New Developments and Capabilities Within WEC-Sim

Preprint

Nathan Tom,[1] Kelley Ruehl,[2] Adam Keester,[2] David Ogden,[3]
Dominic Forbush,[2] Sal Husain,[1] Jorge Leon,[2]
Jeff Grasberger,[2] Matthew Topper,[4] and Yi-Hsiang Yu[5]

1 National Renewable Energy Laboratory
2 Sandia National Laboratories
3 Velocity Global
4 Data Only Greater
5 National Yang Ming Chiao Tung University

Presented at ASME 2023 42nd International Conference on Ocean, Offshore and Arctic Engineering (OMAE 2023)
Melbourne, Australia
June 11–16, 2023

NREL is a national laboratory of the U.S. Department of Energy
Office of Energy Efficiency & Renewable Energy
Operated by the Alliance for Sustainable Energy, LLC

This report is available at no cost from the National Renewable Energy Laboratory (NREL) at www.nrel.gov/publications.

Contract No. DE-AC36-08GO28308

Conference Paper
NREL/CP-5700-85027
June 2023

New Developments and Capabilities Within WEC-Sim

Preprint

Nathan Tom,[1] Kelley Ruehl,[2] Adam Keester,[2] David Ogden,[3] Dominic Forbush,[2] Sal Husain,[1] Jorge Leon,[2] Jeff Grasberger,[2] Matthew Topper,[4] and Yi-Hsiang Yu[5]

1 National Renewable Energy Laboratory
2 Sandia National Laboratories
3 Velocity Global
4 Data Only Greater
5 National Yang Ming Chiao Tung University

Suggested Citation
Tom, Nathan, Kelley Ruehl, Adam Keester, David Ogden, Dominic Forbush, Sal Husain, Jorge Leon, Jeff Grasberger, Matthew Topper, and Yi-Hsiang Yu. 2023. *New Developments and Capabilities Within WEC-Sim: Preprint*. Golden, CO: National Renewable Energy Laboratory. NREL/CP-5700-85027. https://www.nrel.gov/docs/fy23osti/85027.pdf.

Contract No. DE-AC36-08GO28308

Conference Paper
NREL/CP-5700-85027
June 2023

National Renewable Energy Laboratory
15013 Denver West Parkway
Golden, CO 80401
303-275-3000 • www.nrel.gov

NOTICE

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 U.S. Department of Energy Office of Energy Efficiency and Renewable Energy Water Power Technologies Office. The views expressed herein 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.

Cover Photos by Dennis Schroeder: (clockwise, left to right) NREL 51934, NREL 45897, NREL 42160, NREL 45891, NREL 48097, NREL 46526.

NEW DEVELOPMENTS AND CAPABILITIES WITHIN WEC-SIM

Nathan Tom[*]
Salman Husain

National Renewable Energy Laboratory
Colorado, CO 80401
Email: nathan.tom@nrel.gov,
sal.husain@nrel.gov

Kelley Ruehl, Adam Keester,
Jorge Leon, **Dominic Forbush**
Jeff Grasberger

Sandia National Laboratories
Albuquerque, NM, 87123
Email: kmruehl@sandia.gov, akeeste@sandia.gov,
jleonqu@sandia.gov, dforbus@sandia.gov,
jtgrasb@sandia.gov

