

WEC-Sim Training Course

for users and developers

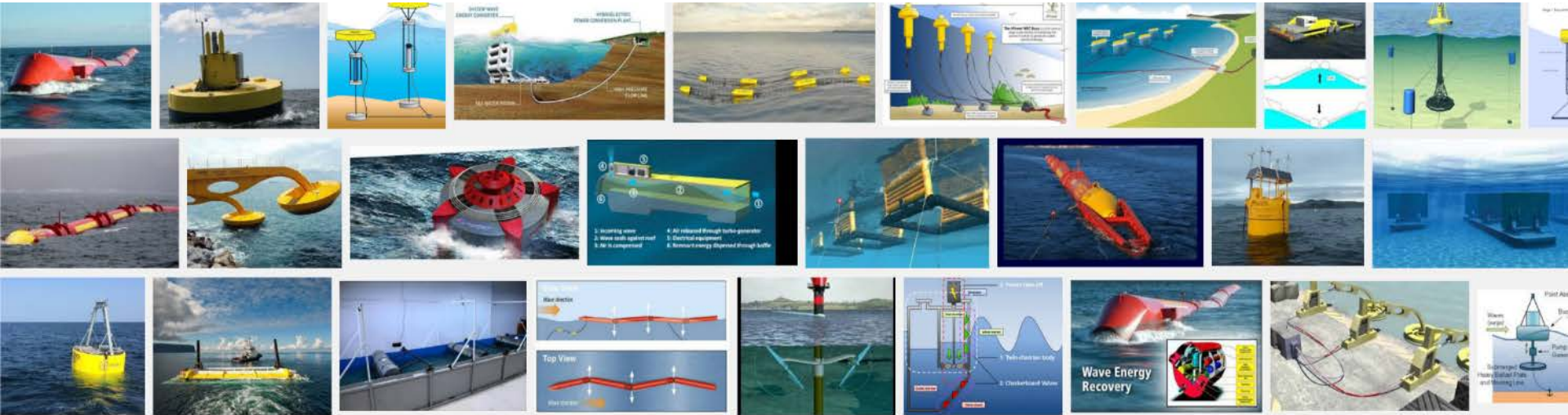
August 17, 2017

Yi-Hsiang Yu (NREL)

Kelley Ruehl (Sandia)

Course Agenda

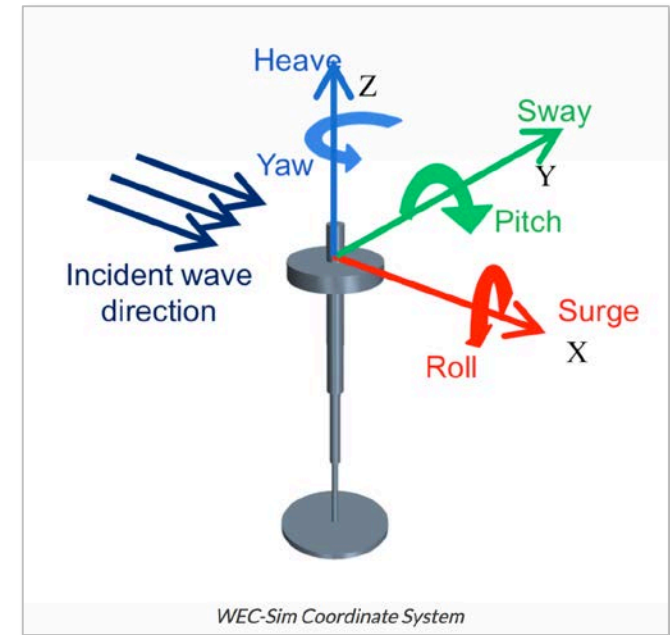
Time	Topic	Description
9:00 am	WEC-Sim Overview ~20min	Overview of course topics and WEC-Sim code
9:30 am	Theory & Workflow ~20min	Cummins' equation and WEC-Sim workflow (BEM->BEMIO->WEC-Sim)
10:00 am	Running WEC-Sim ~30min	Description of what happens when you execute WEC-Sim (<i>wecSim.m</i>)
11:00 am	Code Structure Overview ~1hr total	Overview of WEC-Sim's input file (<i>wecSimInputFile.m</i>), classes (<i>*.m</i>) and library blocks (<i>*.slx</i>)
1:00 pm	Wave Implementation ~30min	Description wave modeling implementation in WEC-Sim, in the classes (<i>*.m</i>) and blocks (<i>*.slx</i>)
1:30 pm	Body Implementation ~30min	Description body implementation in WEC-Sim, in the classes (<i>*.m</i>) and blocks (<i>*.slx</i>)
2:00pm	Q&A ~1hr	Open Q&A for attendees to WEC-Sim Lab team



