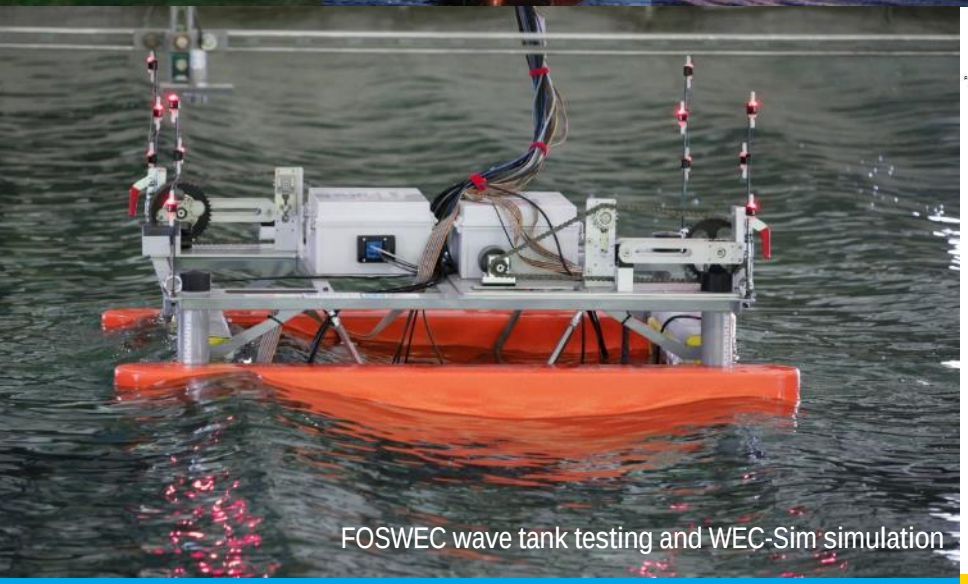
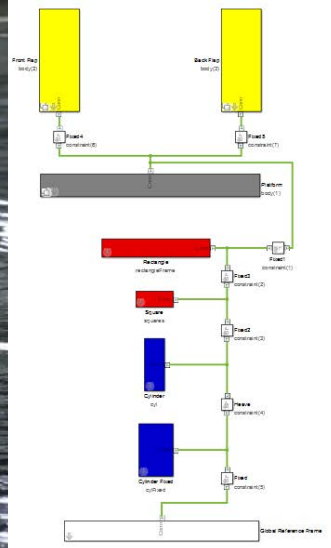
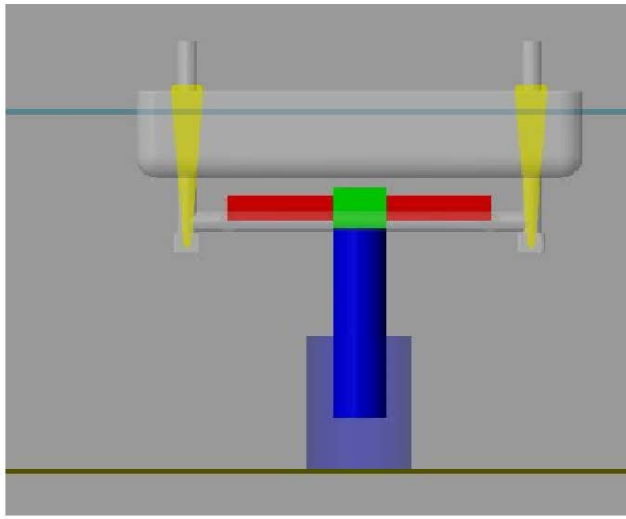
U.S. DEPARTMENT OF ENERGY | Energy Efficiency & Renewable Energy



FOSWEC wave tank testing and WEC-Sim simulation





WEC-Sim Webinar #4

Mooring and Visualization

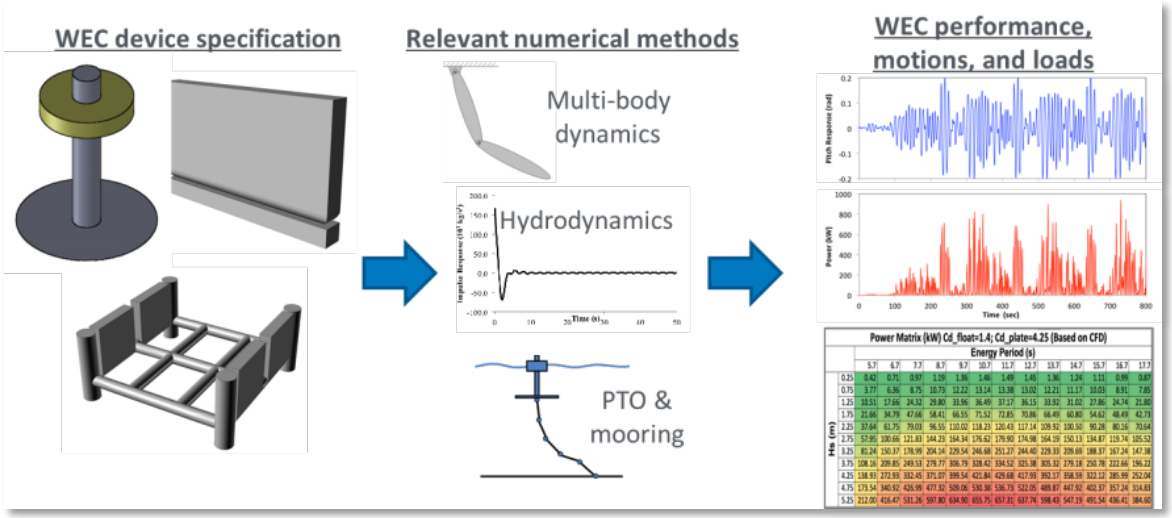
July 18, 2017

Yi-Hsiang Yu (NREL)
Kelley Ruehl (Sandia)

Introduction

WEC-Sim Team

- Kelley Ruehl (Sandia)
- Yi-Hsiang Yu (NREL)
- Jennifer van Rij (NREL)



WEC-Sim Updates

Reduced the size of the repo

- Remove publications from the repo since all the publications are available from the website http://wec-sim.github.io/WEC-Sim/publications.html
- Working on removing the large data file (e.g., *.h5 and *.mat) from repo history

Created a WEC-Sim_Application submodule in WEC-Sim

- This can be pulled into WEC-Sim currently if desired https://github.com/WEC-Sim/WEC-Sim_Applications
- Cleaned up PTO-Sim application cases

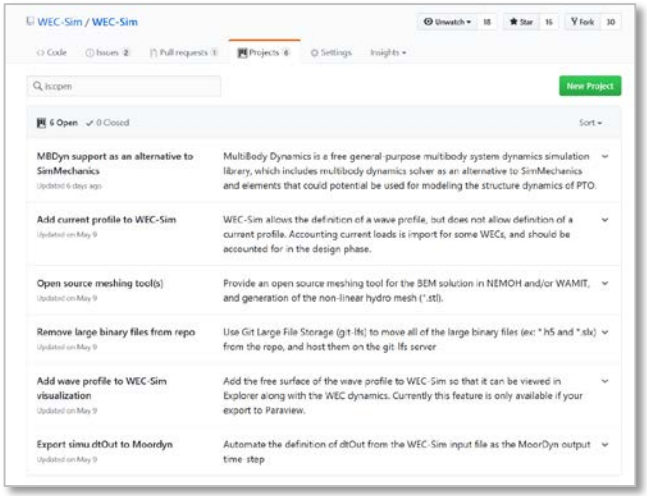
WEC-Sim Updates

Created a separate moorDyn library repo

- Due to different licenses, moorDyn is now saved in another repository
- To use MoorDyn in WEC-Sim, download moorDyn from repo https://github.com/WEC-Sim/moorDyn
- Place all the files and folders under WEC-Sim/source/functions/moorDyn folder

Added a WEC-Sim Projects Page

- Can be used to track requested feature additions, and their status https://github.com/WEC-Sim/WEC-Sim/projects



Schedule

Advanced Feature Webinars *1hr each*

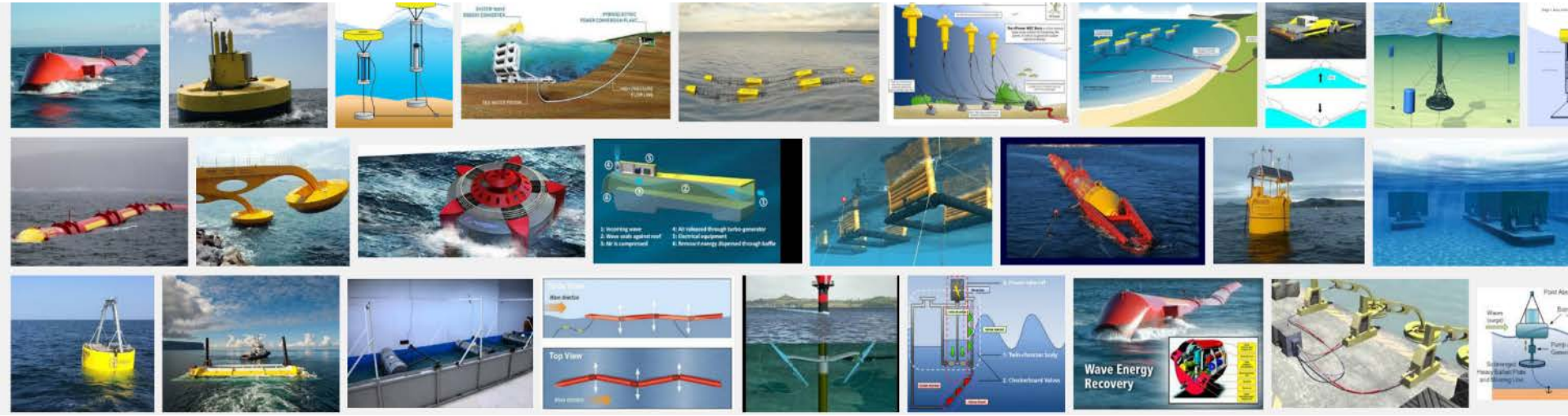
- **April 18:** bemio and mcr, application for power matrix
- **May 24:** nl-hydro, b2b, non-hydro ~~and drag~~
- **June 13:** pto and control, application for desalination
- **July 18:** mooring and visualization
- ***Available Online***: http://wec-sim.github.io/WEC-Sim/webinars.html

Training Courses

- **May 1:** *1hr* WEC-Sim workshop at METS, for new users
- **August 17:** *half-day* WEC-Sim code structure course, for advanced users/developers

