Implementing shear current theory into the `waves2Foam` toolbox

Koray Deniz Göral

Department of Mechanical Engineering,
Fluid Mechanics, Coastal and Maritime Engineering,
Technical University of Denmark,
Kgs. Lyngby, Denmark

January 20, 2022

Outline

- The waves2Foam toolbox
- The simple (uniform) current solution and the shear current mixing length theory from the van Driest solution
- The structure of waves2Foam toolbox (waveTheory class, waveProperties.input file, and makeNewWaveTheory script)
- Implementation of the shear current mixing length theory inside the waves2Foam toolbox
- Tutorial case

waves2Foam toolbox

The waves2Foam toolbox is an open-source plug-in toolbox for the OpenFOAM and was originally developed by Niels Gjøl Jacobsen at the Technical University of Denmark. The libraries in the waves2Foam toolbox are used for generating and absorbing free surface water waves applied with the relaxation zone technique (active sponge layers) where the relaxation zones can take arbitrary shapes.

The waves2Foam toolbox is available for download through the OpenFOAM-Extend SourceForge SVN. One can download and compile the waves2Foam toolbox by using the below coding:

```
// * Source the selected OpenFOAM version * //
svn co http://svn.code.sf.net/p/openfoam-extend/svn/trunk/Breeder_1.6/other/waves2Foam
mkdir -p $WM_PROJECT_USER_DIR/applications/utilities
cp -r waves2Foam $WM_PROJECT_USER_DIR/applications/utilities
cd $WM_PROJECT_USER_DIR/applications/utilities/waves2Foam
./Allwmake
```

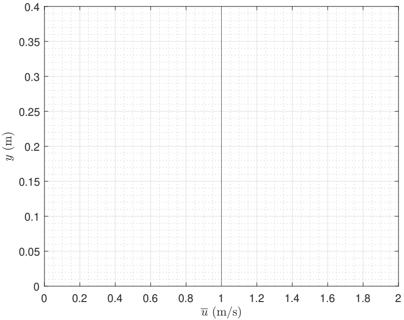
More details are in the waves2Foam toolbox Wiki page:
https://www.openfoamwiki.net/index.php/Contrib/waves2Foam

Simple (uniform) current

The simple (uniform) current also named potential current introduces a current that is uniform over the entire water depth.

$$\overline{u} = U$$

$$\overline{u} = f(U)$$



The potential current approach does not resolve different flow characteristics for the entire water depth as the theory assumes a fixed velocity for the entire flow section. As an example, if one wants to study the sediment motion which mobilizes from the bed and generally transports in the boundary layer (for the current it is the full water depth), the potential current theory results do not give realistic results.

The shear current mixing length theory

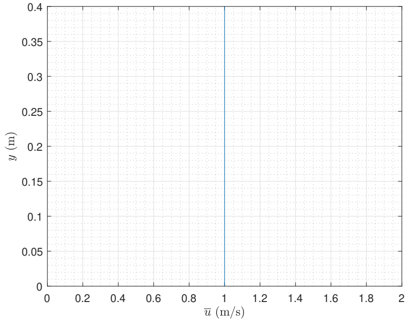
The mean velocity distribution, $\overline{u}$, considering the van Driest solution with the extension of Cebeci & Chang is given as:

$$\overline{u} = 2U_f \int_0^{y^+} \frac{dy^+}{1 + \{1 + 4\kappa^2 (y^+ + \Delta y^+)^2 [1 - \exp(-\frac{(y^+ + \Delta y^+)}{A_d})]^2\}^{1/2}}$$
$$\overline{u} = f(U_f, y, k_s, \nu)$$

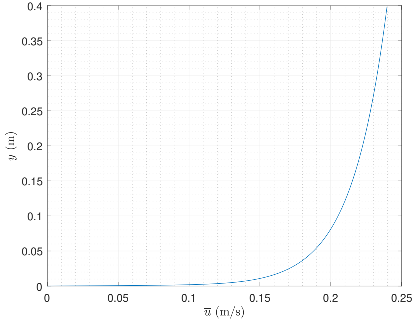
$$\Delta y^+ = 0.9[\sqrt{k_s^+} - k_s^+ \exp(-\frac{k_s^+}{6})] \qquad \text{if} \qquad 5 < k_s^+ < 2000$$
$$\text{where} \quad k_s^+ = \frac{k_s U_f}{\nu}$$

where y is the distance from the wall, U_f is the friction velocity, κ is the von Karman constant, A_d is the damping coefficient, Δy is the coordinate displacement (coordinate shift), and k_s is the Nikuradse's equivalent sand roughness.

