**Documentation for OpenFOAM setup to simulate wind flow over coastal dunes**

The setup is used to simulate Egmond costal dune and idealized syntactic dune.

# **Simulating costal dune at Egmond**

Wind flow over coastal dune located in Egmond, as shown in the next figure, is simulated and CFD model is validated by comparing CFD results with measured data.

|  |  |
| --- | --- |
|  |  |

The boundary conditions and a sample of meshing is shown as following figure.

|  |  |
| --- | --- |
|  |  |

* 1. **Setup**

The setup is in “*Wind\_Dune\_Egmond*” folder. It contains main setup folders for OpenFOAM, including “0”, “constant”, and “system” folder. The folder is also contain bash scripts to run automatically the setup for various inlet velocity direction.

* 1. **How run?**

For meshing using snappyHexMesh run  
  
name:~$ bash snappy\_bash.sh

If you run it in cluster (HPC) machine (velocity which is the name of department cluster system) run bash script *run\_mesh\_all\_nohup\_angle.sh* in terminal

The file run *OpenFOAM* setup for various inlet wind angle and store results in specific folder by corresponding name of inlet wind angle. The inlet wind angle specified in “**for** angle\_parameter **in** 0 10 20 30 40 45 50 60”

|  |
| --- |
| *run\_mesh\_all\_nohup\_angle.sh*  *--------------------------------------------------------------------------------------------------------------*  #!/bin/sh  **for** angle\_parameter **in** 0 10 20 30 40 45 50 60  **do**  **mkdir** run\_angle**$angle\_parameter**  #cp -r {0,system,constant,Allclean,Allrun,plot\_gnuplot\_residuals} run\_Mesh\_factor$Mesh\_refined\_factor\_parameter  **cp** -r **{**0**,**system**,**constant**,**Allrun\_bash.sh**}** run\_angle**$angle\_parameter**  **cd** run\_angle**$angle\_parameter**  # rest of code …  **cd** **..**  **done** |

[Name\_@velocity\_name]$ nohup sh Allrun\_bash.sh &

* To run in your own computer or laptop use the setting in folder “*Wind\_Dune\_Egmond /Run\_in\_a\_usual\_computer\_notebook*”. This setup is also prepared to run the expanded domain which is suitable for inlet velocity angles 70, 80, 89 degrees. To run in the terminal execute bash script *run\_mesh\_all\_angle.sh*

name:~$ bash run\_mesh\_all\_angle.sh

# **Simulation idealized synthetic dune**

* 1. **Setup for creating ground surface *stl* file and meshing**

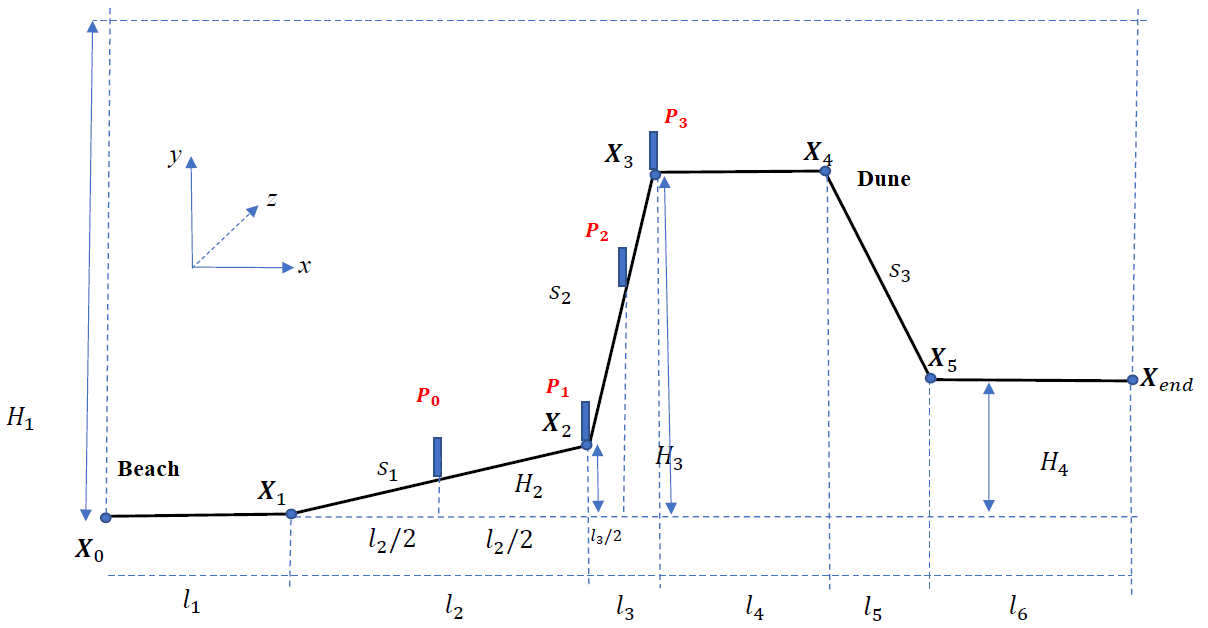
The setup is in “*ideal\_profile\_Saeb*” folder. It contains a shell script and python script to great *stl* file of ground surface regarding the idealized dune parameters, adjusting the corresponding OpenFOAM files and meshing the domain.