WEC-Sim Webinar #4

U.S. DEPARTMENT OF ENERGY | Energy Efficiency & Renewable Energy



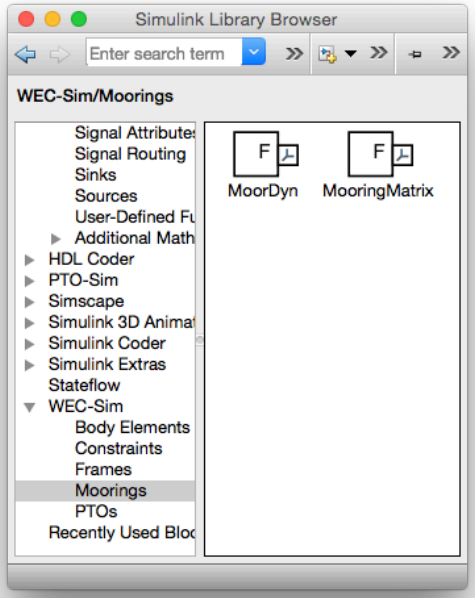
Mooring Applications (Mooring Matrix / MoorDyn)

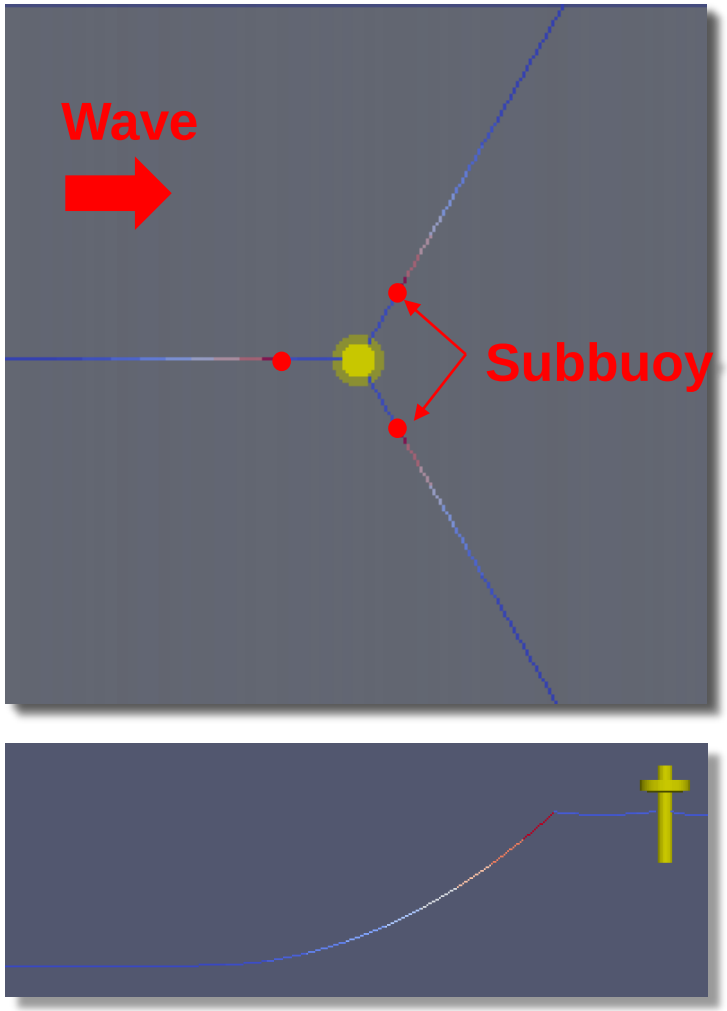
Yi-Hsiang (NREL)

Mooring Dynamics Modeling

- Floating WEC systems are often connected to mooring lines to keep the device in position.

- WEC-Sim allows the user to model the mooring dynamics in the simulation by specifying the Mooring Matrix or coupling with MoorDyn.





WEC-Sim Theory

- Dynamics simulated by solving time-domain equation of motion (Cummins, 1962)

$$m\ddot{x}(t) = f_{hs}(t) + f_{ex}(t) + f_{rad}(t) + f_{v}(t) + f_{pto}(t) + f_{m}(t)$$

Hydrostatic restoring force

Wave excitation & diffraction force (from BEM simulations)

Radiation force: added mass and radiation damping (from BEM simulations)

Viscous force

Power take-off force

Mooring force

- Use radiation and diffraction method and calculate the hydrodynamic forces from frequency-domain Boundary Element Method (BEM)

$$f_{rad}(t) = -A_{\infty}\ddot{X} - \int_0^t K(t-\tau)\dot{X}(\tau)d\tau$$

BEM BEM

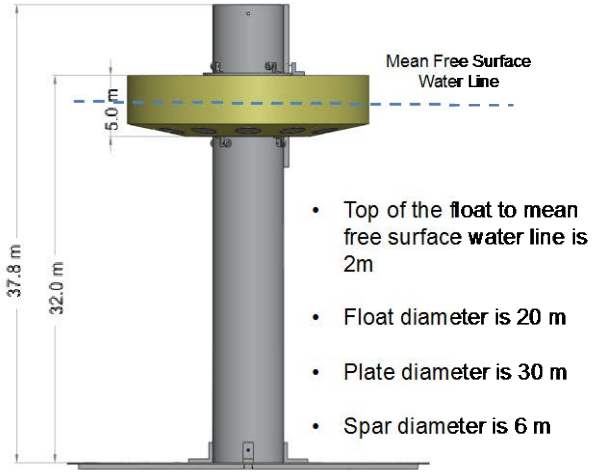
$$f_{ex}(t) = \Re\left[R_f F_X(\omega_r) e^{i(\omega_r t+\phi)} \int_0^{\infty} \sqrt{2S(\omega_r)d\omega_r}\right]$$

BEM

$$= \int_{-\infty}^{\infty} \eta(\tau) f_e(t-\tau)d\tau$$

BEM

Examples: Reference Model 3 (RM3) Using Mooring Matrix



- Top of the float to mean free surface water line is 2m
- Float diameter is 20 m
- Plate diameter is 30 m
- Spar diameter is 6 m

	Center of Gravity (m)	Mass (Ton)	Moment of Inertia (Ton-m^2)		
			20900	0	0
Float	[0, 0, -0.72]	727.01	0	21300	4.3
			0	4.3	37100
			137000	0	0
Spar Plate	[0, 0, -21.29]	878.3	0	137000	218
			0	218	28500

Specify linear mooring matrix

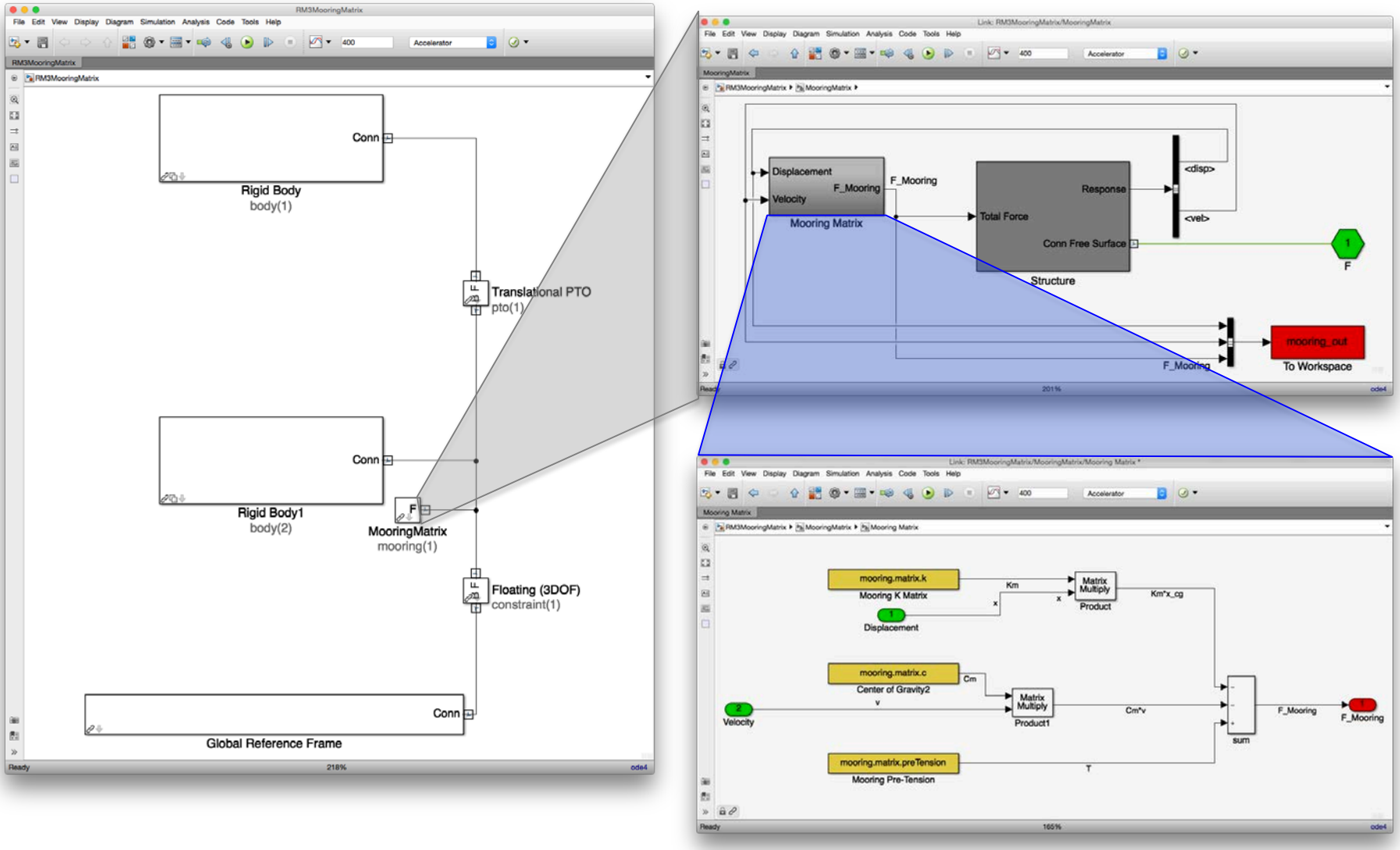
$$\text{Mooring Stiffness } K=\begin{bmatrix} & \cdots & \\ \vdots & 6X6 & \vdots \\ & \cdots & \end{bmatrix}$$

$$\text{Mooring Damping } C=\begin{bmatrix} & \cdots & \\ \vdots & 6X6 & \vdots \\ & \cdots & \end{bmatrix}$$