Theory & Workflow

Yi-Hsiang (NREL)

- X-axis is in the direction of wave propagation if the wave heading angle is equal to zero (following the coordinate system definition in WAMIT).
- The Z-axis is in the vertical upwards direction, and the Y-axis direction is defined by the right-hand rule.
- Surge (x), Sway (y), and Heave (z) correspond to the first, second and third position respectively. Roll (Rx), Pitch (Ry), and Yaw (Rz) correspond to the fourth, fifth, and sixth position respectively.



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

$$m\ddot{x}(t) = F_{hs}(t) + F_{ext}(t) + F_{rad}(t) + F_v(t) + F_{pto}(t) + F_m(t) + F_{nh}(t)$$

The diagram shows the equation of motion with each term in a box. Lines connect the boxes to their respective labels: $F_{hs}(t)$ to Hydrostatic restoring force, $F_{ext}(t)$ to Wave excitation & diffraction force (from BEM simulations), $F_{rad}(t)$ to Radiation force: added mass and radiation damping (from BEM simulations), $F_v(t)$ to Viscous force, $F_{pto}(t)$ to Power take-off force, $F_m(t)$ to Mooring force, and $F_{nh}(t)$ to Nonlinear hydrodynamic force.

Hydrostatic restoring force

Wave excitation & diffraction force (from BEM simulations)