The shear current mixing length theory

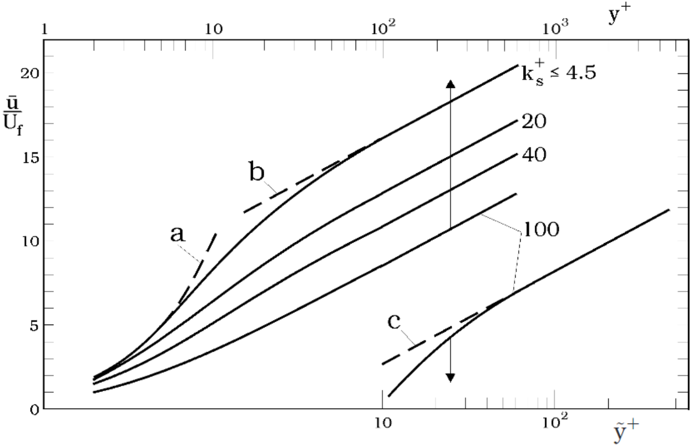


Example mean velocity distribution, $\overline{u}$, considering potential current theory



Example mean velocity distribution, $\overline{u}$, considering shear current theory with the van Driest solution

The shear current mixing length theory



Example of van Driest velocity profiles for different values of roughness (Taken from [SF20], Figure 3.21 at page 101)

`waveTheory` class and sub-classes

Wave Theories:

1. Regular wave theories
   1. Stokes first order
   2. Standing Stokes first order
   3. Stokes second order
   4. Modulated second-order Stokes wave
   5. Stokes fifth order
   6. First-order cnoidal theory
   7. Stream function wave
2. Bichromatic wave theories
   1. First-order bichromatic wave
   2. Second-order bichromatic wave
3. First-order irregular waves
   1. Spectral shape
      1. ...

   ...
4. Potential current

Solvers:

1. waveFoam
2. porousWaveFoam
3. waveDyMFoam (moving meshes)

There is also an additional wave theory class named `combinedWaves`. The wave theories can be combined with the help of this `combinedWaves` wave theory class.

waveTheory class and sub-classes

One should examine the potential current theory library (potentialCurrent) in the waves2Foam toolbox for understanding the coding structure in waves2Foam toolbox where the library is located at:

```
$WM_PROJECT_USER_DIR/applications/utilities/waves2Foam/src/waves2Foam/waveTheories/
    current/potentialCurrent
```

There are two files in the potentialCurrent folder, in which potentialCurrent.H is the class declaration and potentialCurrent.C is the class definition files.

waveProperties.input file structure

One has to do the preparation before executing `waveFoam` solver. The `waveProperties.input` file has to be converted to `waveProperties` file by executing the `setWaveParameters` application, where all the necessary wave parameters are transferred based on physical meaningful properties with the given application. Then, the `setWaveField` application should be executed. The `setWaveField` application sets the initial conditions that the user defined in the wave theory [Jac17]. Some part of the `waveProperties.input` file:

```
seaLevel    0.00;
relaxationNames (inlet outlet);
initializationName outlet;
inletCoeffs
{
    // Wave type to be used at boundary "inlet" and in relaxation zone "inlet"
    waveType    stokesFirst;
    // Ramp time of 2 s
    Tsoft       2;
    // Water depth at the boundary and in the relaxation zone
    depth       0.400000;
    // Wave period
    period      2.0;
    ...
```