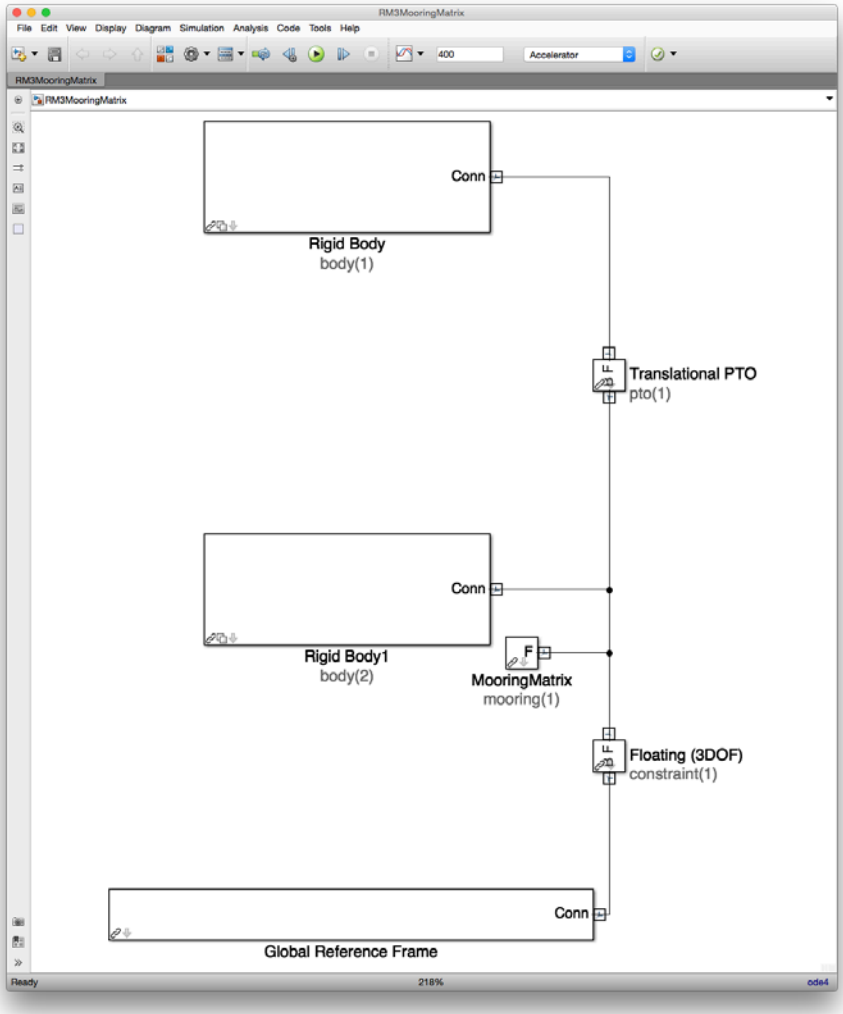
$$\text{Mooring Pretension } T=[\quad 1X6 \quad]$$

Available at:
https://github.com/WEC-Sim/WEC-Sim_Applications

Examples: Reference Model 3 (RM3) Using Mooring Matrix



Examples: Reference Model 3 (RM3) Using Mooring Matrix



- When the mooring matrix block is used, the user first needs to initiate the mooring class by setting

  mooring(i) = mooringClass('mooring name')

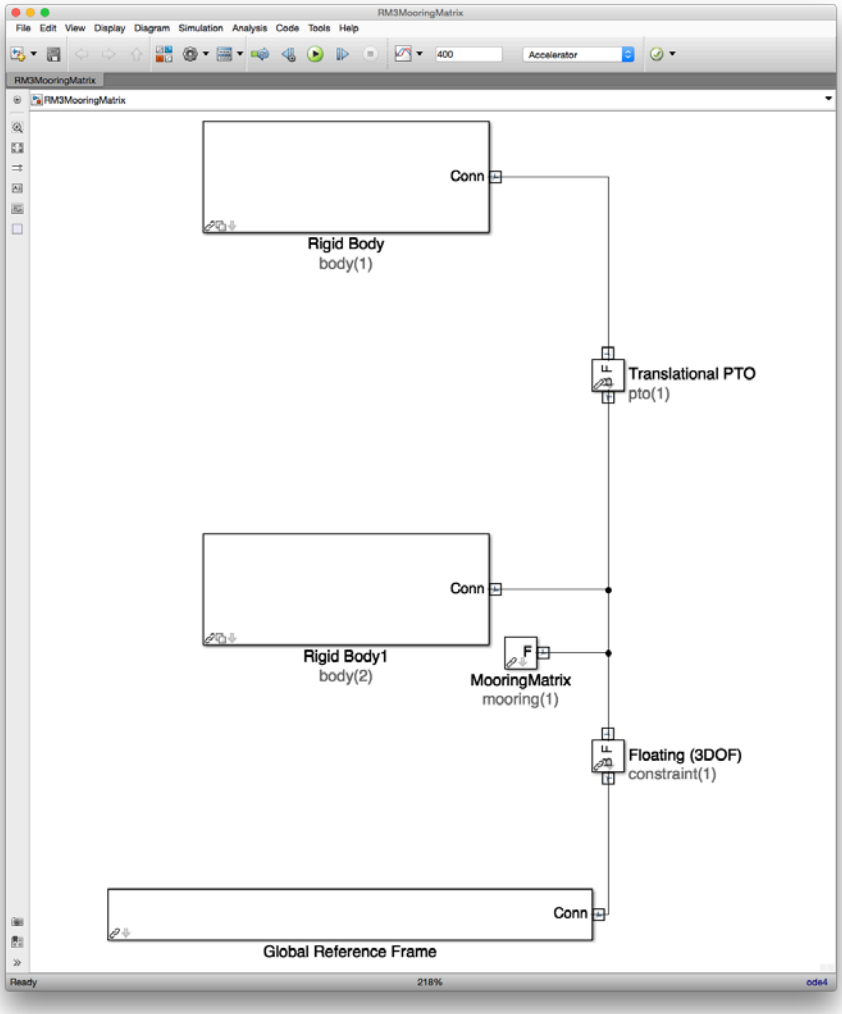
  in the WEC-Sim input file (wecSimInputFile.m)

- Typically, the mooring connection location also need to be specified,

  mooring(i).ref = [1x3]

  Default location is [0 0 0]

RM3 - Using Mooring Matrix

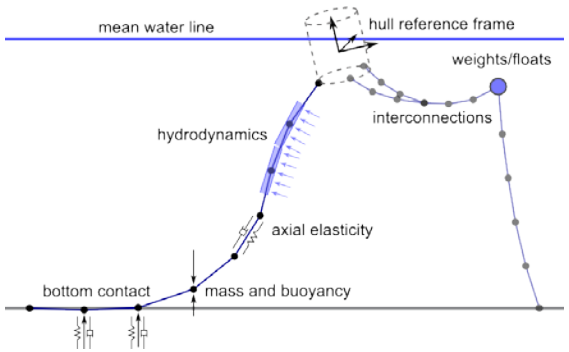


Editor - /Users/yyu/Desktop/RM3_MooringMatrix/wecSimInputFile.m

wecSimInputFile.m

```
%% Simulation Data
simu = simulationClass();
simu.simMechanicsFile = 'RM3MooringMatrix.slx';      % Location of Simulink Model F
simu.mode='accelerator';
simu.explorer = 'off';
simu.rampT = 40;
simu.endTime=400;
simu.dt = 0.01;
simu.dtCITime = 0.05;

%% Wave Information
% User-Defined Time-Series
waves = waveClass('userDefined');               % Create the Wave Variable and Spec
waves.etaDataFile = 'umpqua46229_6_2008.mat';   % Name of User-Defined Time-Series

%% Body Data
% Float
body(1) = bodyClass('hydroData/rm3.h5');
body(1).geometryFile = 'geometry/float.stl';
body(1).mass = 'equilibrium';
body(1).momOfInertia = [20907301 21306090.66 37085481.11];

% Spar/Plate
body(2) = bodyClass('hydroData/rm3.h5');
body(2).geometryFile = 'geometry/plate.stl';
body(2).mass = 'equilibrium';
body(2).momOfInertia = [94419614.57 94407091.24 28542224.82];
body(2).initDisp.initLinDisp = [0 0 -0.21];  % Initial Displacement

%% PTO and Constraint Parameters
% Floating (3DOF) Joint
constraint(1) = constraintClass('Constraint1');
constraint(1).loc = [0 0 0];

% Translational PTO
pto(1) = ptoClass('PTO1');
pto(1).k=0;
pto(1).c=1200000;
pto(1).loc = [0 0 0];

%% Mooring
% Mooring Matrix
mooring(1) = mooringClass('mooring');       % Initialize mooringClass
mooring(1).matrix.k = zeros(6,6);
mooring(1).matrix.k(1,1) = 1e5;
mooring(1).matrix.c = zeros(6,6);
mooring(1).matrix.preTension = zeros(1,6);
```

MoorDyn

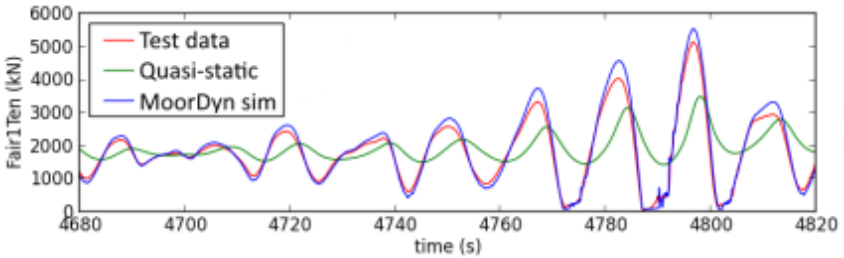


- Simple, flexible, open-source mooring dynamics
  - lumped mass, axial elasticity, Morison hydrodynamics
  - arbitrary line interconnections, weights, floats
  - coded in C++ for speed and coupling as a DLL
- MoorDyn allows modeling of interconnections between lines, and weight or buoyancy elements at the connections.

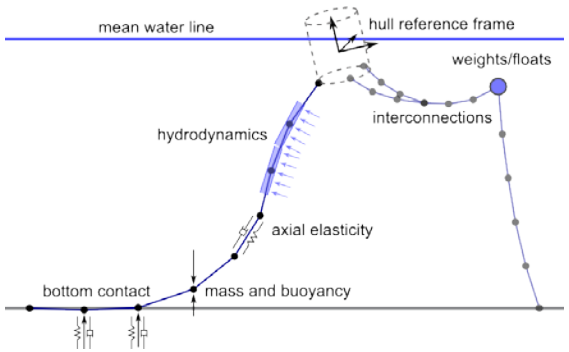
Developed by Matt Hall
(University of Prince Edward Island) mthall@upei.ca

DLLs, Source, User's Guide, Examples
www.matt-hall.ca/software/MoorDyn

- Demonstration of these capabilities has been done for model-scale tests of a pitching WEC (Vissio, et al., EWTEC 2015).
- This modeling approach was successfully validated for 1/50-scale floating wind turbine test data (Hall and Goupee, Ocean Engineering, 2015).



