an optional line of code to tell you to remove the % symbol to run it
5. The section on parameters (lines 30-41) is where you set all the map-specific parameters for this set of outputs. **Note**: The values in there right now are just default values I used for another project, you need to change them to match your map!
6. On line 62, I use a function called do\_residual\_plus. This is an **edited version of the released do\_residual in the fvcomm toolbox.** The script for it is in the zip file and if the folder organisation is correct, the script should find it without issue.
7. You should be able to run the whole script from here. I have broken it into sections so if you Run Section, it should do digestible chunks. It will take some time (especially the interpolation step)! Sometimes I have added commands to output the elements it’s doing just so I can keep track of where it’s at. You can ignore these. **Note**: this script generates a lot of files because I save files along the way, so you don’t have to start from scratch if you want to make changes halfway through the script

# Creating habitat layers in R

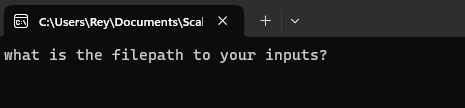
1. In the zip file, you’ll find the sf\_depth\_to\_layers.R script. This script will help create the habitat layers input files that you need for MerMADE
2. What you need for this script to run:
   1. A .txt file for habitat types in your seascape. This is the 2D file that you most likely produced from software like GIS where each cell has a habitat type value. NOTE: if you did produce it from GIS, you need to remove the metadata at the top of the file (ie number of rows, etc). This habitat types file should just include tab-separated numbers, no text. There is an example file in the zip folder called hab\_types.txt in the Data folder.
   2. IF YOU ARE RUNNING A PATCH BASED MODEL: you need the .txt file with patch designations. Again, this was probably produced in software like GIS and needs the same treatment as outlined in the previous point. The numbering of patch-designations should be sequential and start at 0
   3. A .txt file for seafloor depths. This was produced in the section of this pipeline for creating hydrodynamic layers so if you ran that, you will have this in the correct format.
   4. NOTE: all these files need to have the SAME DIMENSIONS!
3. Note that there are areas where you need to change values! These are indicated by the #!!! Symbols and can be found on lines
   1. 18
   2. 21
   3. 26, 32, 36
   4. 42
   5. 55-60
   6. 74
   7. 78
   8. 103, 104
   9. 132-133
   10. 140-150
4. Go through the script line by line, making changes where necessary and create your habitat files
5. Pay attention to your working directory and where your files are being saved!
6. The last section will open a new little plotting window with a 3D rendering of your seascape, you can zoom in, drag it around and change the perspective on it. Have a play!
7. Note that there are two files in the Inputs folder of Sims/Test1. These list the habitat file and patch file names, which is how MerMADE knows what to look for. If you change the names in the R script, these changes need to be reflected in these files! More details can be found in the user manual

# Using layers as MerMADE inputs

1. The zip file also contains a folder called Sims, where I have put a dummy folder called Test1 as well as the MerMADE executable to make running the model easier.
2. Just a quick user-friendly note, if you want to change any of the parameters in the input files, opening the .txt file is not very user-friendly because the columns are all offset and don’t line up. I would recommend copying the material into an excel sheet so that all the columns line up properly.
3. For each simulation you run, you must set up a folder that contains an Inputs and an Outputs folder within it. It doesn’t matter what you name the Test1 folder, but I tend to make them sequential so that running for loops in R is easier.
4. Everything you have produced in this pipeline belongs in the Inputs folder of your simulation. Please refer to the user manual for more detailed information on filling out the input files.
5. I would add details of each simulation to the relevant lines in the “What’s What.txt” file to keep a record.
6. To run simulations, simply direct MerMADE to look in the correct folder for the sim (I’ll detail this down below). **Note:** remember to add your new simulation folders to the “What’s What.txt” file to keep track of everything, otherwise it gets confusing very quickly!

# Running MerMADE simulations

1. Inside the Sims folder, you’ll find the executable for MerMADE\_scallops.exe. This is a windows-only program! To run a simulation, just double click this file and you’ll see a console open with the following prompt:



1. Because the executable is located within the Sims folder, you can use a relative path: “Test1/” or “Test4/” for example. This is all the information you need to provide; the rest is in the input files. MerMADE will run the indicated simulation and save the results in the Outputs folder within that sim folder.
2. Right now, this version of MerMADE still outputs a lot of stuff to the console! This is so I know where it’s at and what’s happening, you can ignore everything that is printing to the console! When the simulation is finished, it will say **Done** at the end after having written all the output files. You can close the console window at that point.

# Visualising results in R

1. Open the pipeline.R script in Rstudio
2. Set your working directory to be within the pipeline folder this is the entire file path to that folder)
3. Again, look out for #!!! Symbols because these are lines where you need to change values. You’ll find them on lines:
   1. 10
   2. 21-22
   3. 39
   4. 52
   5. 63
4. You can just run this script through and it should produce two types of figures: one map with the movement tracks and one map with the settlement sites of competent individuals
5. You probably already know but the size of your plotting window sometimes has a big effect on whether the tracks plot correctly, so if the tracks don’t seem to match up with the seascape properly, try adjusting the size of the plotting window on the right and see if that fixes the problem
6. To plot a new simulation, simply go up to line 22, change the test\_num parameter and run through it again to import new indiv\_mov.txt files and plot new tracks.