



Advanced Features Course



Online Training Materials

PRESENTED BY

Kelley Ruehl, Sandia



Desalination

WEC-Sim Training- Advanced Features



Wave-Powered Desalination System

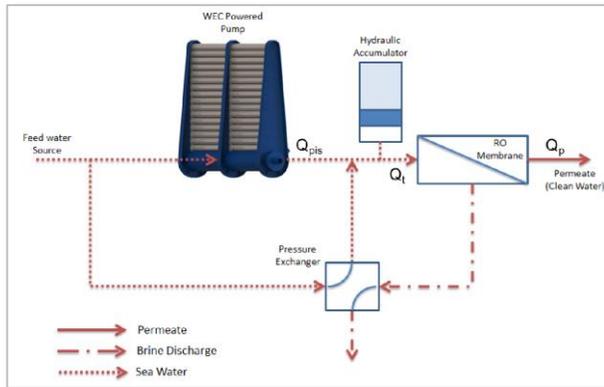
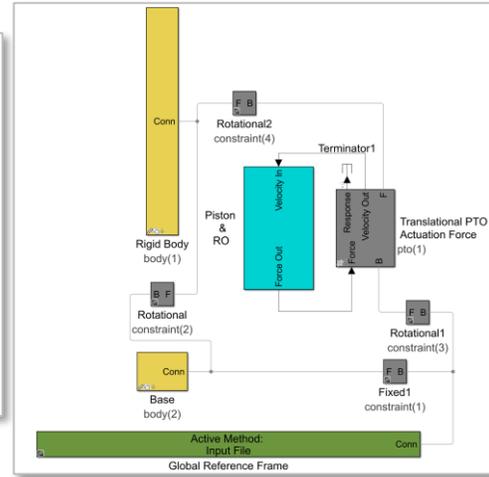
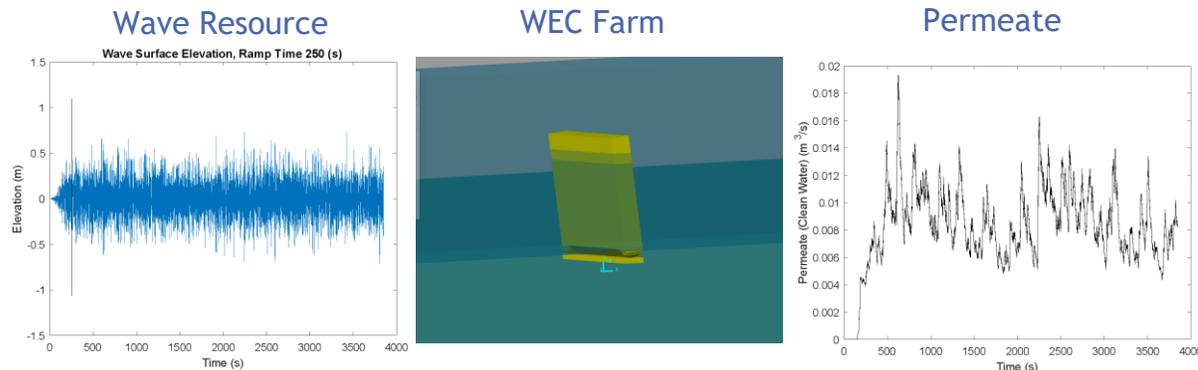


Figure 1 from Yu 2018 refer to [J. Mar. Sci. Eng. 2018, 6, 132](#)



The reverse osmosis (RO) membrane can be modeled similar to power take-off (PTO) systems in WEC-Sim, by providing forces back to the WEC.