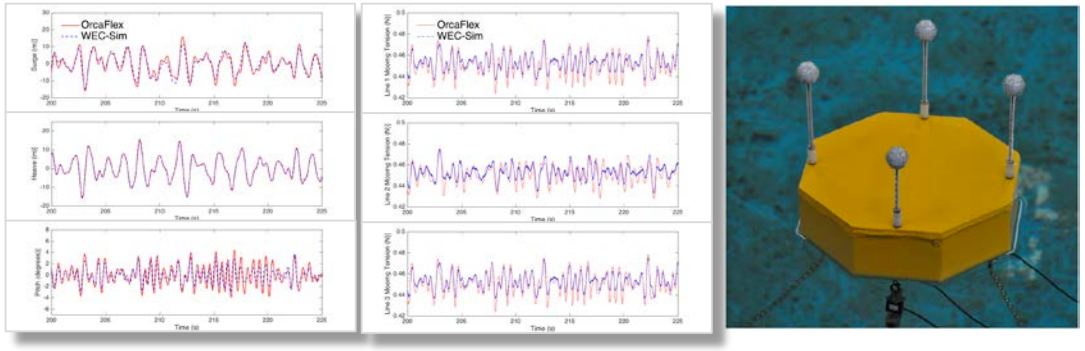
MoorDyn



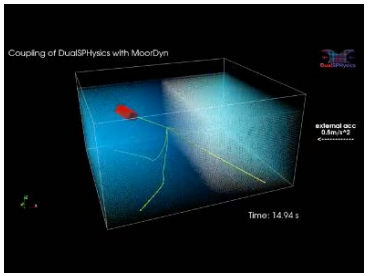
Developed by Matt Hall
(University of Prince Edward Island) mthall@upei.ca

DLLs, Source, User's Guide, Examples
www.matt-hall.ca/software/MoorDyn

- Coupled WEC-Sim/MoorDyn model has been verified and validated with modeling results from OrcaFlex and measurement data from wave tank test (Sirnivas et al. OMAE2016)



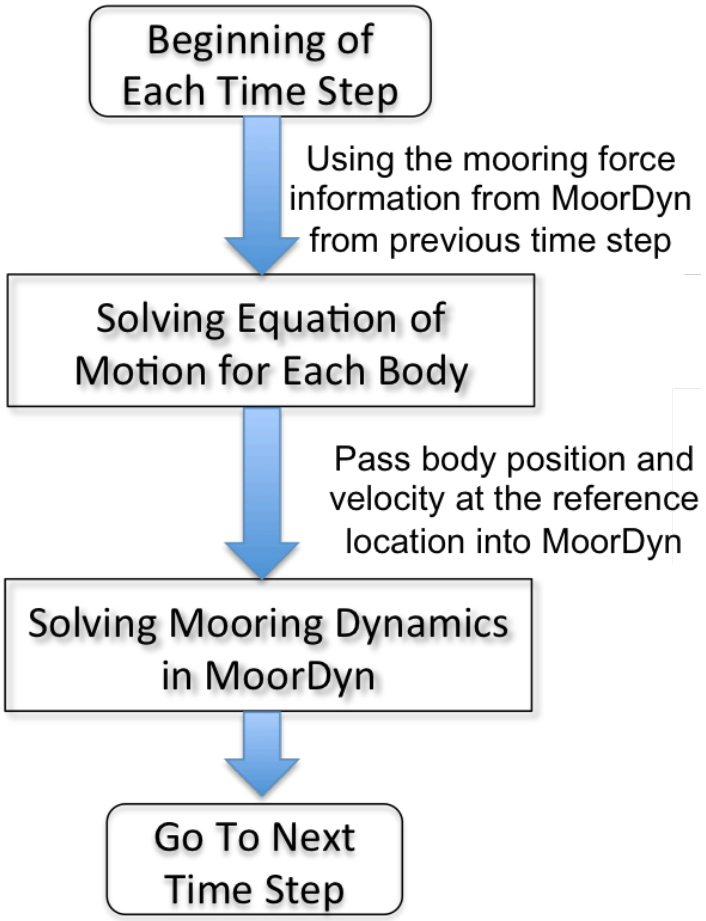
- Other Application:
  - MoorDyn has been coupled with a smoothed particle hydrodynamics code, DualSPHysics, to simulate complex mooring dynamics



https://www.youtube.com/watch?v=ToEsV0FDs0I

WEC-Sim/MoorDyn Coupling

- WEC-Sim is coupled with MoorDyn using a loose-coupling approach.
- WEC-Sim and MoorDyn exchange the forces and response information at a specified reference point that moves with the floating body.
- MoorDyn typically operates at a smaller time step than floating platform models.
- The loose coupling approach generally requires a small time step in the coupled WEC-Sim/MoorDyn simulations.



MoorDyn Installation

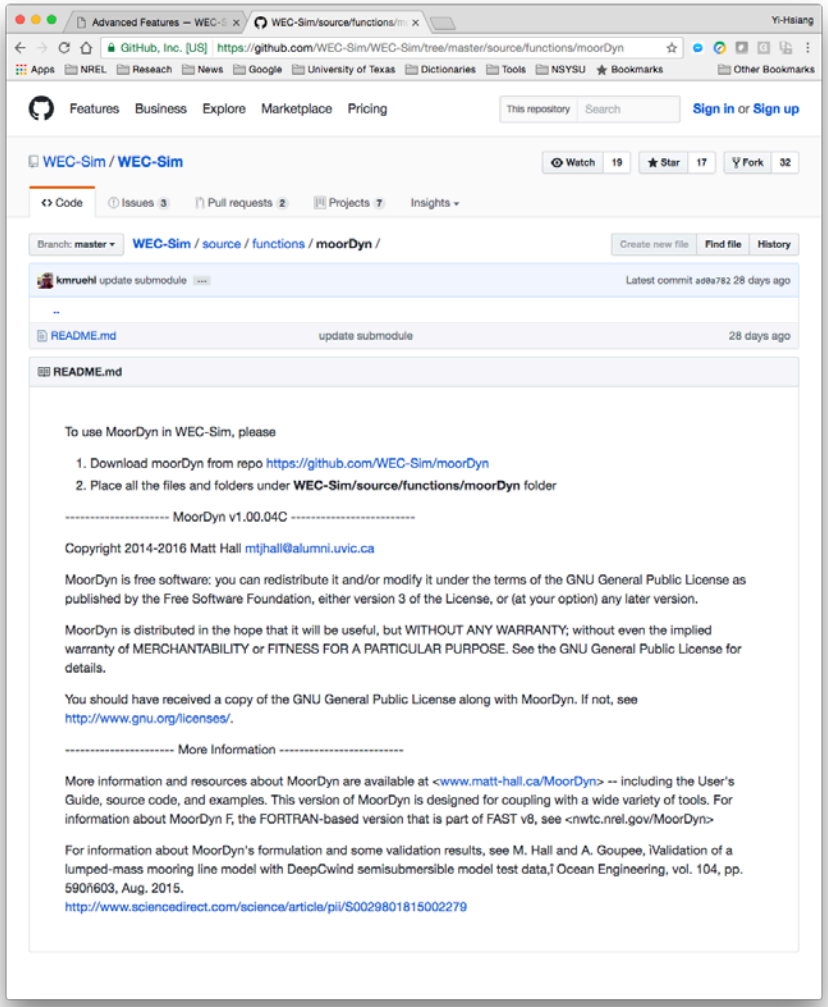
- MoorDyn (complied lib) is hosted on a separate MoorDyn repository. https://github.com/WEC-Sim/moorDyn

- It must be download separately, and all files and folders should be placed in the WEC-Sim/source/functions/moorDyn folder.

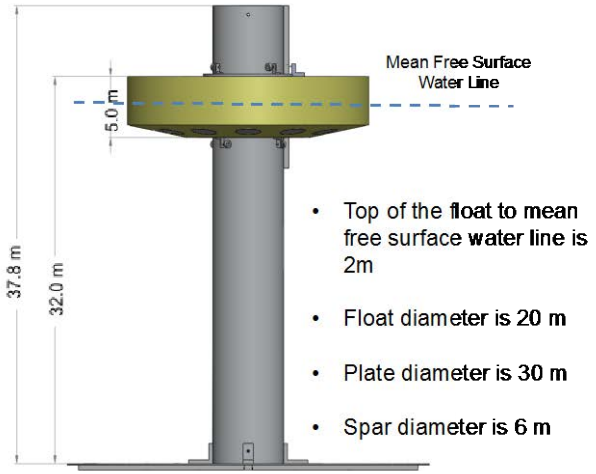
- A C/C++ compiler is required for for Windows operating system

  http://www.mathworks.com/matlabcentral/fileexchange/52848-matlab-support-for-mingw-w64-c-c++-compiler

MinGW-w64

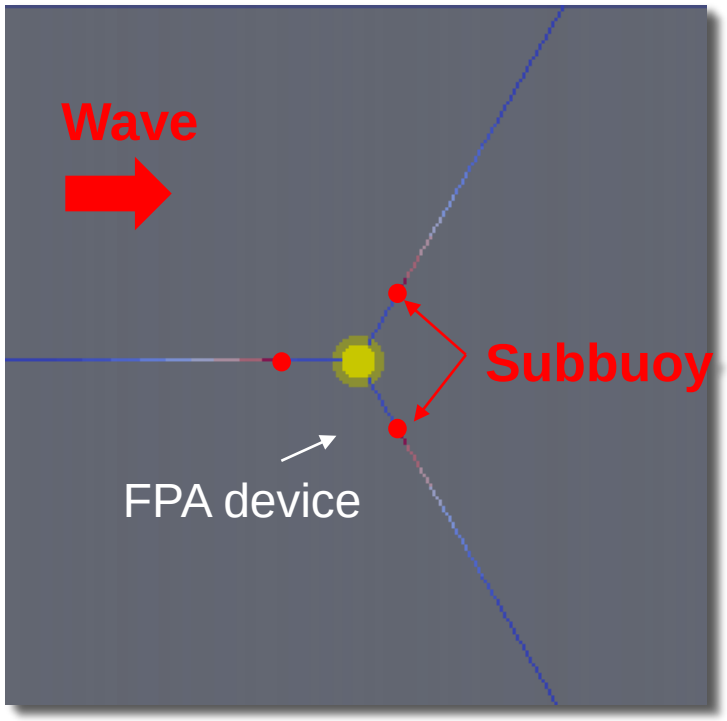


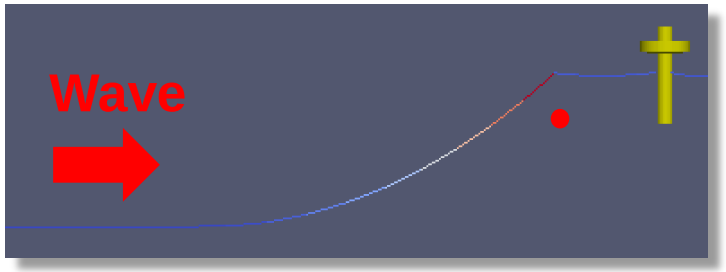
Examples: Reference Model 3 (RM3) Using MoorDyn



- Top of the float to mean free surface water line is 2m
- Float diameter is 20 m
- Plate diameter is 30 m
- Spar diameter is 6 m