makeNewWaveTheory script

The waves2Foam toolbox has a special coding script that should be followed when creating a new wave theory inside this toolbox. Therefore, the creator of the waves2Foam toolbox has templated a small script named makeNewWaveTheory that creates a new wave theory with the given name. Then, the created folder, which contains the declaration and definition files for the new wave theory, can be transferred to the related wave theory's sub-folder inside the waveTheories dictionary with:

```
cd $WM_PROJECT_USER_DIR/applications/utilities/waves2Foam/doc/templateWaveTheory
./makeNewWaveTheory shearCurrent
mv shearCurrent $WM_PROJECT_USER_DIR/applications/utilities/waves2Foam/src/waves2Foam/
    waveTheories/current
```

New shearCurrent library

The new shearCurrent library is created from scratch. Therefore, after creating the shearCurrent scripts with the help of makeNewWaveTheory application, one should implement all the necessary details to the declaration and definition files of the new library. The location of the new shearCurrent library:

```
cd $WM_PROJECT_USER_DIR/applications/utilities/waves2Foam/src/waves2Foam/waveTheories/
    current
```

New shearCurrent library

The class declaration part of the shearCurrent.H file:

```
protected:
    // Protected data
    scalar Tsoft_;
    scalar localSeaLevel_;
    scalar Uf_;
    scalar nu_;
    scalar ks_;
    scalar kappa_;
    scalar Ad_;
    const volScalarField& y_;
    const volVectorField& x_;
```

Friction velocity, U_f (declared as Uf_), fluid kinematic viscosity, ν (declared as nu_), Nikuradse's equivalent sand roughness, k_s (declared as ks_), von Karman constant, κ (declared as kappa_), damping coefficient, A_d (declared as Ad_), T_{soft} (declared as Tsoft_), local sea level (declared as localSeaLevel_) are added to the class declaration part of the shearCurrent.H file.

The vertical distances of each cell center are stored under the name y (declared as y_). The points for each cell in the computational domain are also stored (declared as x_) which can be used for indexing in the van Driest cumulative trapezoidal integration part.

New shearCurrent library

The definition file of the `shearCurrent` library is created with the previously given parameters. In addition, as the vertical distances from the wall are calculated with the help of the `wallDist` function embedded in the OpenFOAM library, `#include "wallDist.H"` should be added to the `#include` section which is located on the top of the definition file.

```
#include "wallDist.H"
...
    Tsoft_(readScalar(coeffDict_.lookup("Tsoft"))),
    localSeaLevel_
    (
        coeffDict_.lookupOrDefault<scalar>("localSeaLevel", seaLevel_)
    ),
    Uf_(readScalar(coeffDict_.lookup("Uf"))),
    nu_(readScalar(coeffDict_.lookup("nu"))),
    ks_(readScalar(coeffDict_.lookup("ks"))),
    kappa_
    (
        coeffDict_.lookupOrDefault<scalar>
        (
            "kappa",
...         0.41
```

New shearCurrent library

The steps for implementing the member functions inside the definition file of the shearCurrent library from the blank shearCurrent.C script are itemized below:

- Eta definition

  The old definition of the eta member function is changed to its new definition considering the given input of local sea level in the waveProperties.input file. The new definition of the eta member function is shown below:

```
scalar shearCurrent::eta
(
    const point& x,
    const scalar& time
) const
{
    scalar eta = localSeaLevel_;
    return eta;
}
```

New shearCurrent library

- Excess pressure definition

The old definition of the excess pressure member function is changed to its new definition considering the excess pressure created by the given local sea level. This pressure definition is also used by the potentialCurrent library, therefore, the definition is directly taken from the potentialCurrent.C for keeping the functionality of the wave2Foam toolbox discussed before. The new definition of the excess pressure member function is shown below:

```
scalar shearCurrent::pExcess
(
    const point& x,
    const scalar& time
) const
{
    return referencePressure(localSeaLevel_);
}
```