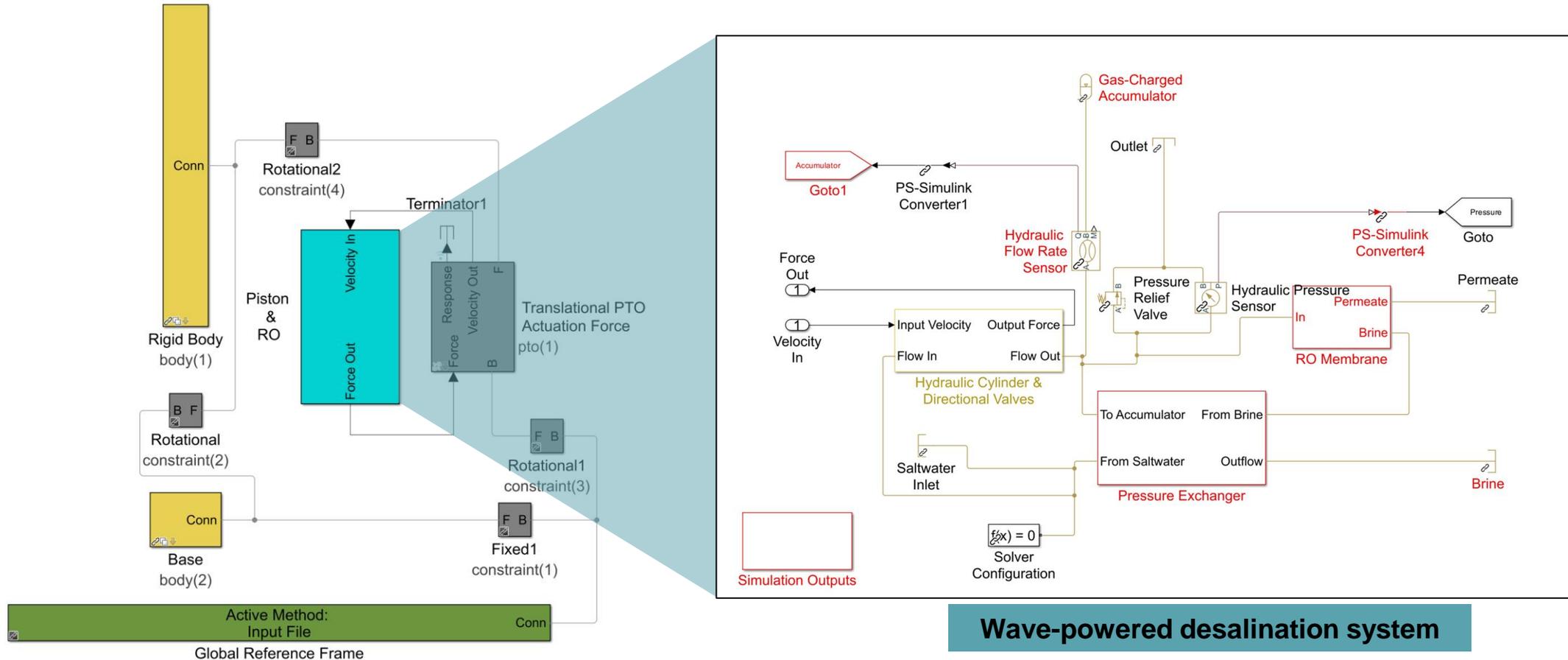
The model takes the wave resource as its input, and provides volumetric flow of permeate (desalinated water) as its output