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

Viscous force

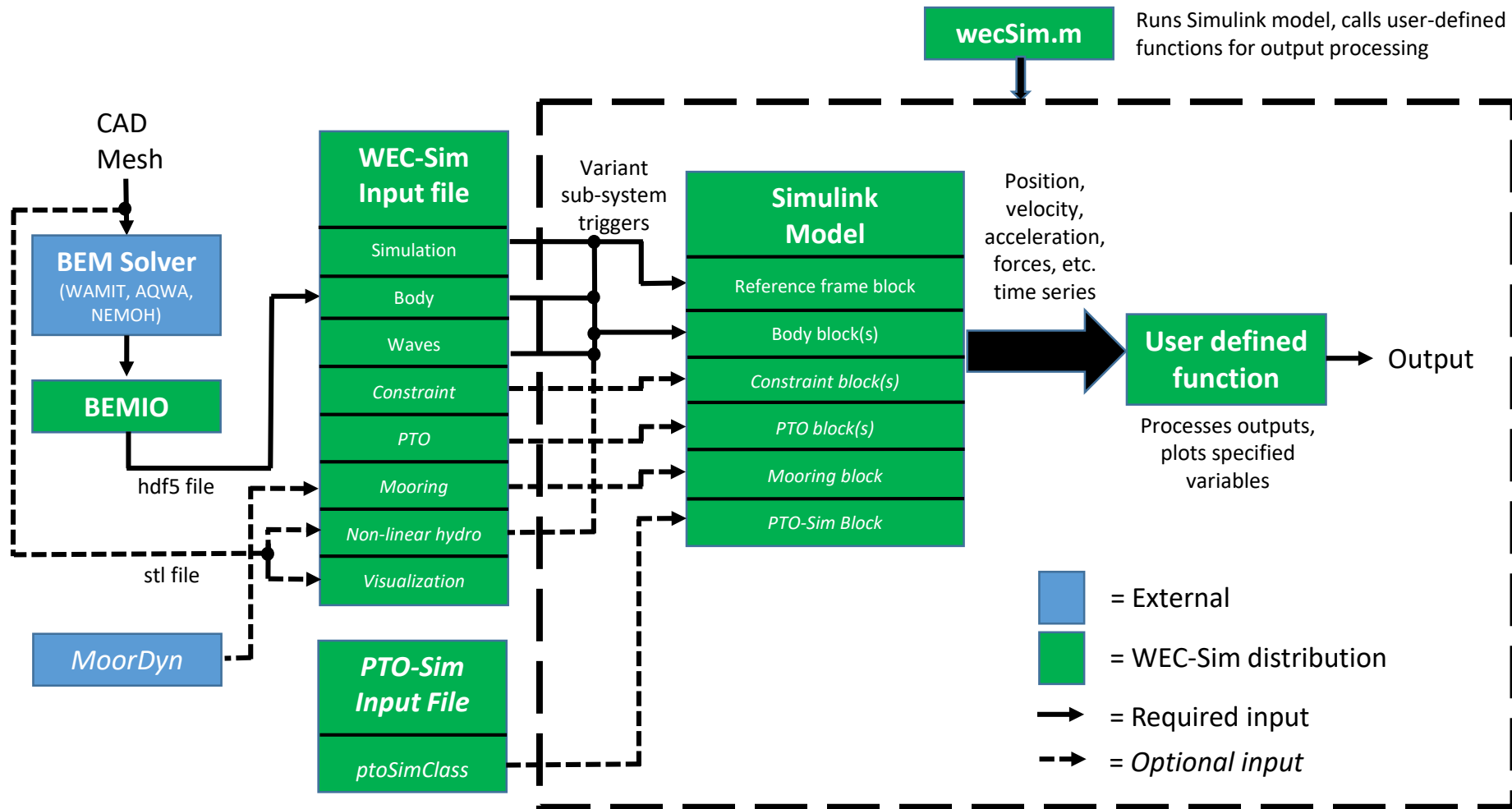
Power take-off force

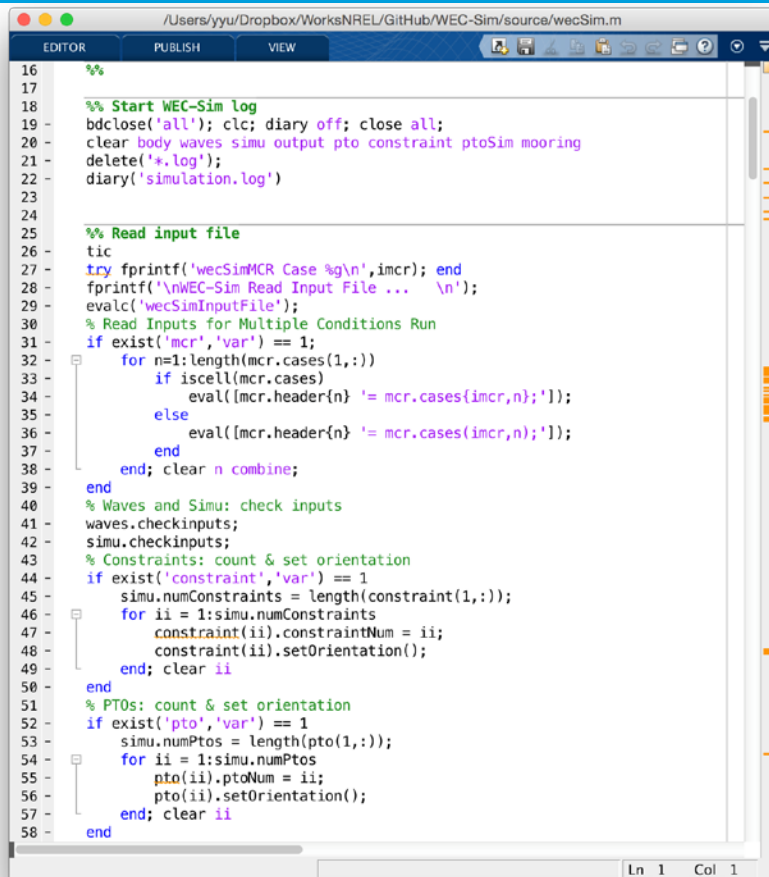
Mooring force

Nonlinear hydrodynamic force

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

Forcing Term	Condition	Theory
Radiation (F_{rad})	Regular Waves	Sinusoidal Steady-State Response $F_{rad} = -A(\omega)\ddot{X} - B(\omega)\dot{X}$
	Irregular Waves	Cummins Equation (Convolution Integral) $F_{rad} = -A_{\infty}\ddot{X} - \int_0^t K(t-\tau)\dot{X}(\tau)d\tau$ State Space Representation $\dot{X}_r(t) = A_r X_r(t) + B_r \dot{\zeta}(t); X_r(0) = 0$ $\int_{-\infty}^t K_r(t-\tau)d\tau \approx C_r X_r(t) + D_r \dot{\zeta}(t)$
Wave Excitation (F_{ext})	Regular Waves	Sinusoidal Steady-State Response $F_{ext} = \Re \left[R_f \frac{H}{2} F_X(\omega) e^{i(\omega t)} \right]$
	Irregular Waves	Wave Spectrum (e.g., JS; BS; PM) $F_{ext} = \Re \left[R_f \int_0^{\infty} F_X(\omega_r) e^{i(\omega_r t + \phi)} \sqrt{2S(\omega_r)} d\omega_r \right]$ Wave Elevation (Convolution Integral) $F_{ext} = \int_{-\infty}^{\infty} \eta(\tau) f_e(t-\tau) d\tau$
PTO (F_{pto})		Linear Spring-Damper $P_{PTO} = C_{PTO} \dot{X}_{rel}^2$
		Hydraulic PTO $P_{PTO} = -F_{PTO} \dot{X}_{rel}$
		Mechanical PTO
Mooring (F_m)		Linear Mooring Matrix (i.e., stiffness, damping and pretension)
		Lumped-Mass Mooring Dynamics Model (MoorDyn)
Additional Added-Mass & Damping (F_v & F_{ME})		Linear & Quadratic Damping Forces $F_v = -C_{ld}\dot{X} - \frac{1}{2}C_d\rho A_D\dot{X} \dot{X} $
		Morison Elements $F_{ME} = \rho\forall\dot{v} + \rho\forall C_a(\dot{v} - \ddot{X}) + \frac{1}{2}C_d\rho A_D(v - \dot{X}) v - \dot{X} $