New shearCurrent library

- Flow velocity definition

The old definition of flow velocity member function is changed to its new definition considering the van Driest solution with the extension of Cebeci & Chang by adding the effect of the bed roughness. The extensions of k_s^+, and Δy^+ by Cebeci & Chang are calculated before the main forAll loop.

```
vector shearCurrent::U
(
    const point& x,
    const scalar& time
) const
{
    volScalarField temp=y_;
    volScalarField temp2=y_;
    temp *=scalar(0.0);
    temp2 *=scalar(0.0);
    scalar yPlus_ = 0.0;
    scalar vanDriest_ = 0.0;
    scalar ksPlus_= ks_*Uf_/nu_;
    scalar deltayPlus_=0.9*(Foam::sqrt(ksPlus_)-ksPlus_*Foam::exp(-ksPlus_/6.0));
...
```

New shearCurrent library

- Continued...

```
    scalar shearU_ = 0.0;
    forAll(temp,index)
    {
        yPlus_=y_[index]*Uf_/nu_;
        vanDriest_=1.0/(1.0+Foam::sqrt(1.0+4.0*Foam::pow(kappa_,2.0)*Foam::pow(
 yPlus_+deltayPlus_,2.0)*Foam::pow(1.0-Foam::exp(-(yPlus_+deltayPlus_)/Ad_),2.0)
 ));
        temp[index] =vanDriest_;
        temp2[index] =vanDriest_;
        if ( index > 0)
        {
            temp[index] = temp[index-1] + 0.5*(temp2[index] + temp2[index-1])*(y_[
 index]*Uf_/nu_ - y_[index-1]*Uf_/nu_);
        }
        if ( x_[index] == x)
        {
            shearU_ = 2.0*Uf_*temp[index];
        }
    }
    return shearU_*vector(1.0,0,0)*factor(time);
}
```

Modification of Make folder

The new shearCurrent wave theory library is implemented both for shearCurrent library and setWaveParameters function. While the Make folder for the shearCurrent library is inside the waves2Foam sub-folder, the setWaveParameters function for the shearCurrent library is located at the waves2FoamProcessing/preProcessing sub-folder. Therefore, one should add the new shearCurrent wave theory library path into the files file inside the Make folders and re-compile the src folder. These mentioned Make folders can be found at:

```
$WM_PROJECT_USER_DIR/applications/utilities/waves2Foam/src/waves2Foam
$WM_PROJECT_USER_DIR/applications/utilities/waves2Foam/src/waves2FoamProcessing
```

Changes in the Make folder both for shearCurrent library and setWaveParameters function are itemized below:

- Adding shearCurrent library path to the Make/files for the waves2Foam solver

```
current=current
$(waveTheories)/$(current)/potentialCurrent/potentialCurrent.C
$(waveTheories)/$(current)/shearCurrent/shearCurrent.C
```

Modification of Make folder

- Adding `shearCurrentProperties` path to the `Make/files` for the `setWaveParameters` function

  The `potentialCurrentProperties` folder and its declaration and definition files can be used as a template for the `shearCurrentProperties`.

```
cd $WM_PROJECT_USER_DIR/applications/utilities/waves2Foam/src/waves2FoamProcessing/
     preProcessing/setWaveProperties/current
cp -r potentialCurrentProperties shearCurrentProperties
cd shearCurrentProperties
mv potentialCurrentProperties.C shearCurrentProperties.C
mv potentialCurrentProperties.H shearCurrentProperties.H
sed -i s/potentialCurrent/shearCurrent/g shearCurrentProperties.C
sed -i s/potentialCurrent/shearCurrent/g shearCurrentProperties.H
```

Modification of Make folder

- Continued...

After creating the `shearCurrentProperties` declaration and definition files, one should change the member functions sections of the definition file `shearCurrentProperties.C`.

```
// Write the already given parameters
writeGiven( os, "waveType" );
writeGiven( os, "Tsoft" );
writeGiven( os, "Uf");
writeGiven( os, "nu");
writeGiven( os, "ks");
    if (dict_.found( "localSeaLevel" ))
    {
        writeGiven( os, "localSeaLevel");
    }
    if (dict_.found( "kappa" ))
    {
        writeGiven( os, "kappa");
    }
...
```

Modification of Make folder

- Continued...