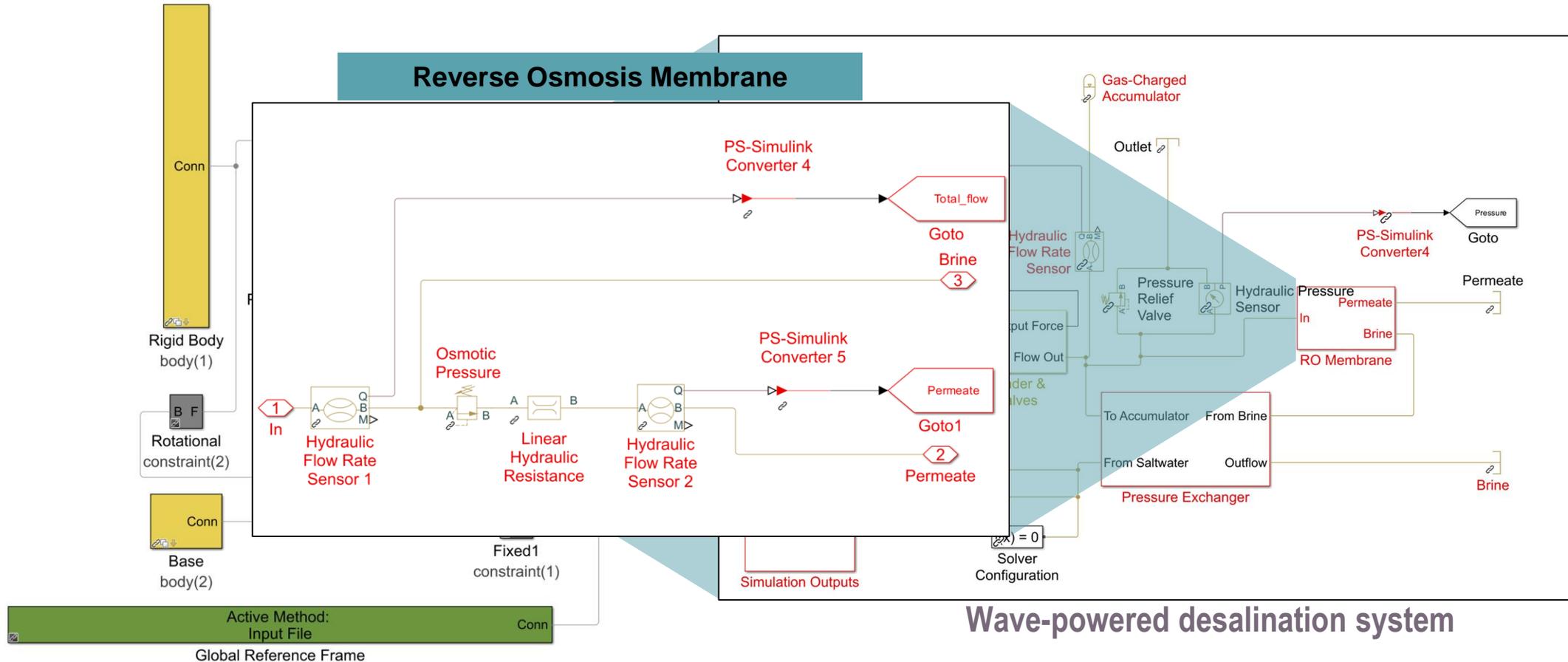
Example using WEC-Sim to model a RO desalination plant available here: https://github.com/WEC-Sim/WEC-Sim_Applications/tree/master/Desalination