Nonlinear Hydrodynamic Forces (F_{nh})	Nonlinear Hydrodynamics	The additional term accounts for the difference between the nonlinear and linear hydrodynamic forces (buoyancy and the Froude-Krylov force components).

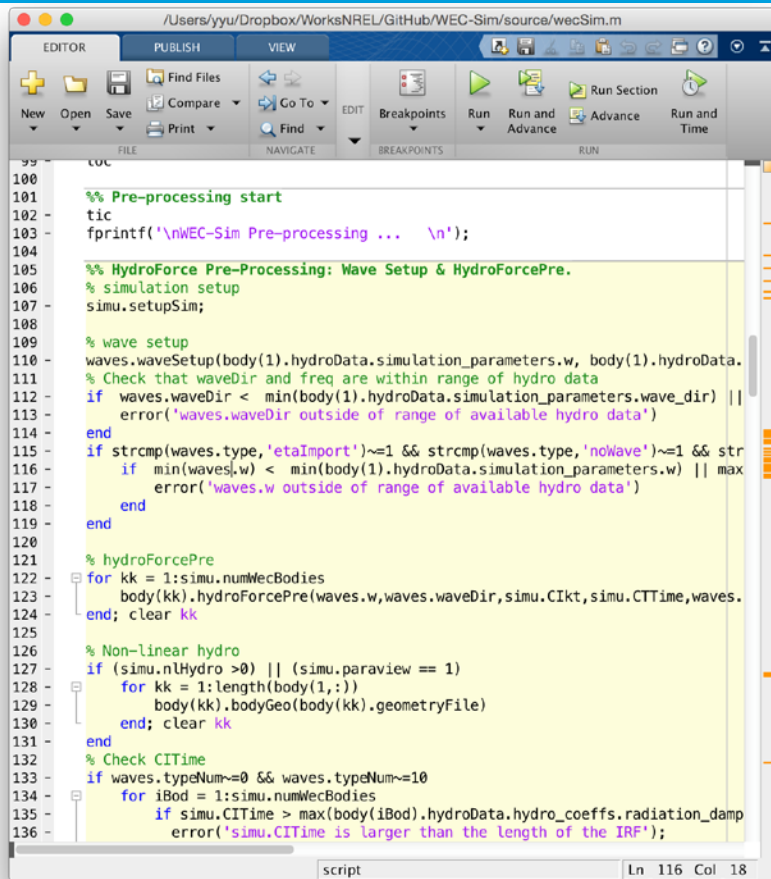




```
16 %  
17  
18 %% Start WEC-Sim log  
19 bdclose('all'); clc; diary off; close all;  
20 clear body waves simu output pto constraint ptoSim mooring  
21 delete('*.log');  
22 diary('simulation.log')  
23  
24  
25 %% Read input file  
26 tic  
27 try fprintf('wecSimMCR Case %g\n',imcr); end  
28 fprintf('\nwecSim Read Input File ... \n');  
29 evalc('wecSimInputFile');  
30 % Read Inputs for Multiple Conditions Run  
31 if exist('mcr','var') == 1;  
32     for n=1:length(mcr.cases(1,:))  
33         if iscell(mcr.cases)  
34             eval([mcr.header{n} ' = mcr.cases{imcr,n};']);  
35         else  
36             eval([mcr.header{n} ' = mcr.cases{imcr,n};']);  
37         end  
38     end; clear n combine;  
39 end  
40 % Waves and Simu: check inputs  
41 waves.checkinputs;  
42 simu.checkinputs;  
43 % Constraints: count & set orientation  
44 if exist('constraint','var') == 1  
45     simu.numConstraints = length(constraint(1,:));  
46     for ii = 1:simu.numConstraints  
47         constraint(ii).constraintNum = ii;  
48         constraint(ii).setOrientation();  
49     end; clear ii  
50 end  
51 % PTOS: count & set orientation  
52 if exist('pto','var') == 1  
53     simu.numPtos = length(pto(1,:));  
54     for ii = 1:simu.numPtos  
55         pto(ii).ptoNum = ii;  
56         pto(ii).setOrientation();  
57     end; clear ii  
58 end
```