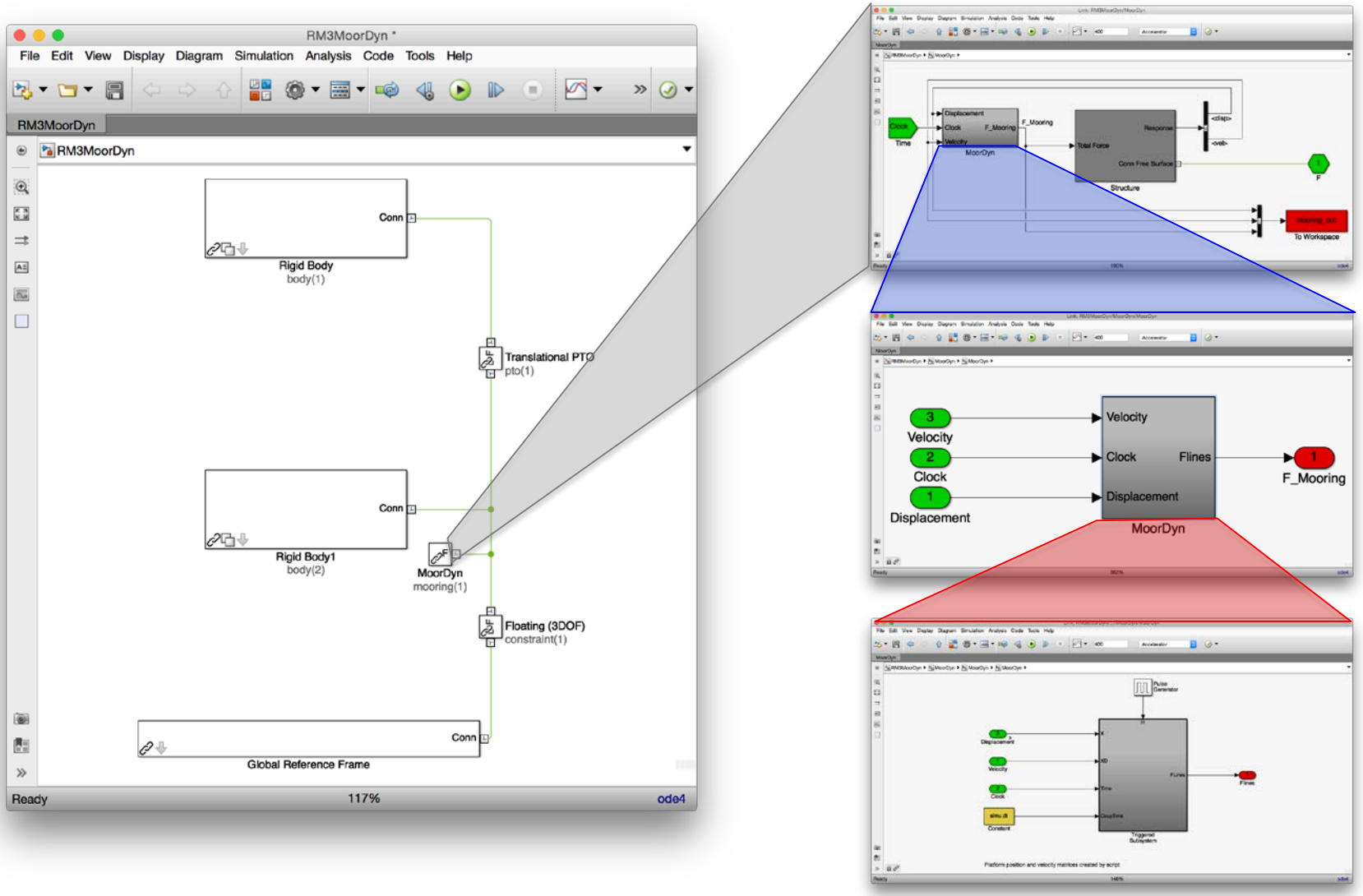
	Center of Gravity (m)	Mass (Ton)	Moment of Inertia (Ton-m^2)		
			20900	0	0
Float	[0, 0, -0.72]	727.01	0	21300	4.3
			0	4.3	37100
			137000	0	0
Spar Plate	[0, 0, -21.29]	878.3	0	137000	218
			0	218	28500



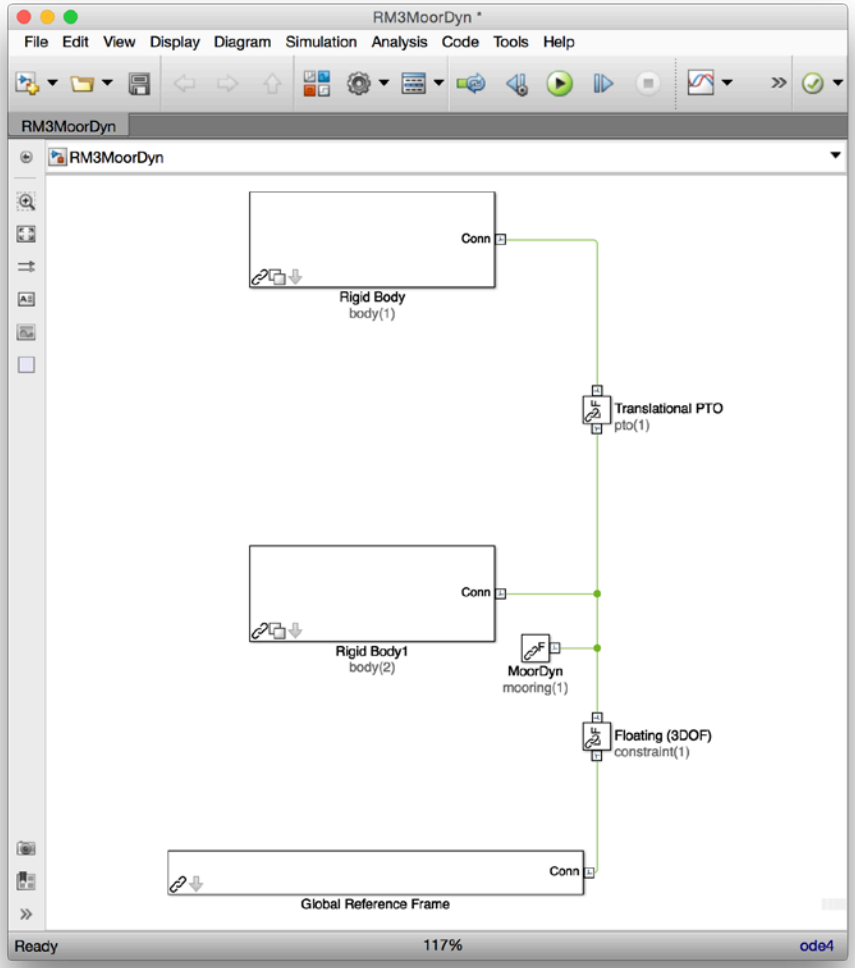


Available at:
https://github.com/WEC-Sim/WEC-Sim_Applications

Examples: Reference Model 3 (RM3) Using MoorDyn



Examples: Reference Model 3 (RM3) Using MoorDyn



- When the moorDyn block is used, the user also needs to initiate the mooring class by setting

  mooring(i) = mooringClass('mooring name')

  in the WEC-Sim input file (wecSimInputFile.m)

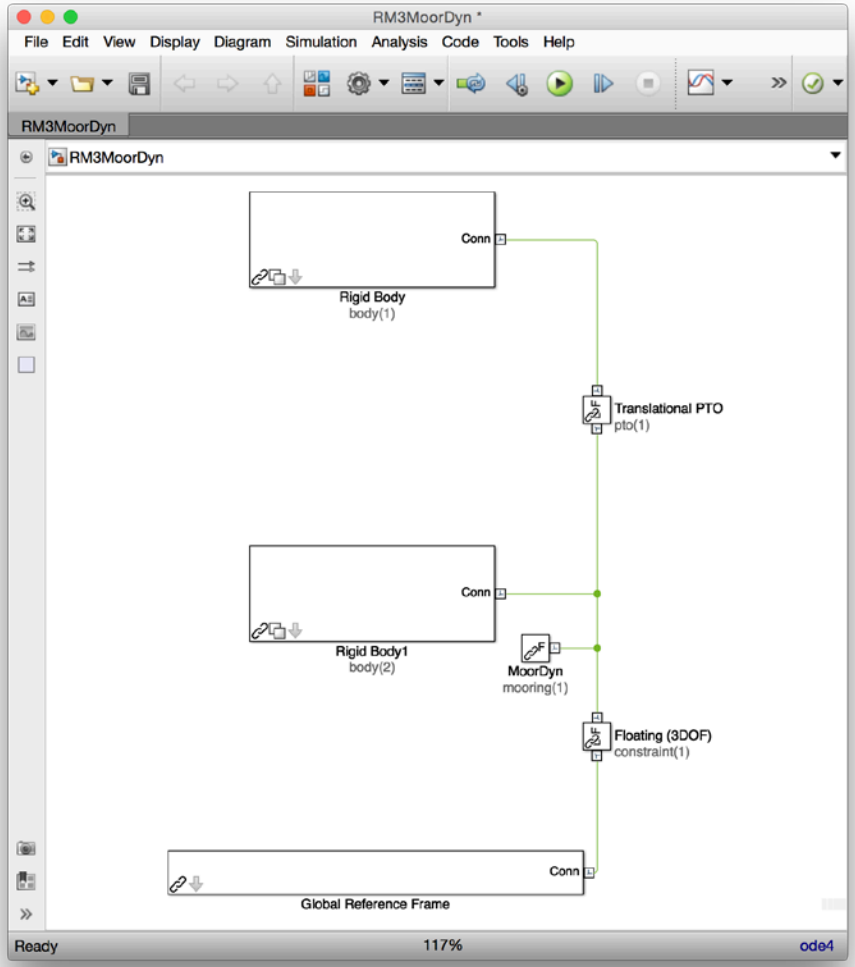
- Followed by number of mooring lines is defined in MoorDyn