NOTE: Requires the MATLAB Toolbox **Simscape Fluids**

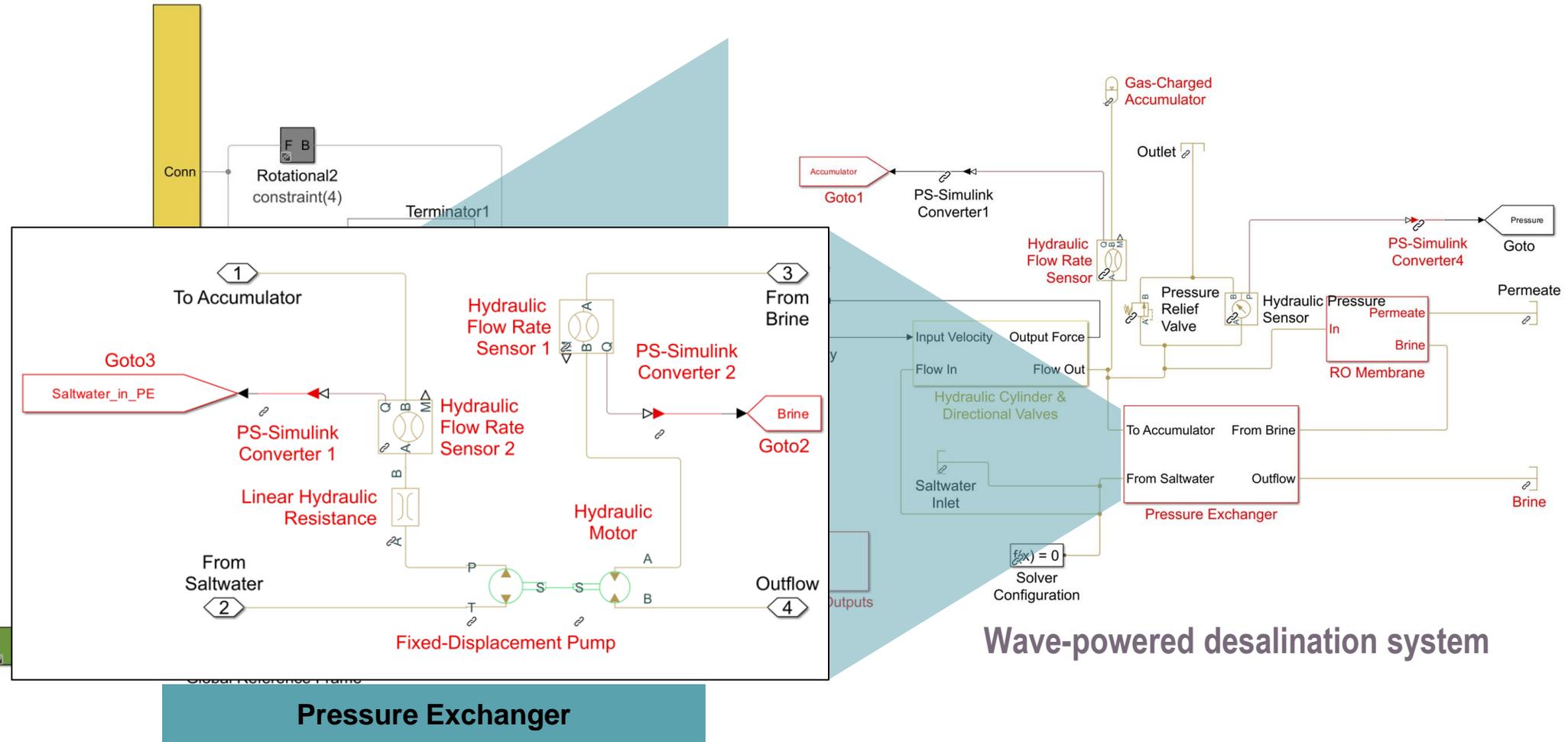
WEC-Sim Model



WEC-Sim Model



WEC-Sim Model

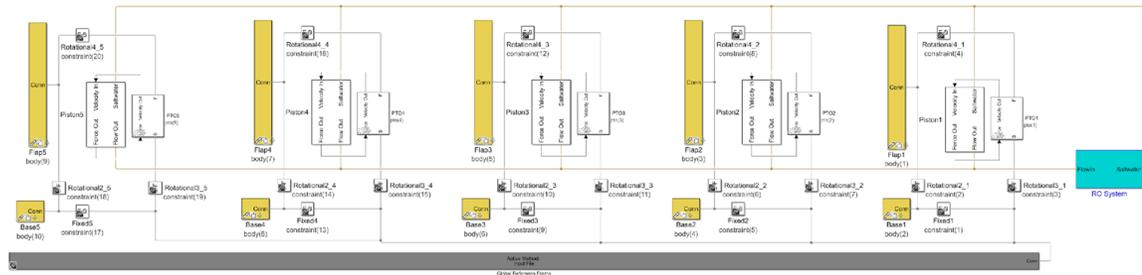


Wave-Powered Desalination Plant

This model can be expanded to include multiple WECs, and for longer durations.

This WEC-Sim run includes five WECs, for a 24-hour simulation.

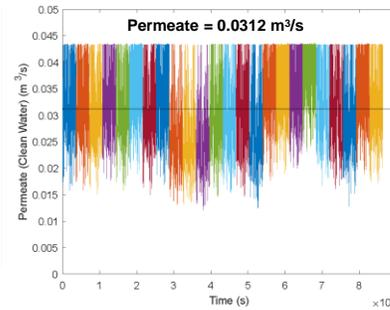
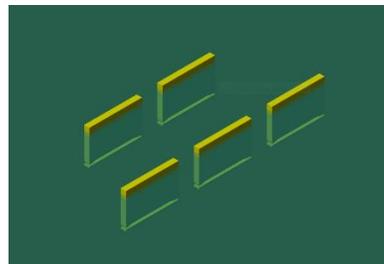
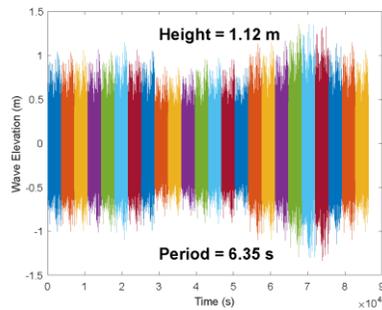
The permeate can then be used as an input to regional water networks models.