Read input file

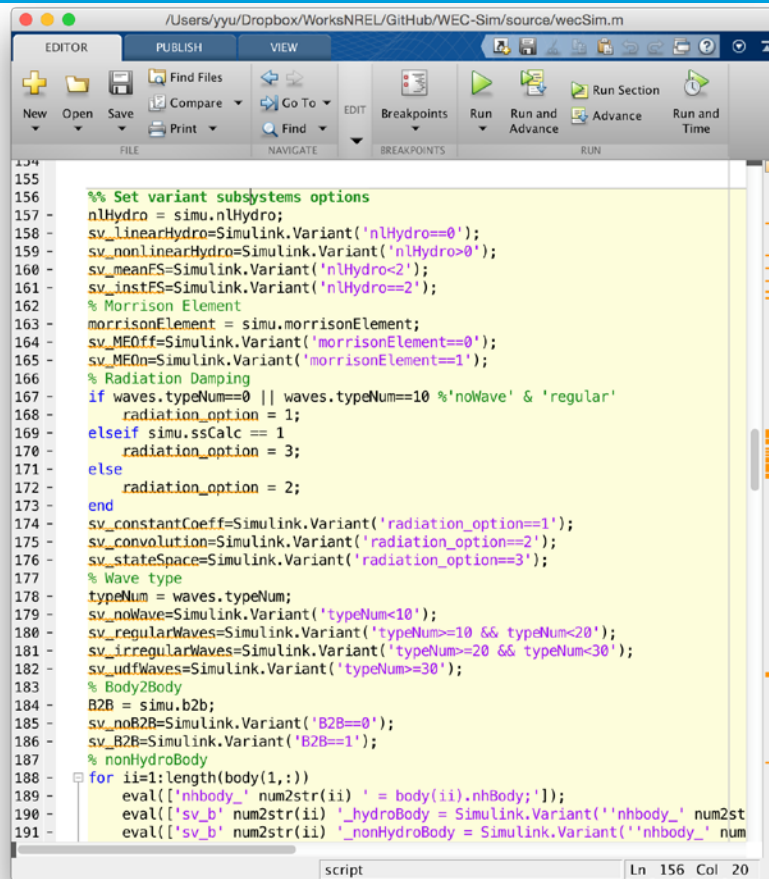
- Read WEC-Sim input file using MATLAB **evalc** to evaluate MATLAB expression with capture
- Check the input parameters and hydrodynamic coefficients for each body



```
100 %
101 %% Pre-processing start
102 tic
103 fprintf('\nwEC-Sim Pre-processing ... \n');
104
105 %% HydroForce Pre-Processing: Wave Setup & HydroForcePre.
106 % simulation setup
107 simu.setupSim;
108
109 % wave setup
110 waves.waveSetup(body(1).hydroData.simulation_parameters.w, body(1).hydroData.
111 % Check that waveDir and freq are within range of hydro data
112 if waves.waveDir < min(body(1).hydroData.simulation_parameters.wave_dir) ||
113     error('waves.waveDir outside of range of available hydro data')
114 end
115 if strcmp(waves.type,'etaImport')==1 && strcmp(waves.type,'noWave')==1 && str
116     if min(waves.w) < min(body(1).hydroData.simulation_parameters.w) || max
117         error('waves.w outside of range of available hydro data')
118     end
119 end
120
121 % hydroForcePre
122 for kk = 1:simu.numWecBodies
123     body(kk).hydroForcePre(waves.w,waves.waveDir,simu.CIkt,simu.CTTime,waves.
124 end; clear kk
125
126 % Non-linear hydro
127 if (simu.nHydro > 0) || (simu.paraview == 1)
128     for kk = 1:length(body(1,:))
129         body(kk).bodyGeo(body(kk).geometryFile)
130     end; clear kk
131 end
132 % Check CITime
133 if waves.typeNum~=0 && waves.typeNum~=10
134     for iBod = 1:simu.numWecBodies
135         if simu.CITime > max(body(iBod).hydroData.hydro_coeffs.radiation_damp
136             error('simu.CITime is larger than the length of the IRF');
```