David Ogden

Velocity Global
London, United Kingdom
Email: david.ogden.nrel@outlook.com

Mathew B. R. Topper

Data Only Greater
Maynooth, Ireland
Email: mathew.topper@dataonlygreater.com

Yi-Hsiang Yu

National Yang Ming Chiao University
Hsinchu, Taiwan
Email: yyu@nycu.edu.tw

ABSTRACT

WEC-Sim is an open-source software for simulating wave energy converters and has been actively developed and applied since its initial release in 2014 to simulate a wide variety of device archetypes. WEC-Sim is developed jointly by the National Renewable Energy Laboratory and Sandia National Laboratories within the MATLAB/SIMULINK environment. A general wave-to-wire model begins with a deployment site resource characterization, which is used to complete the hydrodynamic simulation of wave energy converters (WEC), with the power generation profile imported to a grid simulator to understand the influence on the local electrical network. While modeling the entire wave-to-wire is difficult and encompasses multiple time scales and physics, WEC-Sim is focused on the hydrodynamics simulation to predict, analyze, and optimize WEC dynamics and power performance. WEC-Sim simulations are performed in the time domain based on the radiation and diffraction method using hydrodynamics coefficients derived from boundary element method (BEM)-based frequency-domain potential flow solvers (e.g., WAMIT, NEMOH, Capytaine, or ANSYS-AQWA). With this level of modeling fidelity, WEC-Sim can handle floating body hydrodynamics, mechanical and electrical power generation methods, advanced control implementation, mooring systems, and other unique applications such as desalination. Additional WEC-Sim functionalities include pre-built Simulink blocks and MATLAB scripts that can simulate a wide range of floating systems and the corresponding auxiliary subsystems. The developers of WEC-Sim continue to release new versions of the software, at least annually, with the latest release in September 2022. These releases include bug fixes, updates to software documentation, as well as new features to expand WEC-Sim's capabilities to model a wide range of WEC concepts. This publication will highlight the new features added to WEC-Sim between versions 4.1.0 to 5.0.1, which spans a 2-year period from June 2020 to September 2022. New features described here include topics such as continuous integration checks, revised Morison

---

[*] Address all correspondence to this author.

Element and nonlinear hydro implementations, run directly from Simulink (required for hardware-in-the-loop execution), BEMIO updates to import Capytaine BEM hydrodynamics, addition of cable blocks, and new wave visualization features.

1 INTRODUCTION

Despite research into wave energy dating back to the 1970s, the wave energy converter (WEC) field continues to host a wide range of WEC technologies that remain under development to this day. From a quick literature search an interested reader can find examples of wave energy converter architectures ranging from single- or multi-body point absorbers, attenuators, oscillating water columns (OWCs), oscillating surge wave energy converters (OSWECs), pressure differential units, internal reaction mass, and others that might even blend these traditional categories. Unlike the convergence of the wind energy industry to the three-bladed horizontal wind turbine design, the WEC modeling community faces the challenge of developing numerical models and software to support this wide variety of designs [1]. This only compounds the multidisciplinary problems that WEC developers face, including mooring dynamics, system dynamics, fluid mechanics, power systems, and controls. In an effort to understand the existing software tools available to the marine energy community, in 2012 the U.S. Department of Energy Water Power Technologies Office (US DOE WPTO) funded an analysis by Cardinal Engineering to complete a marine and hydrokinetic software landscape and needs assessment to identify key areas of software development to help accelerate the pace of technology development. The report highlighted a key need for a verified and validated open-source numerical software that could be customized to meet unique end-user needs, which was also supported by a similar study [2] during that time. Such a numerical software could assist with reducing WEC design and analysis uncertainty, enhance understanding of device performance, improve device survivability, and lower the barrier of entry to new market entrants.

In response to these recommendations, the WPTO sponsored the development of such a software, and since 2014 the National Renewable Energy Laboratory (NREL) and Sandia National Laboratories (Sandia) have been maintaining and releasing new versions of WEC-Sim [3, 4]. WEC-Sim is a numerical modeling software that has the ability to model the dynamics of WEC systems that comprise rigid/flexible bodies, power take-off (PTO) systems, and mooring systems. It uses a radiation and diffraction method [5], where the hydrodynamic forces are often obtained from frequency-domain boundary element method models [6–9], to solve the device system dynamics in the time domain [10, 11].

Since its release, WEC-Sim has become a popular tool for WEC numerical modeling across academia and industry for many different device types — and even for some non-wave energy applications [12]. The success and external recognition is highlighted with a perfect score at the 2022 WPTO Peer Review [13] and through a R&D100 Award in the fall of 2021 [14]. WEC-Sim has been used for a variety of purposes and a range of different devices - even including some floating offshore wind turbines (FOWTs) and hybrid FOWT-WEC systems [15]. Many of these studies have taken advantage of WEC-Sim's application library, where "off-the-shelf" WEC models are available to be combined with separate PTO/control models. WEC-Sim's full integration in MATLAB/Simulink can also be leveraged to good effect for hardware-in-the-loop simulations and deployment of controllers on target machines.

Furthermore, experimental validation studies have been conducted with a range of different device types, building confidence in WEC-Sim's versatility and ability to accurately model a range of different device types [16]. This paper will focus discussions on the growth in the WEC-Sim user base, new and recent collaborative activities, and technical and software management updates to improve the readability and usability of the WEC-Sim software. The paper will conclude with comments about the WEC-Sim development team's plan for future work.