Wave Resource

Five WEC Farm

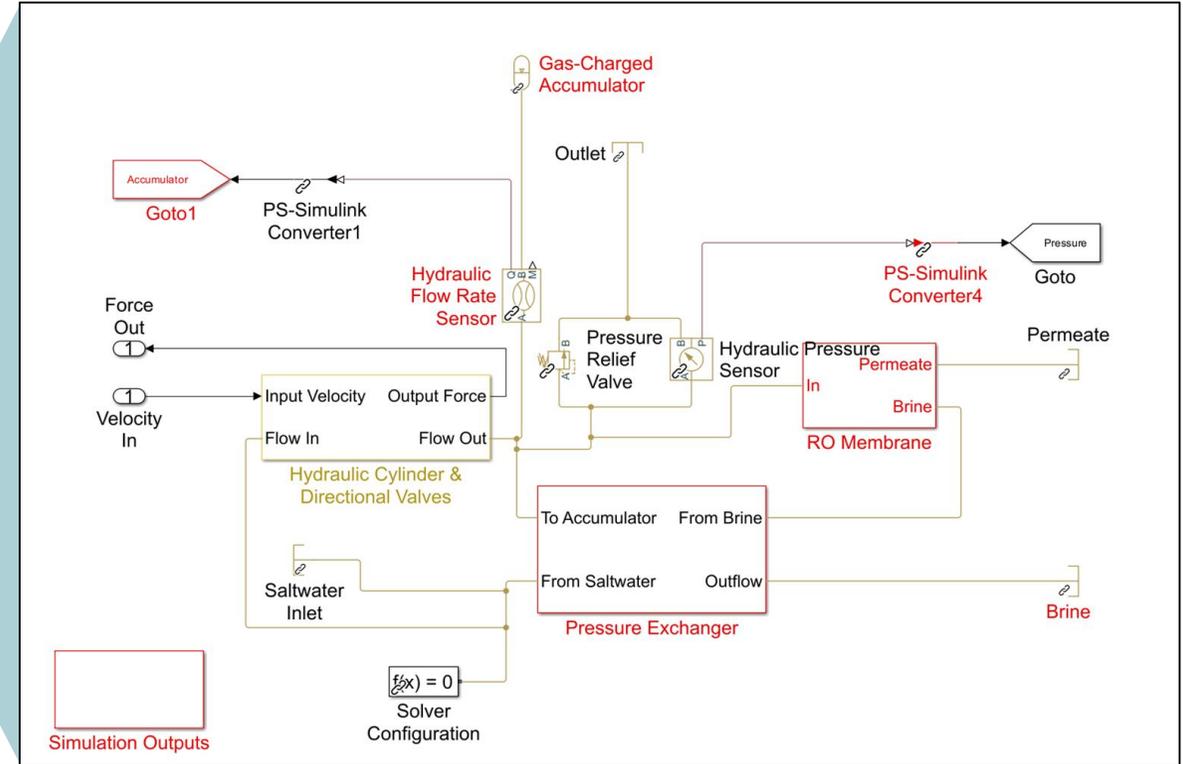
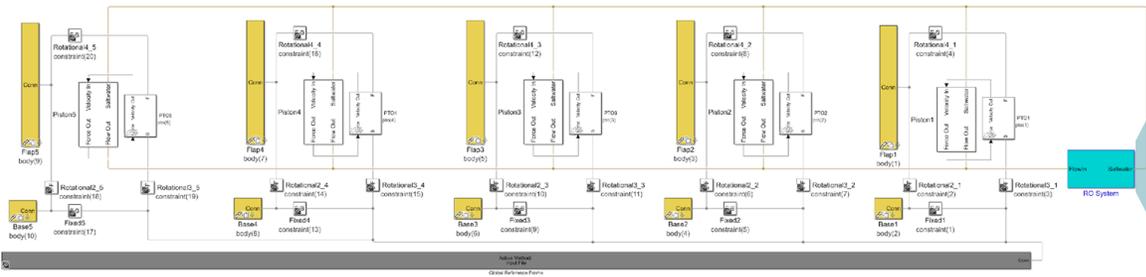
Desalination