Pre-processing

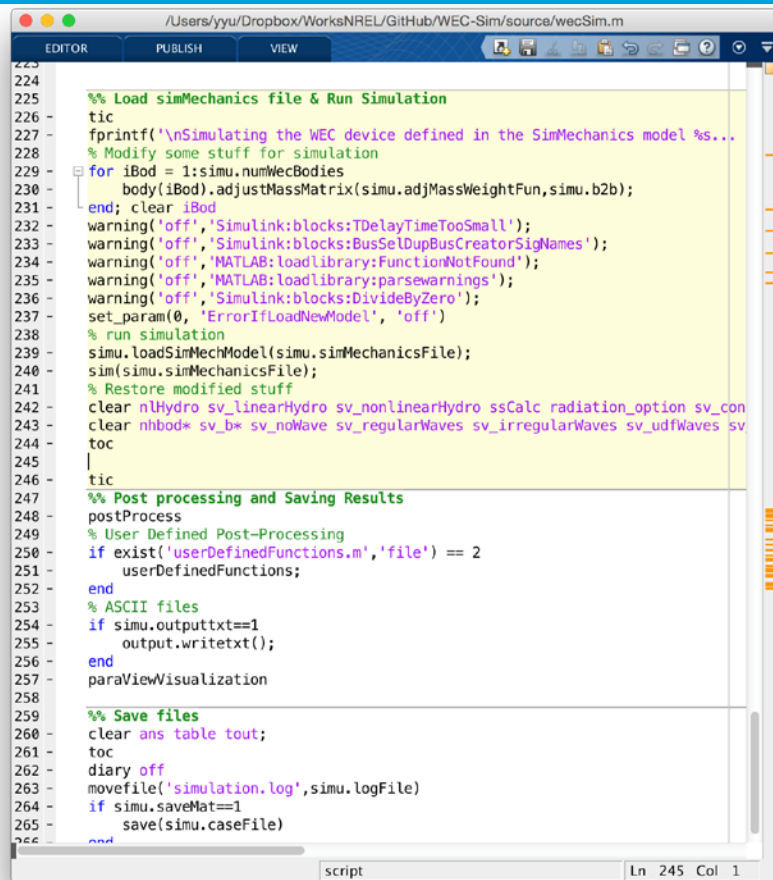
- Setup Simulation parameters
simu.setupSim
- Setup the wave environment and generate wave elevation time history
waves.waveSetup
- Convert non- dimensional hydrodynamic coefficients to dimensional hydrodynamic forces
Body(kk).hydroForcePre



```
154  
155  
156 %% Set variant subsystems options  
157 n1Hydro = simu.n1Hydro;  
158 sv_linearHydro=Simulink.Variant('n1Hydro==0');  
159 sv_nonlinearHydro=Simulink.Variant('n1Hydro>0');  
160 sv_meanFS=Simulink.Variant('n1Hydro<2');  
161 sv_instFS=Simulink.Variant('n1Hydro==2');  
162 % Morrison Element  
163 morrisonElement = simu.morrisonElement;  
164 sv_MEOff=Simulink.Variant('morrisonElement==0');  
165 sv_MEOOn=Simulink.Variant('morrisonElement==1');  
166 % Radiation Damping  
167 if waves.typeNum==0 || waves.typeNum==10 %'noWave' & 'regular'  
168     radiation_option = 1;  
169 elseif simu.ssCalc == 1  
170     radiation_option = 3;  
171 else  
172     radiation_option = 2;  
173 end  
174 sv_constantCoeff=Simulink.Variant('radiation_option==1');  
175 sv_convolution=Simulink.Variant('radiation_option==2');  
176 sv_stateSpace=Simulink.Variant('radiation_option==3');  
177 % Wave type  
178 typeNum = waves.typeNum;  
179 sv_noWave=Simulink.Variant('typeNum<10');  
180 sv_regularWaves=Simulink.Variant('typeNum>=10 && typeNum<20');  
181 sv_irregularWaves=Simulink.Variant('typeNum>=20 && typeNum<30');  
182 sv_udfWaves=Simulink.Variant('typeNum>=30');  
183 % Body2Body  
184 B2B = simu.b2b;  
185 sv_noB2B=Simulink.Variant('B2B==0');  
186 sv_B2B=Simulink.Variant('B2B==1');  
187 % nonHydroBody  
188 for ii=1:length(body(1,:))  
189     eval(['nhbody_' num2str(ii) ' = body(ii).nhBody;']);  
190     eval(['sv_b_' num2str(ii) '_hydroBody = Simulink.Variant(''nhbody_' num2str(ii) '==0');']);  
191     eval(['sv_b_' num2str(ii) '_nonHydroBody = Simulink.Variant(''nhbody_' num2str(ii) '==1');']);  
192 end
```

