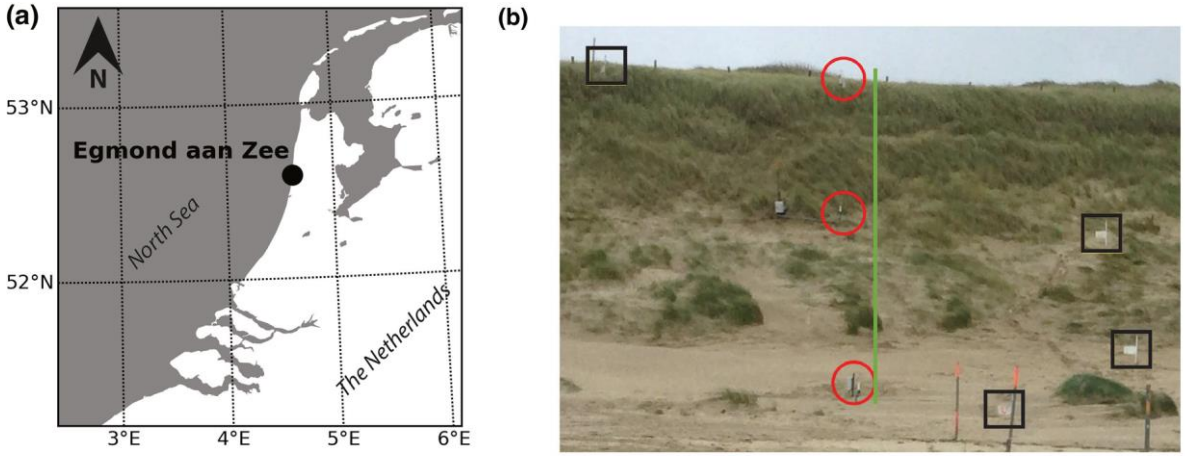
Documentation for OpenFOAM setup to simulate wind flow over coastal dunes

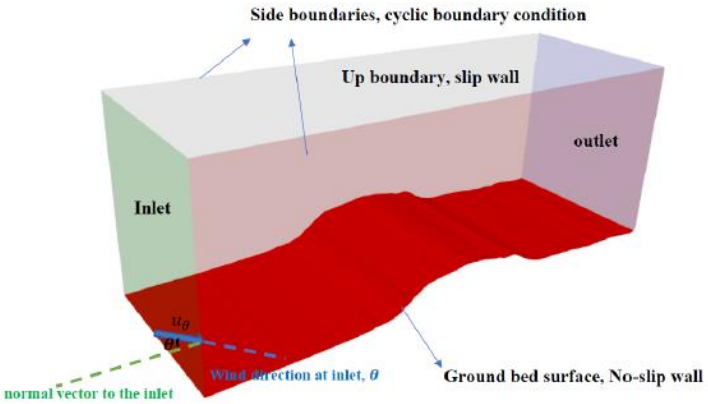
The setup is used to simulate Egmond costal dune and idealized syntactic dune.

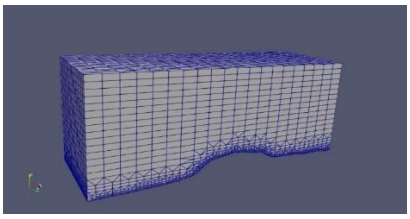
1. Simulating costal dune at Egmond

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.





1.1. 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.

1.2. 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

2. Simulation idealized synthetic dune

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

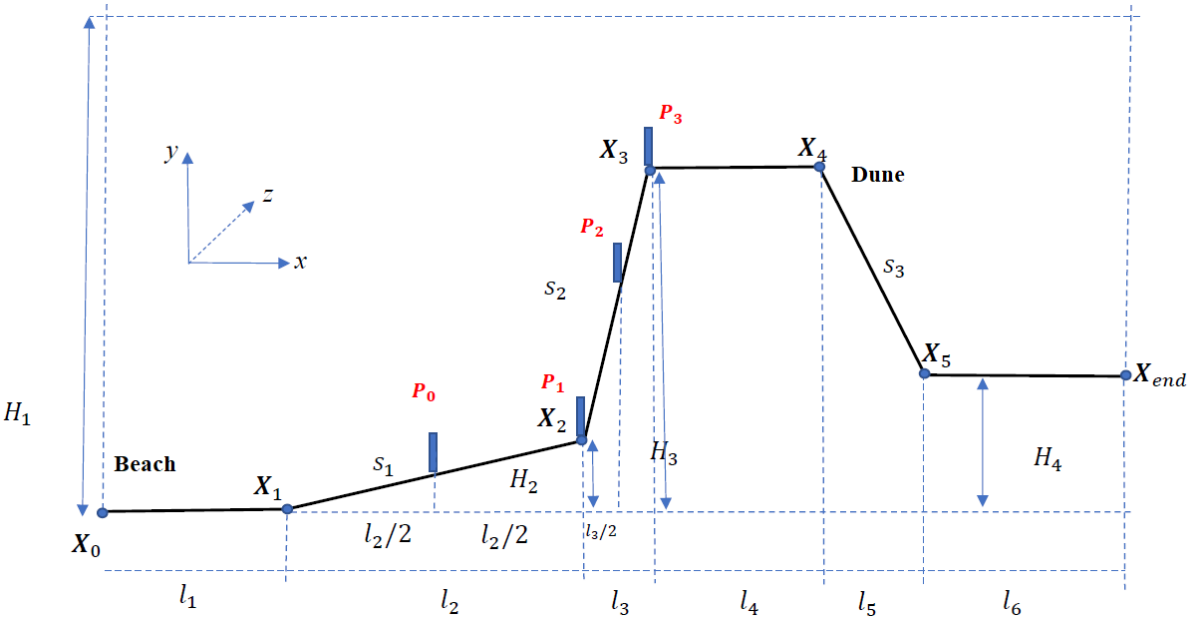
The setup is in "*ideal_profile_Saeb*" folder. It contains a shell script and python script to great *stl* file of ground surface regarding the idealized dune parameters, adjusting the corresponding OpenFOAM files and meshing the domain.

2.2.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*".