WEC-Sim Input File

```
Editor - C:\Users\krmueh\Documents\GitHub\WEC-Sim\Wave-Desal-Seedling\WEC-Sim\OSWEC_RO5_24hr\wecSimInputFile.m
wecSimInputFile.m
1 %% Simulation Data
2 simu = simulationClass();
3 simu.simMechanicsFile = 'OSWEC_RO5_s1x'; % Specify Simulink Model File
4 % Specify Simulation Mode ('normal','accelerato
5 % Turn SimMechanics Explorer (on/off)
6 % Simulation Start Time [s]
7 simu.startTime = 0;
8 % 1 hour simulation, plus the wave ramp time
9 simu.endTime = 3600;
10 % Simulation Start Time [s]
11 % Irregular Waves using PM Spectrum with Convolution Integral Calculation
12 % Initialize Wave Class and Specify Type
13 % Specify Wave Spectrum Type NOTE: this should b
14 % Specify MCR case file
15 %% Wave Information
16 % Irregular Waves using PM Spectrum with Convolution Integral Calculation
17 waves = waveClass('irregular'); % Initialize Wave Class and Specify Type
18 waves.spectrumType = 'PM'; % Specify Wave Spectrum Type NOTE: this should b
19 waves.bem.option = 'EqualEnergy';
20 waves.bem.count = 250;
21 waves.phaseSeed = 1;
22
23 % Puerto Rico wave environment --> see mcrMatFile
24 % waves.height = 0.75; %
25 % waves.period = 5; %
26
27 %% WEC 1 <=0,0>
28
29 % Flap
30 body(1) = bodyClass('.../hydroData/WAMIT/oswec_5/oswec.h5'); % Initialize bodyClass for Flap
31 body(1).geometryFile = './geometry/flap.stl'; % Geometry File
32 body(1).mass = 127000; % User-Defined mass [kg]
33 body(1).inertia = [1.85e6 1.85e6 1.85e6]; % Moment of Inertia [kg-m^2]
34 body(1).morisonElement.option = 1;
35 body(1).morisonElement.cd = ones(5,3);
36 body(1).morisonElement.area = zeros(5,3);
37 body(1).morisonElement.area(:,1) = 18*1.8;
38 body(1).morisonElement.area(:,3) = 18*1.8;
39 body(1).morisonElement.VME = zeros(5,1);
40 body(1).morisonElement.rgME = [0 0 -3; 0 0 -1.2; 0 0 0.6; 0 0 2.4; 0 0 4.2];
41
42 % Base 1
43 body(2) = bodyClass('.../hydroData/WAMIT/oswec_5/oswec.h5'); % Initialize bodyClass for Base
44 body(2).geometryFile = './geometry/base.stl'; % Geometry File
45 body(2).mass = 999; % Creates Fixed Body
46 body(2).inertia = [999 999 999]; % Moment of Inertia [kg-m^2]
47
48 % PTO 1
49 pto(1) = ptoClass('PTO1'); % Initialize ptoClass for PTO1
50 pto(1).stiffness = 0; % PTO Stiffness Coeff [Nm/rad]
51 pto(1).damping = 0; % PTO Damping Coeff [Ns/m]
52 pto(1).location = [2.35106397378+0.9 25 -7.849998936]; % PTO Global Location [m]
53 pto(1).orientation.z = [-4.7021271782/5 0 1.7/5]; % PTO orientation
54
55 % Fixed Constraint 1
56 constraint(1) = constraintClass('Fixed1'); % Initialize ConstraintClass
57 constraint(1).location = [0 0 -10];
58 % Rotational Constraint 2
59 constraint(2) = constraintClass('Rotational2_1'); % Initialize ConstraintClass
60 constraint(2).location = [0 0 -8.9];
61 % Rotational Constraint 3
62 constraint(3) = constraintClass('Rotational3_1'); % Initialize ConstraintClass
63 constraint(3).location = [4.7021271782+0.9 25 -8.7];
64 % Rotational Constraint 4
65 constraint(4) = constraintClass('Rotational4_1'); % Initialize ConstraintClass
66 constraint(4).location = [0+0.9 25 -7];
67
68 %% WEC 2 <=-25,12.5>
69
70 % Flap 2
71 body(3) = bodyClass('.../hydroData/WAMIT/oswec_5/oswec.h5'); % Initialize bodyClass for Flap
72 body(3).geometryFile = './geometry/flap.stl'; % Geometry File
73 body(3).mass = 127000; % User-Defined mass [kg]
74 body(3).inertia = [1.85e6 1.85e6 1.85e6]; % Moment of Inertia [kg-m^2]
75 body(3).morisonElement.option = 1;
76 body(3).morisonElement.cd = ones(5,3);
77 body(3).morisonElement.area = zeros(5,3);
78 body(3).morisonElement.area(:,1) = 18*1.8;
79 body(3).morisonElement.area(:,3) = 18*1.8;
80 body(3).morisonElement.VME = zeros(5,1);