Variant Subsystems

- WEC-Sim allows user to perform the simulation using different options (e.g., regular/irregular waves, linear/nonlinear model, state-space representation, etc.)
- The options are defined in wecSimInputFile.m and renamed here (Simulink does not like function has a name with '.')



```
224  
225 % Load simMechanics file & Run Simulation  
226 tic  
227 fprintf('\nSimulating the WEC device defined in the SimMechanics model %s...  
228 % Modify some stuff for simulation  
229 for iBod = 1:simu.numWecBodies  
230     body(iBod).adjustMassMatrix(simu.adjMassWeightFun,simu.b2b);  
231 end; clear iBod  
232 warning('off','Simulink:blocks:TDelayTimeTooSmall');  
233 warning('off','Simulink:blocks:BusSelDupBusCreatorSigNames');  
234 warning('off','MATLAB:loadlibrary:FunctionNotFound');  
235 warning('off','MATLAB:loadlibrary:parsewarnings');  
236 warning('off','Simulink:blocks:DivideByZero');  
237 set_param(0, 'ErrorIfLoadNewModel', 'off')  
238 % run simulation  
239 simu.loadSimMechModel(simu.simMechanicsFile);  
240 sim(simu.simMechanicsFile);  
241 % Restore modified stuff  
242 clear nHydro sv_linearHydro sv_nonlinearHydro ssCalc radiation_option sv_con  
243 clear nhbod* sv_b* sv_noWave sv_regularWaves sv_irregularWaves sv_udfWaves sv  
244 toc  
245 |  
246 tic  
247 % Post processing and Saving Results  
248 postProcess  
249 % User Defined Post-Processing  
250 if exist('userDefinedFunctions.m','file') == 2  
251     userDefinedFunctions;  
252 end  
253 % ASCII files  
254 if simu.outputtxt==1  
255     output.writetxt();  
256 end  
257 paraViewVisualization  
258  
259 % Save files  
260 clear ans table tout;  
261 toc  
262 diary off  
263 movefile('simulation.log',simu.logFile)  
264 if simu.saveMat==1  
265     save(simu.caseFile)  
266 end
```