2 WEC-SIM USER COMMUNITY

Since its initial release in 2014, WEC-Sim has become a popular numerical modeling and simulation tool for a variety of wave energy archetypes—and for applications beyond wave energy. A literature review of WEC-Sim applications found over 145 publications featuring WEC-Sim between 2013 and 2022, shown in Fig. 1. The publication statistics have an upward trend in number of WEC-Sim publications annually. Early WEC-Sim publications were primarily authored by members of the WEC-Sim software development team from NREL and Sandia, but in recent years the balance has shifted toward a larger number of external publications. Additionally, early WEC-Sim publications were published in conference proceedings, whereas more recent publications have shifted toward publications in peer-reviewed journals. This shift from publications authored by the WEC-Sim development team in conference proceedings to externally authored journal publications is an indicator of the large-scale adoption of the WEC-Sim software by external users.

GitHub repository metrics are another indicator of the growing WEC-Sim user base, and productivity of the WEC-Sim team. Figure 2 shows the number of closed issues and merged pull requests on the WEC-Sim GitHub repository. Closed issues refer to the number of questions submitted by users and resolved by the WEC-Sim team. They can range broadly in scope, from general support questions, to feature requests, to reporting software bugs. Merged pull requests refer to the number of new features merged into the WEC-Sim source code. Pull requests can be made by members of the WEC-Sim team and by external authors. They are the mechanism through which the WEC-Sim

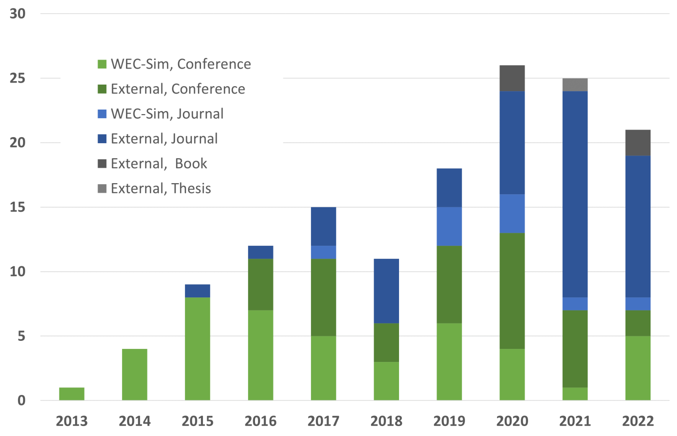


FIGURE 1. WEC-SIM PUBLICATIONS FROM 2013 TO 2022. LIGHT COLORS REFER TO WEC-SIM AUTHORS, DARK COLORS REFER TO EXTERNAL AUTHORS. GREEN REFERS TO CONFERENCE PUBLICATIONS, BLUE TO JOURNAL PUBLICATIONS, GRAY TO THESES, AND BLACK TO BOOKS.

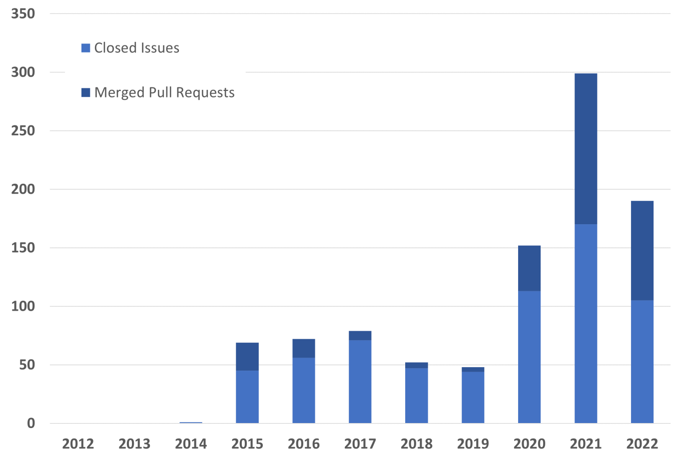


FIGURE 2. WEC-SIM GITHUB REPOSITORY STATISTICS FROM 2013 TO 2022.

user-developer community contributes to the WEC-Sim software. Generally speaking, issues are categorized as WEC-Sim support, and pull requests are categorized as WEC-Sim development. The GitHub repository data show rapid increase in GitHub productivity metrics in the last three years, with triple the number of closed issues and merged pull requests in 2021 compared to 2019. Combined with the publication data, the GitHub data demonstrates the growing WEC-Sim user-base, and establishes WEC-Sim as an industry standard tool for modeling wave energy converters.