  After creating the `potentialCurrentProperties` folder, one should add the directory of the `potentialCurrentProperties.C` file to the `files` file located under the `src/waves2FoamProcessing/Make` folder.

```
/* Current type */
current=current
$(waveProp)/$(current)/potentialCurrentProperties/potentialCurrentProperties.C
$(waveProp)/$(current)/shearCurrentProperties/shearCurrentProperties.C
```

After creating all the necessary steps for the `shearCurrent` wave theory mentioned above, one should re-compile the `waves2Foam` and `waves2FoamProcessing/preProcess` wave theory libraries. Before the re-compilation, one should clean the `Make` folders by using `wclean` command.

```
cd $WM_PROJECT_USER_DIR/applications/utilities/waves2Foam/src/waves2Foam
wclean
cd $WM_PROJECT_USER_DIR/applications/utilities/waves2Foam/src/waves2FoamProcessing
wclean
```

Modification of Make folder

After the cleaning process, one can recompile all the contents under the `src` folder by executing the `Allwmake` file.

```
cd $WM_PROJECT_USER_DIR/applications/utilities/waves2Foam/src
./Allwmake
```

Preparation

The tutorial cases can be found in the waves2Foam_tutorials folder in the waves2Foam toolbox. The waveFlume tutorial inside the waveFoam dictionary is selected for creating the shearCurrent tutorial, which is named as currentFlume. The waveFlume tutorial can be copied to the $WM_PROJECT_USER_DIR/run folder by changing it's name to currentFlume:

```
mkdir --parents $WM_PROJECT_USER_DIR/run/currentTutorial/tutorials/waveFoam
cp -r $WM_PROJECT_USER_DIR/applications/utilities/waves2Foam/tutorials/waveFoam/waveFlume
      $WM_PROJECT_USER_DIR/run/currentTutorial/tutorials/waveFoam/currentFlume
cp -r $WM_PROJECT_USER_DIR/applications/utilities/waves2Foam/tutorials/commonFiles
     $WM_PROJECT_USER_DIR/run/currentTutorial/tutorials
cp -r $WM_PROJECT_USER_DIR/applications/utilities/waves2Foam/bin $WM_PROJECT_USER_DIR/run
     /currentTutorial
cd $WM_PROJECT_USER_DIR/run/currentTutorial/tutorials/waveFoam/currentFlume
sed -i 's/20/10/g' system/controlDict
```

Preparation

The currentFlume tutorial can be checked by implementing the available potentialCurrent library by changing the waveProperties.input file. The old waveProperties.input file in the constant folder should be changed to:

```
seaLevel    0.00;
relaxationNames (inlet outlet);
initializationName outlet;
inletCoeffs
{
waveType    potentialCurrent;
U           (1 0 0);
Tsoft       0;
relaxationZone
{
    relaxationScheme Spatial;
    relaxationShape  Rectangular;
    beachType        Empty;
    relaxType   INLET;
    startX      (0 0.0 -1);
    endX        (5 0.0  1);
    orientation      (1.0 0.0 0.0);
}
```

```
};
outletCoeffs
{
waveType    potentialCurrent;
U           (1 0 0);
Tsoft       0;
relaxationZone
{
    relaxationScheme Spatial;
    relaxationShape  Rectangular;
    beachType        Empty;
    relaxType   OUTLET;
    startX      (13 0.0 -1);
    endX        (18 0.0  1);
    orientation      (1.0 0.0 0.0);
}
};
```

Preparation

The type of the outlet boundary condition of the flow velocity, `U`, located at the `0.org/U.org` file should be changed from `fixedValue` to `zeroGradient` boundary condition. Then, one can run the `potentialCurrent` tutorial (the streamwise current velocity is set to 1 m/s with the Froude number equal to 0.5) by executing the `Allrun` file in the `currentFlume` folder.