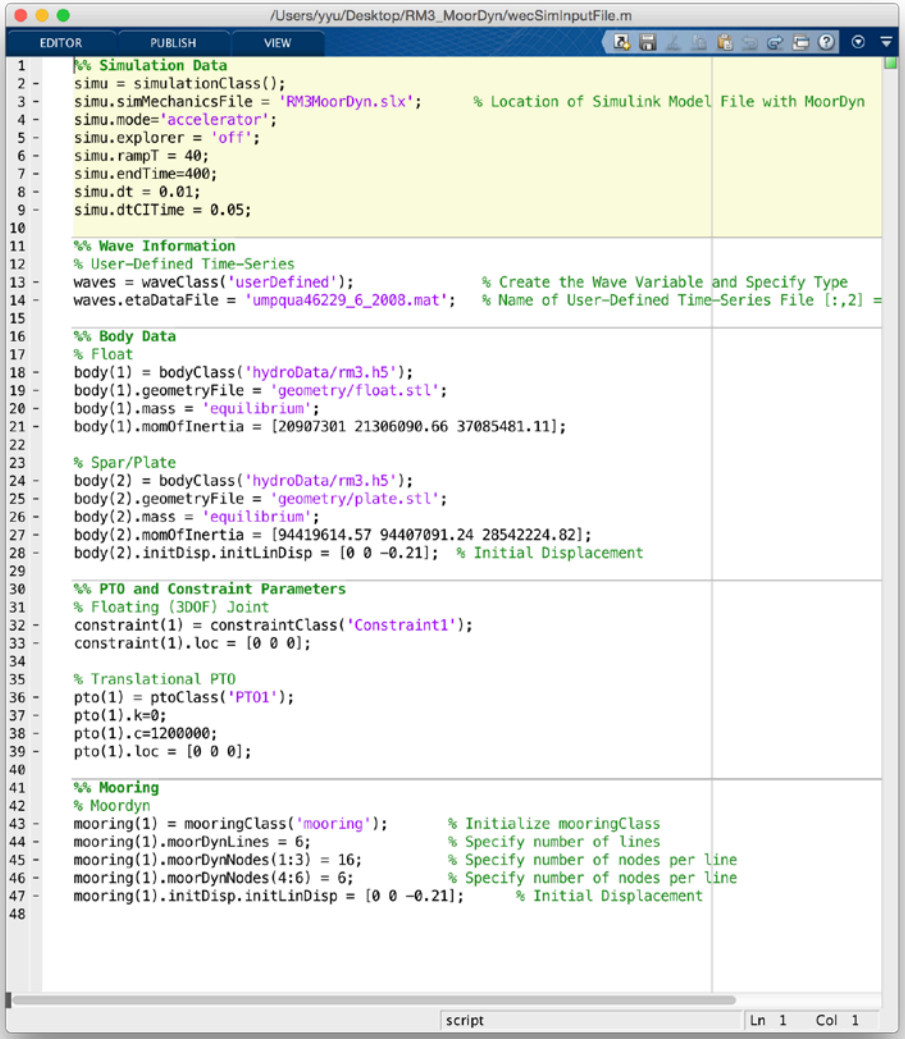
  mooring(1).moorDynLines = <Number of lines>

- A mooring folder that includes a moorDyn input file (lines.txt) is required in the simulation folder.

- The RM3 with MoorDyn folder is located under WEC-Sim Applications repository.

Examples: Reference Model 3 (RM3) Using MoorDyn





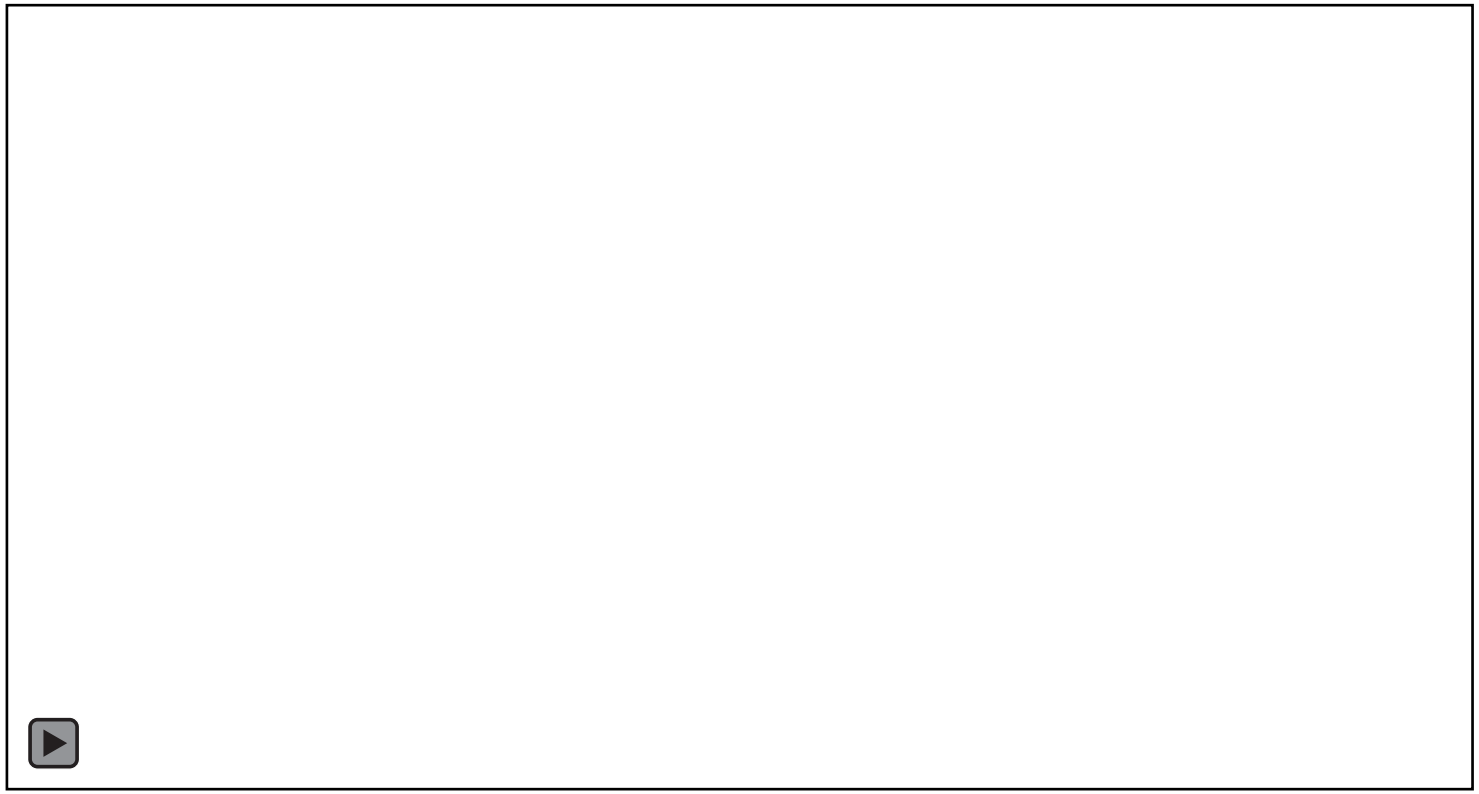
Examples: Reference Model 3 (RM3) Using MoorDyn

- Requires smaller time step size.
- Works with simu.mode='accelerator' but not simu.mode='rapid-accelerator';

  Unable to build a standalone executable to simulate the model 'RM3MoorDyn' in rapid accelerator mode.
- Subbuoys can be simulated using "Connect" in MoorDyn
- WEC-Sim/MoorDyn coupling only allow one set of moorings

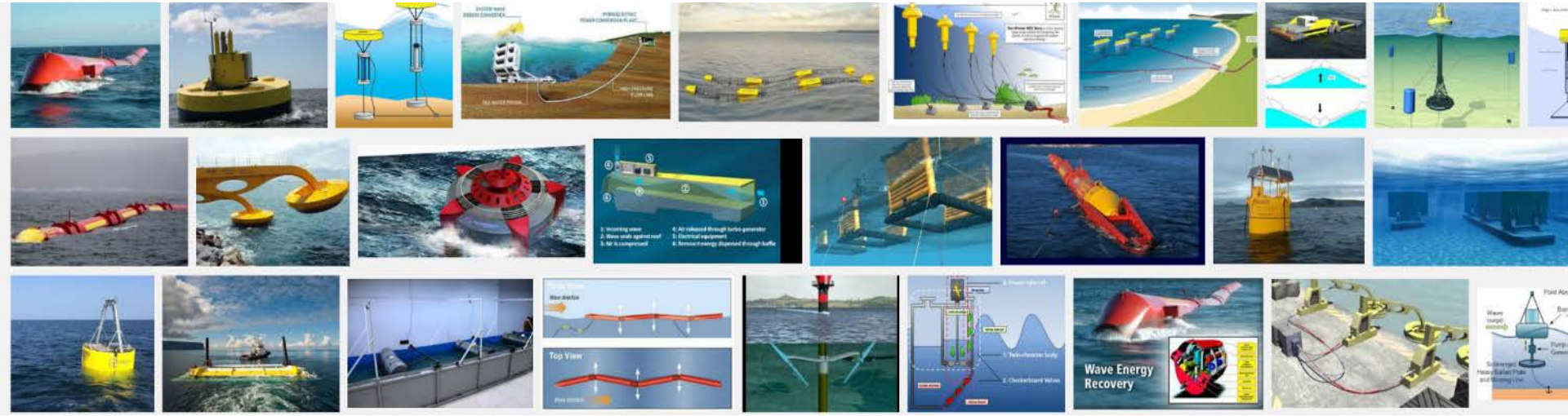
Examples: Reference Model 3 (RM3) Using MoorDyn

- The simulation was conducted using a 400-s time series (measured in 2008) from the National Data Buoy Center buoy #46229 located off the coast of Oregon.



WEC-Sim Webinar #4

U.S. DEPARTMENT OF ENERGY | Energy Efficiency & Renewable Energy

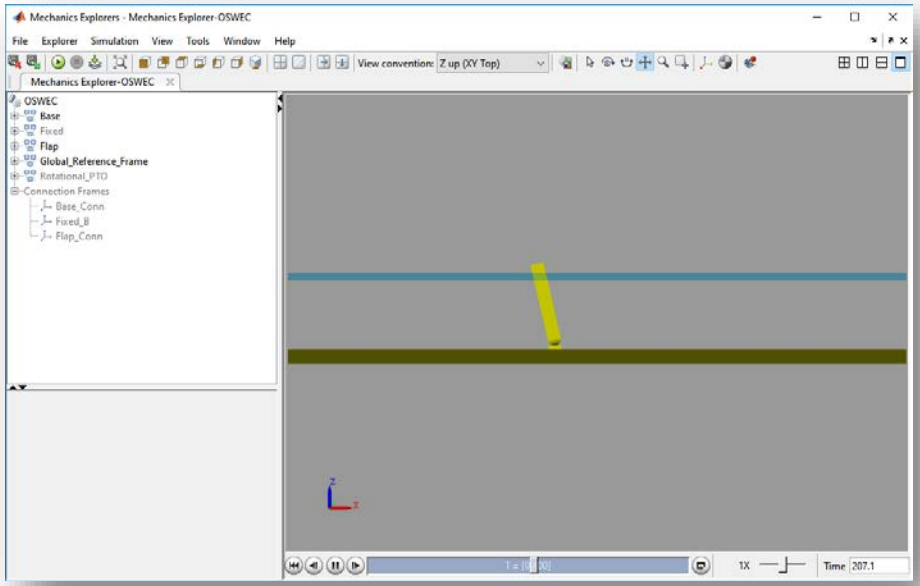


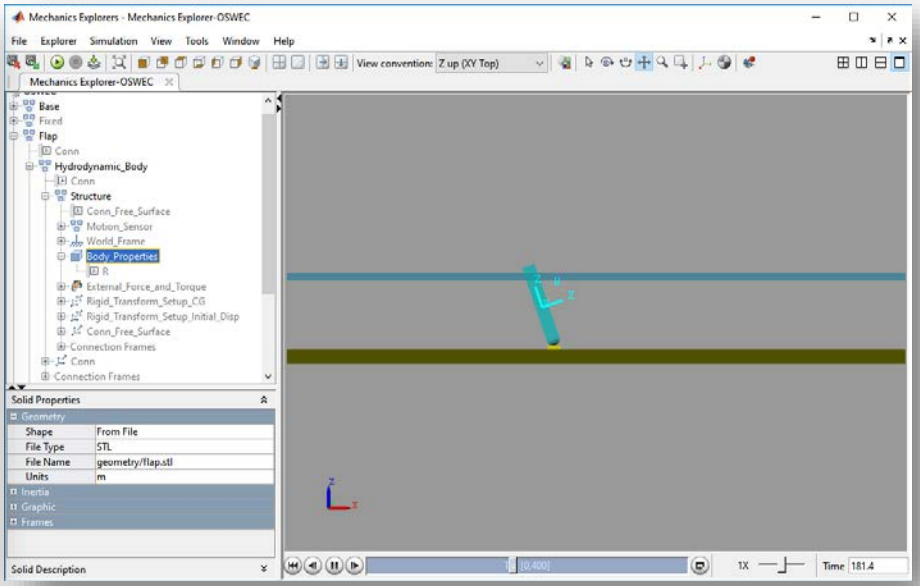
Visualization with Simscape and ParaView