2.1 WEC-Sim TEAMER Support

Testing Expertise and Access for Marine Energy Research (TEAMER) is a WPTO sponsored program, directed by the Pacific Ocean Energy Trust [17]. The program aims to "move marine energy technologies closer to market... and advance their progression toward achieving commercial viability, while navigating development and testing barriers along the way." The program accelerates that process through 2–3 annual request for technical support (RFTS) funding calls each year to support developers seeking access to the nation's best facilities and expertise. The TEAMER facility network covers expertise in:

1. Numerical modeling and analysis
2. Laboratory and bench testing
3. Tank, flume, tunnel, and basin testing
4. Open-water testing

TEAMER support is available to both domestic and international developers and researchers, which opens the door for new collaboration with the WEC-Sim team. Since the program launched in 2020, there have been eighteen TEAMER awards for WEC-Sim numerical model development support in collaboration with industry developers and academic researchers. TEAMER awards are typically completed in 9 months and have supported a range of WEC technologies, including point absorbers, oscillating surge wave energy converters, oscillating water columns, and a growing list of unique concepts. Most TEAMER awards consist of collaborating with the WEC-Sim team to run BEM, build a baseline model in WEC-Sim, perform model verification and validation, and accelerate technology transfer to teach awardees how to run and modify the WEC-Sim model on their own upon completion of the award.

3 NEW FEATURE AND CAPABILITIES DEVELOPMENT

WEC-Sim is continually being improved to increase the code's ability to model novel WEC devices by adding new and novel features. These features were identified and implemented in response to participating in code comparison studies such as the International Energy Agency Ocean Energy Systems Task 10 code verification study [18], responding to issue and support request by the WEC-Sim user base, as well as supporting technology developers through TEAMER technical service requests. The following sections will discuss in greater detail the key updates and new features introduced into WEC-Sim since the release of v4.1.

3.1 Running WEC-Sim from Simulink

Run From Simulink is a new feature that allows users to initialize WEC-Sim from the command window and start a simulation directly from Simulink. This feature makes WEC-Sim easily compatible with other Simulink models or hardware-in-the-loop simulations because simulations can be started at the same time as other Simulink models. WEC-Sim simulations can now be initialized from the command window at any point. Then, a complex system involving WEC-Sim, other models, or hardware-in-the-loop can begin directly in Simulink. In this way, WEC-Sim and external models run together, making it easier to couple them together.

Internally, the Run From Simulink functionality differs in how the input file is run. Normally, the “wecSim” command begins by running the “wecSimInputFile” in the current directory, then continuing with the preprocessing steps. The Run From Simulink feature differs by either:

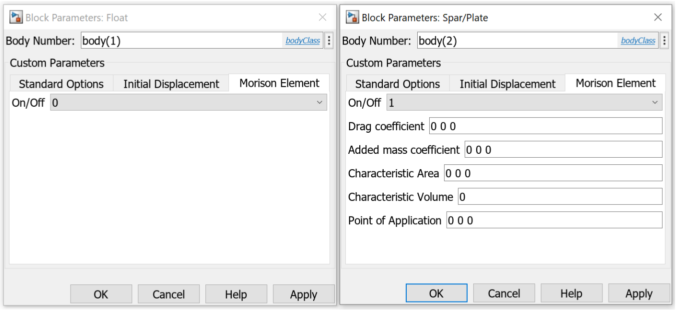
- Running the input file selected in the Global Reference Frame (when the “Input File” option is selected) or
- Writing and then running a new input file when the “Custom Parameters” option is selected in the Global Reference Frame.

The Run From Simulink option allows users to customize input parameters directly in Simulink blocks. Simulation and wave class parameters can be set in the global reference frame, body class parameters can be set in each body block, constraint parameters can be set in each constraint block, etc. The visibility of various parameters depends on the corresponding input flag. For example, Fig. 3 show how when “body(1).morisonElement.option=1,” a user can see and define the rest of the inputs of the Morison element structure. If this option is 0 (off), then the other values are not visible and hidden. This interaction can help new users understand the interdependency of WEC-Sim inputs. Not all input parameters are currently available in the block masks. Additional parameters can be added to this functionality as required.

All of the other source functions, postprocessing, and simulation setup are identical in the Run From Simulink feature as with the standard “wecSim” command. The key differences come from where the user inputs are defined and how the model initialization is split from the model simulation.

3.2 Continuous Integration

WEC-Sim v5.0 development focused on significant refactoring of the software testing methodology. Prior to v5.0, WEC-Sim testing was primarily conducted through acceptance tests based on previously developed WEC-Sim models and validation studies [4]. These models established benchmarks that were used to verify changes in the software in the form of regression testing. This approach is useful to catch source code modifications causing errors with existing models; however, it has a number of drawbacks from a software engineering perspective. Limitations include:



Body Number: body(1)
On/Off 0

Body Number: body(2)
On/Off 1
Drag coefficient 0 0 0
Added mass coefficient 0 0 0
Characteristic Area 0 0 0
Characteristic Volume 0
Point of Application 0 0 0

FIGURE 3. VISUALIZING PARAMETER DEPENDENCY WHEN CUSTOMIZING INPUTS IN SIMULINK.