Example `potentialCurrent` tutorial result at $t = 10$ s

shearCurrent tutorial

The currentFlume tutorial mentioned in the preparation section is converted for the shearCurrent tutorial. First, one has to change the context of the waveProperties.input file for the shearCurrent tutorial:

```
inletCoeffs
{
    waveType    shearCurrent;
    Tsoft       0;
    Uf          0.01;      // Friction velocity (m/s)
    nu          1.0e-06;       // Fluid kinematic viscosity (m^2/s)
    ks          1.25e-03;  // Nikuradse's equivalent sand roughness (m)
    relaxationZone
    {
        relaxationScheme Spatial;
        relaxationShape  Rectangular;
        beachType        Empty;
        relaxType   INLET;
        startX      (0 0.0 -1);
        endX        (5 0.0  1);
        orientation      (1.0 0.0 0.0);
    }
};
```

shearCurrent tutorial

```
outletCoeffs
{
    waveType    shearCurrent;
    Tsoft       0;
    Uf          0.01;      // Friction velocity (m/s)
    nu          1.0e-06;       // Fluid kinematic viscosity (m^2/s)
    ks          1.25e-03;  // Nikuradse's equivalent sand roughness (m)
    //Optional entries
    //kappa         0.41;   // von Karman constant
    //Ad            25.0;   // Damping coefficient
    relaxationZone
    {
        relaxationScheme Spatial;
        relaxationShape  Rectangular;
        beachType        Empty;

        relaxType   OUTLET;
        startX      (13 0.0 -1);
        endX        (18 0.0  1);
        orientation      (1.0 0.0 0.0);
    }
};
```

shearCurrent tutorial

The dimensionless numbers for the shearCurrent tutorial are calculated as:

$$Re_\tau = \frac{U_f * D}{\nu} = 4000, \quad Fr = \frac{V}{\sqrt{g * D}} = 0.108, \quad \text{and} \quad k_s^+ = \frac{U_f * k_s}{\nu} = 12.5$$

Calculation of the wall distances of each mesh cell is done by the wallDist function, therefore, the method for the wallDist function should be added at the end of the fvSchemes file in the system folder.

```
wallDist
{
    method          meshWave;
}
```

Additionally, U values of each cell center are sampled for comparing the simulated results with the theoretical van Driest solution with the extension of Cebeci & Chang.

shearCurrent tutorial

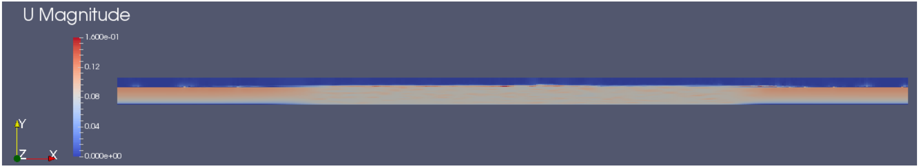
After cleaning the time folders except the 0 and 0.org and other unnecessary files (log, postprocess etc.) in the currentFlume tutorial, one can execute the new shearCurrent tutorial.

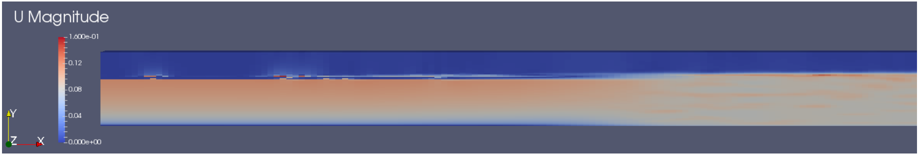
```
foamListTimes -rm
yes | rm log.*
rm -rf postProcessing
rm -rf waveGaugesNProbes

setWaveParameters
setWaveField
waveFoam
postProcess -func sample
```

shearCurrent tutorial

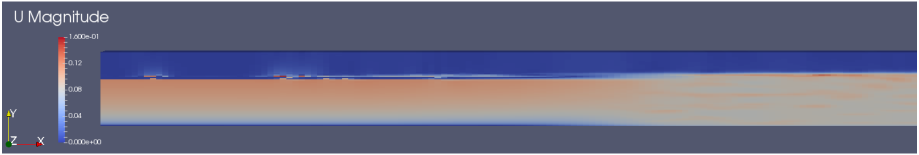
The graphical result of the `shearCurrent` tutorial at $t = 10$ s is shown below.