* 1. **How run?**

To run in terminal run the following bash script “*ideal\_profile\_Saeb /run\_bash\_ideal\_profile.sh*”

name:~$ bash run\_bash\_ideal\_profile.sh

* *“ideal\_profile\_Saeb/goemetry\_setup/* *profile\_geometry”* folder contains a text file *“data\_file.txt”.* This file defines the idealized dune geometry features, including dune height, length, and slope. For explanations of the parameters in *"data\_file.txt"* please refer to the following figure.



* *“ideal\_profile\_Saeb/goemetry\_setup/OpenFoam\_file”* folder contains OpenFOAM files(*blockMeshDict, initialConditions, snappyHexMeshDict*) that should be adjusted to correspond with the dune geometry features. The OpenFOAM files in this folder use as base script and then revised and coped to the OpenFOAM setup for run.
* *“ideal\_profile\_Saeb/OpenFoam\_setup”* contains the main defualt OpenFOAM setup.
* Running the shell script *“run\_bash\_ideal\_profile.sh”* create *stl* file, adjusting OpenFOAM files, copy the OpenFOAM main default setup and adjusted files in “*run\_setup”* folder for each case of dune geometry.
* In shell script *“run\_bash\_ideal\_profile.sh”* dune geometry parameters can be defined by changing

H3\_values=(6 9 12 15 18 21 25), s1\_values=(0.01 0.05 0.1) and s2\_values=(0.25 0.33 0.5 1)

run\_bash\_ideal\_profile.sh

----------------------------------------------------------------------------------------------------------------

#!/bin/sh

H3\_values**=(**6 9 12 15 18 21 25**)**

s1\_values**=(**0**.**01 0**.**05 0**.**1**)**

s2\_values=(0.25 0.33 0.5 1)

#-------create required stl and initial Conditions file ----------------------

**cd** created\_info\_profile

**for** H3 **in** "${H3\_values[@]}"**;** **do**

**for** s1 **in** "${s1\_values[@]}"**;** **do**

**for** s2 **in** "${s2\_values[@]}"**;** **do**

**mkdir** H3\_${H3}\_s1\_${s1}\_s2\_${s2}

# code here

**done**

**done**

**done**

#----------------setup OpenFoam files ------------------------

**cd** **../**run\_setup

**for** H3 **in** "${H3\_values[@]}"**;** **do**

**for** s1 **in** "${s1\_values[@]}"**;** **do**

**for** s2 **in** "${s2\_values[@]}"**;** **do**

# code here

**done**

**done**

**done**

#--create required stl and initial Conditions file 708090-----

**cd** **../**created\_info\_profile

**for** H3 **in** "${H3\_values[@]}"**;** **do**

**for** s1 **in** "${s1\_values[@]}"**;** **do**

**for** s2 **in** "${s2\_values[@]}"**;** **do**

# code here

**done**

**done**

**done**

#------------------------------------------------setup OpenFoam files 708090---------------------------------------------------------

**cd** **../**run\_setup

**for** H3 **in** "${H3\_values[@]}"**;** **do**

**for** s1 **in** "${s1\_values[@]}"**;** **do**

**for** s2 **in** "${s2\_values[@]}"**;** **do**

# code here

**done**

**done**

**done**

* For each case a folder of OpenFOAM setup is created by corresponding name in *“run\_setup”* folder.
* Meshing is implemented for created OpenFOAM setups. These setups can be run separately to simulate difference different dune geometries.
* Python scripts are available for postprocessing and proting the figures if you needed in *“ideal\_profile\_Saeb/run\_setup/20230717-postprocessing\_prob*” and *“ideal\_profile\_Saeb/run\_setup/20230719\_result\_proccession\_plot*”.