Load Simulink/Simscape Model

- Modify the mass matrix to account for the added-mass term to avoid **algebraic loop** and improve numerical stability
body(iBod).adjustMassMatrix(simu.adjMassWeightFun,simu.b2b)
- Load the **slx** file and perform the simulation
- Post processing and saving results

- Different classes can be specified in any order in the WEC-Sim input file (`wecSimInputFile.m`).
- Multiple Condition Runs function (`wecSimMCR.m`) is just a wrapper that execute `wecSim.m` multiple times.
- Warning message for the Simulink/Simscape Model if MATLAB 2015b is used.

Notes/Warnings/Errors

The image shows the MATLAB R2015b interface. The main window displays a script named `wecSimInputFile.m` in the Editor. The script defines two bodies, `body(1)` and `body(2)`, using the `bodyClass` function. `body(1)` is a float body with a geometry file `'geometry/float.stl'` and a mass of `'equilibrium'`. `body(2)` is a Spar/Plate body with a geometry file `'geometry/plate.stl'` and a mass of `'equilibrium'`. The script also sets the moment of inertia for both bodies.

The Command Window shows the following error messages:

```
Error using wecSim (line 240)
Error due to multiple causes.

Caused by:
Error using wecSim (line 240)
Error evaluating parameter 'constraint' in 'RM3/Floating (3DOF)'
Error using wecSim (line 240)
Undefined function 'constraint' for input arguments of type 'double'.
Error using wecSim (line 240)
Error evaluating parameter 'pto' in 'RM3/Translational PTO'
Error using wecSim (line 240)
Undefined function 'pto' for input arguments of type 'double'.
```

The Workspace window shows the following variables:

Name	Value
ans	1x1 simulationCl...
B2B	0
body	1x2 bodyClass
morrisonelement	0
nhbody_1	0
nhbody_2	0
nlHydro	0
radiation_option	1
simu	1x1 simulationCl...
sv_b1_hydroBody	1x1 Variant
sv_b1_nonHydro...	1x1 Variant
sv_b2_hydroBody	1x1 Variant
sv_b2_nonHydro...	1x1 Variant
sv_B2B	1x1 Variant
sv_constantCoeff	1x1 Variant
sv_convolution	1x1 Variant
sv_instFS	1x1 Variant
sv_irregularWaves	1x1 Variant
sv_linearHydro	1x1 Variant
sv_meanFS	1x1 Variant
sv_MEOFF	1x1 Variant
sv_MEOFF	1x1 Variant
sv_MEOFF	1x1 Variant
sv_noB2B	1x1 Variant
sv_nonlinearHydro	1x1 Variant
sv_noWave	1x1 Variant
sv_regularWaves	1x1 Variant
sv_stateSpace	1x1 Variant
sv_udfWaves	1x1 Variant
typeNum	10
waves	1x1 waveClass

Thank you!

All the webinar materials and recordings are available online:

<http://wec-sim.github.io/WEC-Sim/webinars.html>



The screenshot shows the WEC-Sim webinars page. On the left is a navigation menu with links: Getting Started, Examples, Theory, Code Structure, Advanced Features, Webinars (selected), License, Publications, Release Notes, and Contact Us. The main content area is titled "Webinars" and includes a description: "The WEC-Sim team is hosting a series of advanced features webinars. Dates and topics are listed below. Once completed, the recordings and presentations will be posted to this page." Below this is a table with two columns: Date and Topic.

Date	Topic
April 18, 2017	BEMIO and MCR
May 24, 2017	Nonlinear Hydro, Non-Hydro, and B2B
June 13, 2017	PTO and Control
July 18, 2017	Moorings and Visualization
August 17, 2017	WEC-Sim Training Course

Below the table is the section "WEC-Sim Webinar #1 - BEMIO & MCR" with the text: "The presentation and recordings of WEC-Sim Webinar #1 on BEMIO & MCR hosted on April 18, 2017 are available below. Download the presentation by clicking the image below." At the bottom is a large image of the webinar presentation slides, with the title "WEC-Sim Webinar #1" and the date "April 18, 2017" visible.