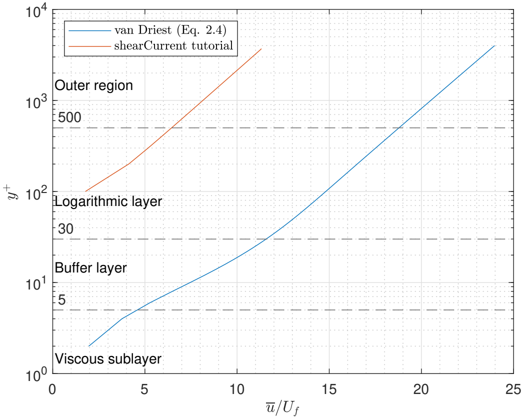
The mean velocity profile is increasing from the wall to the free surface which is expected from the van Driest velocity profile. Although it seems that the `shearCurrent` library is giving reasonable results, one can notice problems. It can be seen that the fluid section of the computational domain is not homogeneous considering the van Driest velocity profile, where the velocities of the entire section should be 0 at the wall, then increasing up to the free surface like the inlet section.

shearCurrent tutorial



BUT: One can notice that there are some nonphysical small velocity pockets inside the fluid section. The reason for this problem can be attributed to a bug inside of the newly implemented `shearCurrent` wave theory library or as `waves2Foam` toolbox is also manipulating the flow velocities at the background considering the relaxation zones etc., there can be problems regarding the `shearCurrent` library which is not fully adapted considering the internal functionalities in the `waves2Foam` toolbox.

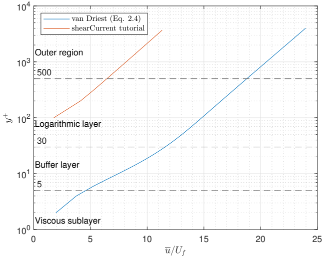
shearCurrent tutorial



Comparison of `shearCurrent` tutorial sampled (inlet) result at $t = 10$ s with van Driest profile [SF20]

Additionally, it can be seen that although the slopes of the simulated and theoretical velocity profiles are nearly the same for logarithmic layer and outer region sections, they are shifted from each other.

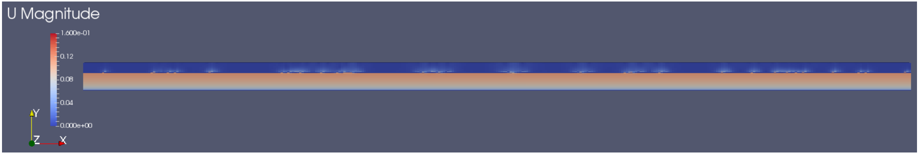
shearCurrent tutorial



A possible reason for this problem is the mesh density of the tutorial which is not dense enough to resolve the section close to the wall. As van Driest's solution uses cumulative integration, the streamwise velocity values at that unresolved section could not be added to the computation, therefore due to the lack of these velocity values contribution to the van Driest velocity profile, there is a shift between the computed and theoretical results. Although this may be a possible problem, due to the computational time, the mesh density has not been increased as the simulation has not finished within a reasonable time.

shearCurrent tutorial

As a different approach to see the reaction of waves2Foam toolbox to try to alter possible mentioned problems, the relaxation zone for the inlet and outlet area is increased where both relaxation zones are structured as fully covering the entire fluid region (from startX=0 to endX=18 m).



It can be seen that the fluid domain is showing expected results considering the velocity increase from the wall to the free surface for the entire fluid region, homogeneously. Although this shows that the shearCurrent theory is working, as the relaxation zone is set for the entire domain, the shearCurrent theory is not projected across the domain as an initial condition. Therefore, one should consider the possible problems for the shearCurrent library and tutorial.

Thank you for your kind attention!

References

N. G. Jacobsen, *waves2foam manual*, 2017.

B. M. Sumer and D. R. Fuhrman, *Turbulence in coastal and civil engineering*, World Scientific, 2020.