81 body(3).morisonElement.rgME = [0 0 -3; 0 0 -1.2; 0 0 0.6; 0 0 2.4; 0 0 4.2];
82
83 % Base 2
84 body(4) = bodyClass('.../hydroData/WAMIT/oswec_5/oswec.h5'); % Initialize bodyClass for Base
85 body(4).geometryFile = './geometry/base.stl'; % Geometry File
86 body(4).mass = 999; % Creates Fixed Body
87 body(4).inertia = [999 999 999]; % Moment of Inertia [kg-m^2]
88
89 % PTO 2
90 pto(2) = ptoClass('PTO2'); % Initialize ptoClass for PTO1
91 pto(2).stiffness = 0; % PTO Stiffness Coeff [Nm/rad]
92 pto(2).damping = 0; % PTO Damping Coeff [Ns/m]
93 pto(2).location = [2.35106397378+0.9 25 12.5 -7.849998936]; % PTO Global Location [m]
94 pto(2).orientation.z = [-4.7021271782/5 0 1.7/5]; % PTO orientation
95
96 % Fixed Constraint 2
97 constraint(5) = constraintClass('Fixed2'); % Initialize ConstraintClass
98 constraint(5).location = [-25 12.5 -10];
99 % Rotational Constraint 6
100 constraint(6) = constraintClass('Rotational2_2'); % Initialize ConstraintClass
101 constraint(6).location = [-25 12.5 -8.9];
102 % Rotational Constraint 7
103 constraint(7) = constraintClass('Rotational3_2'); % Initialize ConstraintClass
104 constraint(7).location = [4.7021271782+0.9 25 12.5 -8.7];
105 % Rotational Constraint 8
106 constraint(8) = constraintClass('Rotational4_2'); % Initialize ConstraintClass
107 constraint(8).location = [0+0.9 25 12.5 -7];
108
109 %% WEC 3 <=-25,-12.5>
110
111 % Flap 3
112 body(5) = bodyClass('.../hydroData/WAMIT/oswec_5/oswec.h5'); % Initialize bodyClass for Flap
113 body(5).geometryFile = './geometry/flap.stl'; % Geometry File
114 body(5).mass = 127000; % User-Defined mass [kg]
115 body(5).inertia = [1.85e6 1.85e6 1.85e6]; % Moment of Inertia [kg-m^2]
116 body(5).morisonElement.option = 1;
117 body(5).morisonElement.cd = ones(5,3);
118 body(5).morisonElement.area = zeros(5,3);
119 body(5).morisonElement.area(:,1) = 18*1.8;
120 body(5).morisonElement.area(:,3) = 18*1.8;
121 body(5).morisonElement.VME = zeros(5,1);
122 body(5).morisonElement.rgME = [0 0 -3; 0 0 -1.2; 0 0 0.6; 0 0 2.4; 0 0 4.2];
123
124 % Base 3
125 body(6) = bodyClass('.../hydroData/WAMIT/oswec_5/oswec.h5'); % Initialize bodyClass for Base
126 body(6).geometryFile = './geometry/base.stl'; % Geometry File
127 body(6).mass = 999; % Creates Fixed Body
128 body(6).inertia = [999 999 999]; % Moment of Inertia [kg-m^2]
129
130 % PTO 3
131 pto(3) = ptoClass('PTO3'); % Initialize ptoClass for PTO1
132 pto(3).stiffness = 0; % PTO Stiffness Coeff [Nm/rad]
133 pto(3).damping = 0; % PTO Damping Coeff [Ns/m]
134 pto(3).location = [2.35106397378+0.9 25 -0.125 -7.849998936]; % PTO Global Location [m]
135 pto(3).orientation.z = [-4.7021271782/5 0 1.7/5]; % PTO orientation
136
137 % Fixed Constraint 9
138 constraint(9) = constraintClass('Fixed3'); % Initialize ConstraintClass
139 constraint(9).location = [-25 -12.5 -10];
140
141 %% WEC 4
142
143 % Flap 4
144 body(7) = bodyClass('.../hydroData/WAMIT/oswec_5/oswec.h5'); % Initialize bodyClass for Flap
145 body(7).geometryFile = './geometry/flap.stl'; % Geometry File
146 body(7).mass = 127000; % User-Defined mass [kg]
147 body(7).inertia = [1.85e6 1.85e6 1.85e6]; % Moment of Inertia [kg-m^2]
148 body(7).morisonElement.option = 1;
149 body(7).morisonElement.cd = ones(5,3);
150 body(7).morisonElement.area = zeros(5,3);
151 body(7).morisonElement.area(:,1) = 18*1.8;
152 body(7).morisonElement.area(:,3) = 18*1.8;
153 body(7).morisonElement.VME = zeros(5,1);
154 body(7).morisonElement.rgME = [0 0 -3; 0 0 -1.2; 0 0 0.6; 0 0 2.4; 0 0 4.2];
155
156 % Base 4
157 body(8) = bodyClass('.../hydroData/WAMIT/oswec_5/oswec.h5'); % Initialize bodyClass for Base