Kelley Ruehl (Sandia)

Visualization with Simscape

- WEC-Sim default visualization uses Simscape
  - **simu.explorer = 'on'** turns explorer on (default is on)
  - Can't use Simscape viz with accelerator or rapid-accelerator
- Simscape provides animation of WEC-Sim run
  - Shows rigid body motion, cg, coordinates, etc
  - Can create video
  - Does not show wave surface elevation, etc



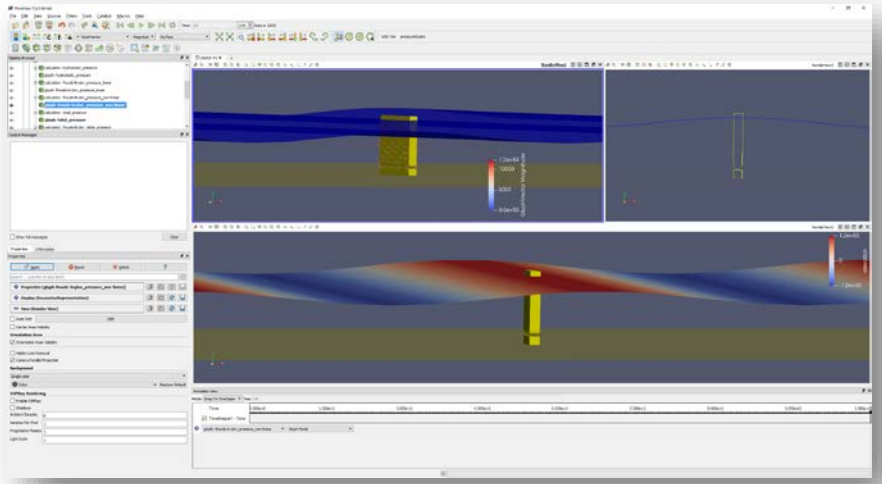


OSWEC Visualization in Simscape

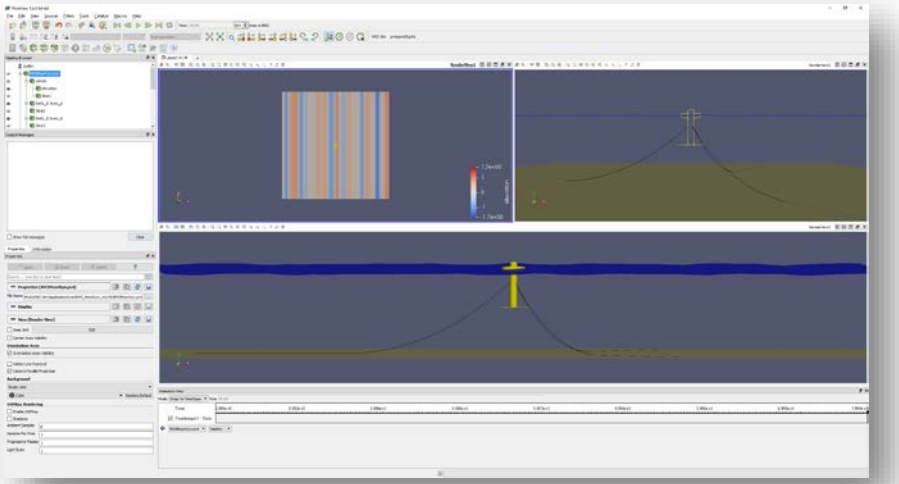
Visualization with ParaView

- ParaView is an open-source, data analysis and visualization application.
  - Pro: WEC-Sim uses ParaView to create videos, visualize wave field, cell-by-cell non-linear forces, and other features.
  - Con: Takes *a lot* more time to run
- Examples using ParaView for WEC-Sim data visualization available on the applications repository:

  https://github.com/WEC-Sim/WEC-Sim_Applications



OSWEC Nonlinear Viz with ParaView



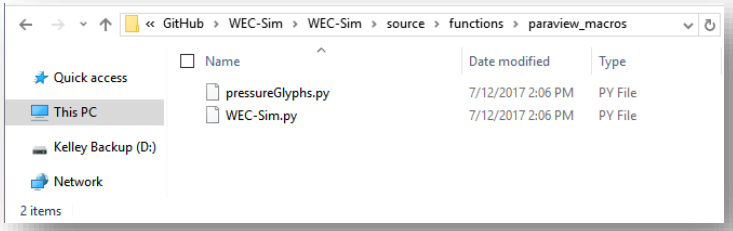
RM3 MoorDyn Viz with ParaView

Installing ParaView

- Download and Install ParaView
  - For ParaView visualization https://www.paraview.org/

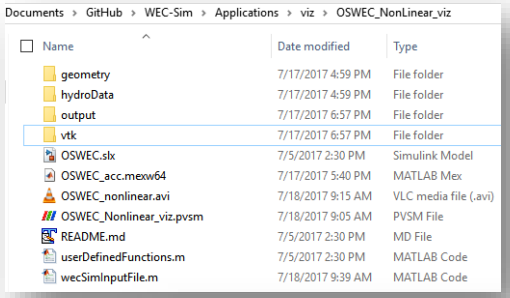
- Download and Install Python
  - For ParaView macros https://www.paraview.org/Wiki/ParaView_and_Python

- Install the WEC-Sim macros in ParaView
  - Open ParaView
  - Click on **'Macros => Add new macro'**
  - Navigate to the ***WEC-Sim/source/functions/paraview_macros*** directory
  - Select ***WEC-Sim.py*** macro file and click 'OK'
  - Select ***pressureGlyphs.py*** macro file and click 'OK'

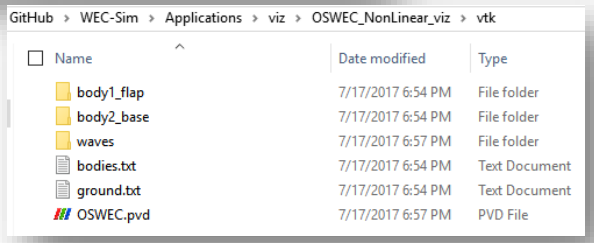


WEC-Sim ParaView Visualization

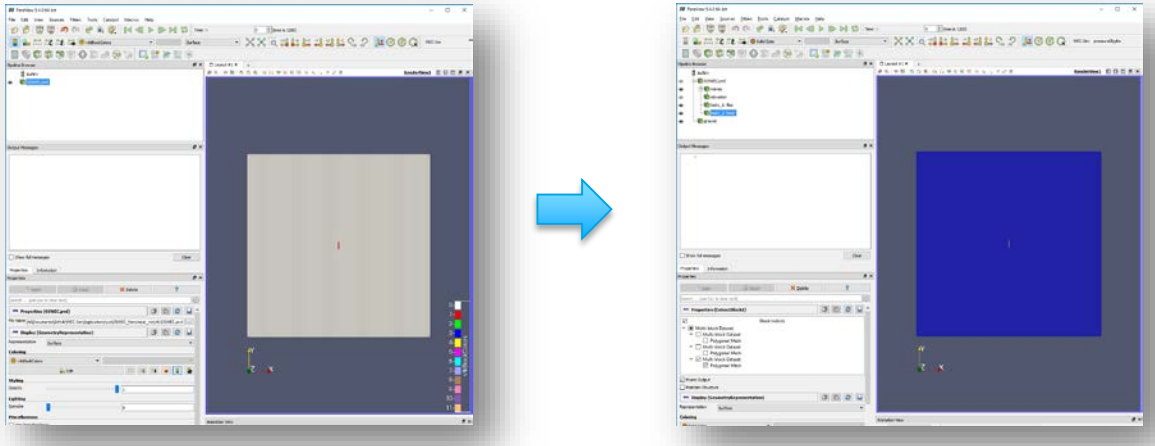
- **‘simu.paraview = 1’** turns Paraview on
  - creates a ***/vtk*** directory in the WEC-Sim case directory
  - saves ParaView data files



- Open the ***.pvd** file to view data visualization in ParaView and Click **‘Apply’**

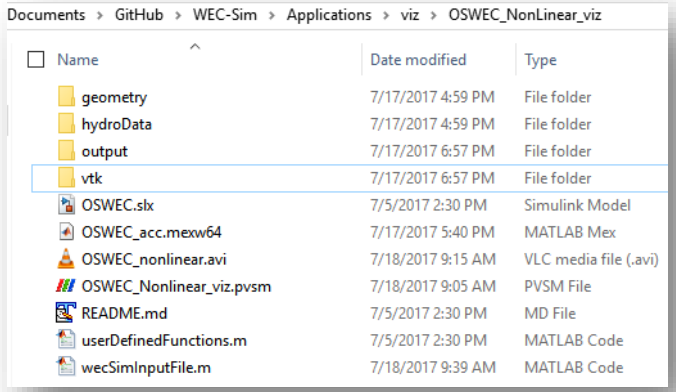


- Select model in pipeline and run the ‘**WEC-Sim’** macro to import default WEC-Sim colors and orientation



Non-Linear Hydro, videos, etc

- For non-linear hydro, select body in pipeline and run the **'pressureGlyphs'** macro, adds glyphs for:
  - Hydrostatic Pressure
  - Linear Froude-Krylov pressure
  - Non-linear Froude-Krylov pressure
  - Total pressure (hydostatic plus non-linear Foude-Krylov)
  - Froude-Krylov delta (non-linear minus linear)
- Add multiple views, slices and data filters
- Save State to recreate viz for many runs
  - 'File => Save State', saves state as a **.pvsm*
- Create video file
  - 'File => Save Animation', saves video as **.avi*

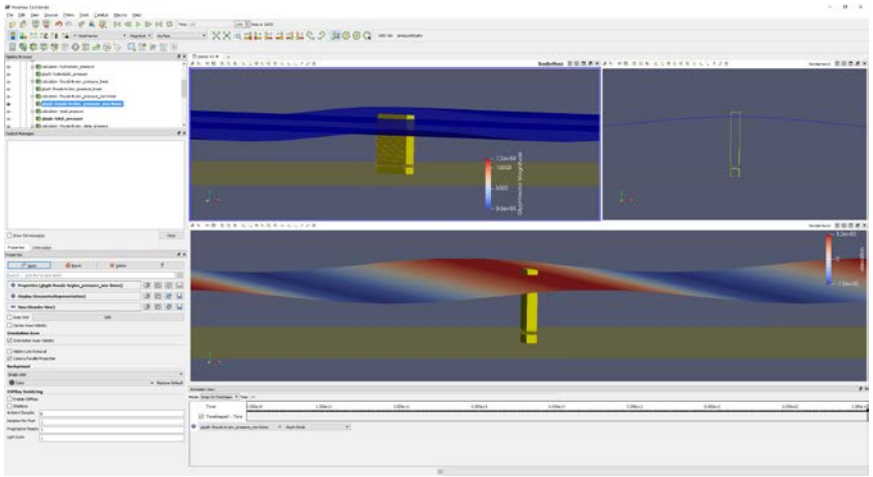


Visualization with ParaView

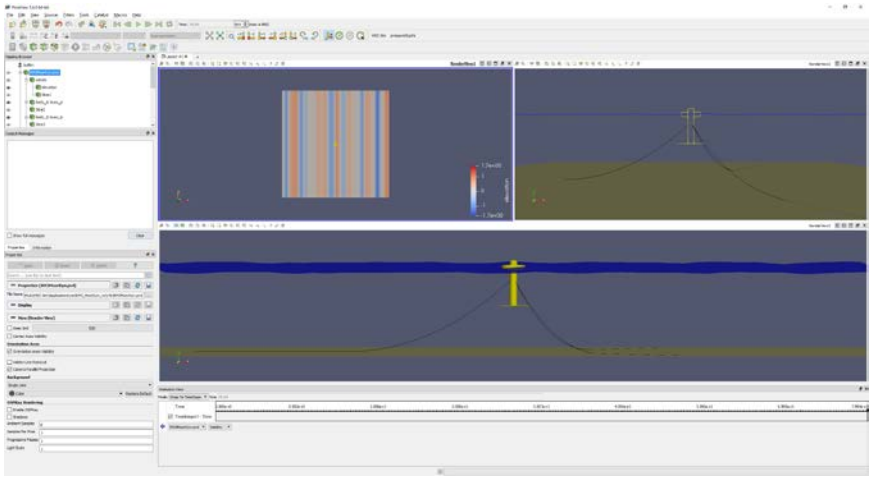
- Examples using ParaView for WEC-Sim data visualization available on the Applications repository:

  https://github.com/WEC-Sim/WEC-Sim_Applications

- 'Viz' Examples include
  - OSWEC with non-linear hydro
  - RM3 coupled with MoorDyn



OSWEC_Nonlinear_Viz



RM3_MoorDyn_Viz

ParaView for RM3 coupled with MoorDyn

wecSimInputFile.m | wecSimInputFile.m

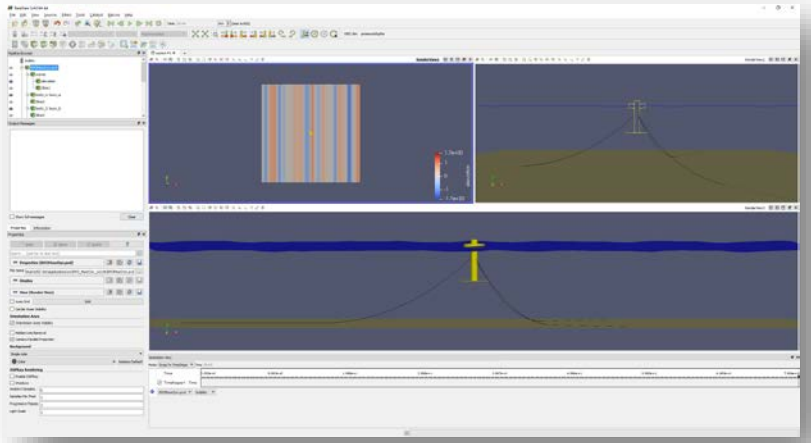
This file can be opened as a Live Script. For more information, see Creating Live Scripts.

```
% Simulation Data
simu = simulationClass();
simu.simMechanicsFile = 'RM3MoorDyn.slx';   % Location of Simulink Model File with MoorDyn
simu.mode='accelerator';
simu.explorer = 'off';
simu.rampT = 0;
simu.endTime=80;
simu.dt = 0.01;
simu.dtOut = 0.1;                           % Specifies output time-step
simu.dtCITime = 0.05;
simu.solver = 'ode45';                      % Runs WEC-Sim with variable time-step
simu.paraview = 1;                          % Saves data to *.vtp for Paraview
simu.domainSize = 300;                      % Changes default domain size

%% Wave Information
% Irregular Waves using PM Spectrum with Convolution Integral Calculation
waves = waveClass('irregular');             % Create the Wave Variable and Specify Type
waves.H = 2;
waves.T = 8;
waves.spectrumType = 'JS';
%waves.viz.numPointsX = 1000;
%waves.viz.numPointsY = 2;

%% Body Data
% Float
body(1) = bodyClass('hydroData/rm3.h5');
body(1).geometryFile = 'geometry/float.stl';
body(1).mass = 'equilibrium';
body(1).momOfInertia = [20907301 21306090.66 37085481.11];

% Spar/Plate
body(2) = bodyClass('hydroData/rm3.h5');
body(2).geometryFile = 'geometry/plate.stl';
body(2).mass = 'equilibrium';
body(2).momOfInertia = [94419614.57 94407091.24 28542224.82];
body(2).initDisp.initLinDisp = [0 0 -0.21];     % Initial Displacement

%% PTO and Constraint Parameters
% Floating (3DOF) Joint
constraint(1) = constraintClass('Constraint1');
constraint(1).loc = [0 0 0];

% Translational PTO
pto(1) = ptoClass('PTO1');
pto(1).k=0;
pto(1).c=1200000;
pto(1).loc = [0 0 0];

%% Mooring
% Moordyn
mooring(1) = mooringClass('mooring');           % Initialize mooringClass
mooring(1).moorDynLines = 3;                    % Specify number of lines
mooring(1).moorDynNodes = [21 21 21];           % Specify number of nodes per line
```

Set data dtOut for ParaView viz

Saves data for ParaView viz

Wave field viz options



RM3_MoorDyn_Viz

Sets ParaView Viz lines and nodes

ParaView for OSWEC with Non-linear hydrodynamics

wecSimInputFile.m | wecSimInputFile.m

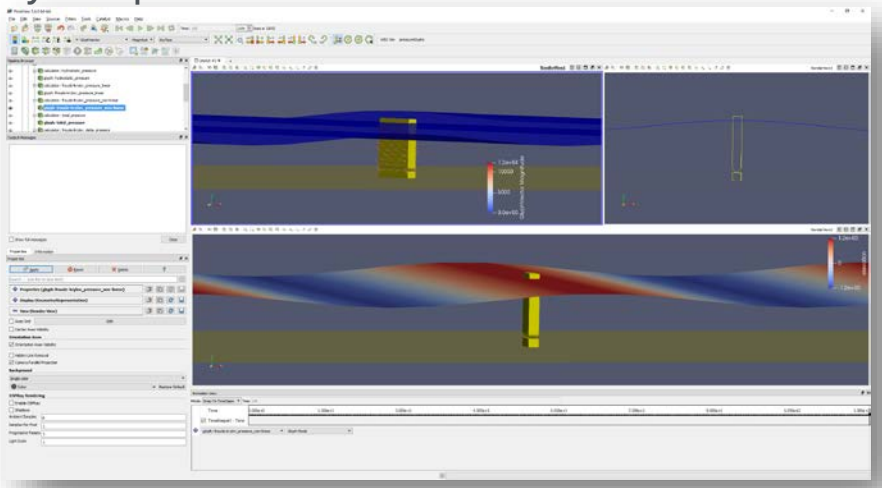
This file can be opened as a Live Script. For more information, see Creating Live Scripts.

```
%% Simulation Data
simu = simulationClass();
simu.simMechanicsFile = 'OSWEC.slx';
simu.mode='accelerator';
simu.explorer = 'off';
simu.startTime = 0;
simu.rampT = 0;
simu.endTime=80;
simu.dt = 0.1;
simu.nlHydro = 2;              % Turns non-linear hydro on
simu.pressureDis = 1;          % Saves pressures data for Paraview
simu.paraview = 1;             % Saves data to *.vtp for Paraview

%% Wave Information
% Regular Waves
waves = waveClass('regular');
waves.H = 2.5;
waves.T = 8;
%waves.viz.numPointsX = 1000;   % wave plane discretization: # X points [default 50]
%waves.viz.numPointsY = 100;    % wave plane discretization: # Y points [default 50]

%% Body Data
% Flap
body(1) = bodyClass('hydroData/oswec.h5');
body(1).geometryFile = 'geometry/flap.stl';
body(1).mass = 127000;
body(1).momOfInertia = [1.85e6 1.85e6 1.85e6];
% body(1).viz.color = [1 1 0];    % [RGB] body color (default [1 1 0])
% body(1).viz.opacity =1;         % body opacity (default 1)

% Base (Non-hydro Body)
body(2) = bodyClass('');                    % Initialize bodyClass without an *.h5 file
body(2).geometryFile = 'geometry/base.stl';    % Geometry File
body(2).nhBody = 1;                         % Turn non-hydro body on
body(2).name = 'base';                      % Specify body name
body(2).mass = 999;                         % Specify Mass
body(2).momOfInertia = [1 1 1];             % Specify MOI
body(2).cg = [0 0 -10.9];                   % Specify Cg
body(2).dispVol = 0;                        % Specify Displaced Volume

%% PTO and Constraint Parameters
% Fixed Constraint
constraint(1)= constraintClass('Constraint1');
constraint(1).loc = [0 0 -10];

% Rotational PTO
pto(1) = ptoClass('PTO1');
pto(1).k = 0;
pto(1).c = 0;
pto(1).loc = [0 0 -8.9];
```

Saves non-linear data for ParaView viz
Saves data for ParaView viz

Wave field viz options

Body viz options



OSWEC_Nonlinear_Viz

New Viz Features?

- Create new macros for ParaView visualization
  - Add glyphs for radiation and diffraction
- Add wave surface elevation to SimScape

Thank you!

U.S. DEPARTMENT OF ENERGY | Energy Efficiency & Renewable Energy

Upcoming scheduled webinars and training courses…

Advanced Feature Webinars *1hr each*

- **April 18:** bemio and mcr, application for power matrix
- **May 24:** nl-hydro, b2b, non-hydro ~~and drag~~
- **June 13:** pto and control, application for desalination
- **July 18:** mooring and visualization
- ***Available Online***: http://wec-sim.github.io/WEC-Sim/webinars.html

Training Courses

- **May 1:** *1hr* WEC-Sim workshop at METS, for new users
- **August 17:** *half-day* WEC-Sim code structure course, for advanced users/developers

U.S. DEPARTMENT OF ENERGY

Sandia National Laboratories

NREL NATIONAL RENEWABLE ENERGY LABORATORY