158 body(8).geometryFile = './geometry/base.stl'; % Geometry File
159 body(8).mass = 999; % Creates Fixed Body
160 body(8).inertia = [999 999 999]; % Moment of Inertia [kg-m^2]
161
162 % PTO 4
163 pto(4) = ptoClass('PTO4'); % Initialize ptoClass for PTO1
164 pto(4).stiffness = 0; % PTO Stiffness Coeff [Nm/rad]
165 pto(4).damping = 0; % PTO Damping Coeff [Ns/m]
166 pto(4).location = [2.35106397378+0.9 25 -7.849998936]; % PTO Global Location [m]
167 pto(4).orientation.z = [-4.7021271782/5 0 1.7/5]; % PTO orientation
168
169 % Fixed Constraint 13
170 constraint(13) = constraintClass('Fixed4'); % Initialize ConstraintClass
171 constraint(13).location = [0 25 -10];
172 % Rotational Constraint 14
173 constraint(14) = constraintClass('Rotational2_4'); % Initialize ConstraintClass
174 constraint(14).location = [0 25 -8.9];
175 % Rotational Constraint 15
176 constraint(15) = constraintClass('Rotational3_4'); % Initialize ConstraintClass
177 constraint(15).location = [4.7021271782+0.9 25 -8.7];
178 % Rotational Constraint 16
179 constraint(16) = constraintClass('Rotational4_4'); % Initialize ConstraintClass
180 constraint(16).location = [0+0.9 25 -7]; % Ke
181
182 %% WEC 5 <=0,-25>
183
184 % Flap 5
185 body(9) = bodyClass('.../hydroData/WAMIT/oswec_5/oswec.h5'); % Initialize bodyClass for Flap
186 body(9).geometryFile = './geometry/flap.stl'; % Geometry File
187 body(9).mass = 127000; % User-Defined mass [kg]
188 body(9).inertia = [1.85e6 1.85e6 1.85e6]; % Moment of Inertia [kg-m^2]
189 body(9).morisonElement.option = 1;
190 body(9).morisonElement.cd = ones(5,3);
191 body(9).morisonElement.area = zeros(5,3);
192 body(9).morisonElement.area(:,1) = 18*1.8;
193 body(9).morisonElement.area(:,3) = 18*1.8;
194 body(9).morisonElement.VME = zeros(5,1);
195 body(9).morisonElement.rgME = [0 0 -3; 0 0 -1.2; 0 0 0.6; 0 0 2.4; 0 0 4.2];
196
197 % Base 5
198 body(10) = bodyClass('.../hydroData/WAMIT/oswec_5/oswec.h5'); % Initialize bodyClass for Base
199 body(10).geometryFile = './geometry/base.stl'; % Geometry File
200 body(10).mass = 999; % Creates Fixed Body
201 body(10).inertia = [999 999 999]; % Moment of Inertia [kg-m^2]
202
203 % PTO 5
204 pto(5) = ptoClass('PTO5'); % Initialize ptoClass for PTO1
205 pto(5).stiffness = 0; % PTO Stiffness Coeff [Nm/rad]
206 pto(5).damping = 0; % PTO Damping Coeff [Ns/m]
207 pto(5).location = [2.35106397378+0.9 25 -7.849998936]; % PTO Global Location [m]
208 pto(5).orientation.z = [-4.7021271782/5 0 1.7/5]; % PTO orientation
209
210 % Fixed Constraint 17
211 constraint(17) = constraintClass('Fixed5'); % Initialize ConstraintClass
212 constraint(17).location = [0 25 -10];
213 % Rotational Constraint 18
214 constraint(18) = constraintClass('Rotational2_5'); % Initialize ConstraintClass
215 constraint(18).location = [0 25 -8.9];
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
Zoom: 100% UTF-8 CRLF script Ln 94 Col 78
```

WEC-Sim Model



Wave-powered desalination system

Thank you



For more information please visit the WEC-Sim website:

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

If you have questions on this presentation please reach out to any of the WEC-Sim Developers on GitHub:

<https://github.com/WEC-Sim/WEC-Sim>

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

This work was authored in part by the National Renewable Energy Laboratory, operated by Alliance for Sustainable Energy, LLC, for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308.

Funding provided by the U.S. Department of Energy Office of Energy Efficiency and Renewable Energy Water Power Technologies Office. The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes.