- Acceptance tests are very resource -intensive, leading to long testing times
- Using existing acceptance tests as a benchmark may lead to inadvertent rejection of changes that increase accuracy
- Acceptance tests may not exercise all of the software’s source code, allowing bugs to be introduced, even if the tests pass.

Continuous integration (CI) is a software engineering process that aims to synchronize the work of software development teams by encouraging the merging of changes at frequent intervals. This is enabled by automatic testing of software updates, leveraging CI frameworks to ensure that revisions to the source code are stable. In addition to challenges faced by developers running the tests locally (e.g., operating systems, software versions), the resource-intensive nature of acceptance tests are also challenging for integration with CI frameworks (e.g., GitHub Actions). For this reason, the WEC-Sim acceptance tests are still used but have been migrated to the WEC-Sim Applications repository, where a suite of WEC-Sim models are hosted using a variety of WEC-Sim features (e.g., nonlinear hydrodynamics, mooring, parallel and batch runs, power take-off and controls, etc.) [19]. Figure 4 outlines the testing workflow and how commits to the WEC-Sim source trigger tests that run and flag a development as passing or failing.

Within the primary WEC-Sim software repository, the development of efficient, high-coverage, unit tests is now the focus. Unit tests are tests for small units of source code—the rationale being that it is easier to design tests for the intended scope of smaller logical units. The number of lines of source code that are exercised in a suite of tests is referred to as the code coverage, typically presented as a percentage of the total lines of code. Code coverage is an important metric, but rigorously testing the scope of the software is equally as important. Thus, used alone,

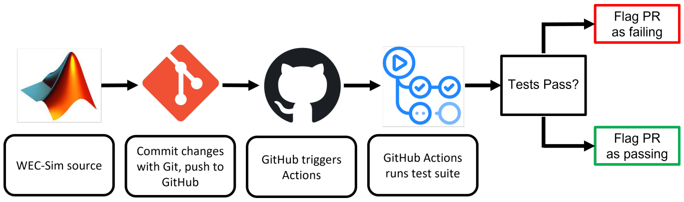


FIGURE 4. WEC-SIM CONTINUOUS INTEGRATION TESTING WORKFLOW.

code coverage is an incomplete metric.

The WEC-Sim test testing suite is implemented using GitHub Actions and runs on both the development (dev) and master branches. Tests are run for each commit to the WEC-Sim GitHub repository as well as for commits to open pull requests, thus ensuring the stability of the software as revisions are made to the source code. Since commits to the WEC-Sim master branch are only made before a release or to resolve bugs, acceptance tests are only run on commits to the WEC-Sim master branch.

3.3 WEC-Sim Refactoring

Without adequate software development guidelines, any software will have a variety of programming styles (e.g., variable names, documentation), and will ultimately result in source code that is very difficult to read by users and developers alike. Even for an established software development team, source code management guidelines should be established and documented. This is especially true for open-source software, like WEC-Sim, where contributions are made by members of the development team, and externally by user-developers. Additionally, since WEC-Sim's initial development in 2014, the project's funding has varied year to year, and thus the level of software support the laboratories can provide has also varied. The WEC-Sim source code was inconsistent and difficult to read; thus, WEC-Sim v5.0 development was also focused on refactoring the WEC-Sim source code. The goal for refactoring was threefold: 1) make the source code more approachable for user-developers, 2) improve the development team's workflow, and 3) clarify the source code development guidelines. An example of one such refactoring improvement that addressed all three is breaking up the WEC-Sim library into sub-libraries. This revision did not change the functionality of the WEC-Sim software, but it clearly distinguished each library from one another (e.g., frame, body, PTO), and it made revisions to the library easier to track (addressing a unique challenge with version control of binary files).

3.4 WEC-Sim PCT Feature

Since the release of WEC-Sim v4.2, WEC-Sim has had the ability to perform parallel execution of batch runs using the MATLAB parallel computing toolbox (PCT). In previous versions, the multiple condition runs (MCR) option allowed a user to setup a series of batch runs to run automatically one after another unsupervised but did not utilize parallel computing to speed up simulations. WEC-Sim's "wecSimPCT" function executes the MATLAB (PCT) which allows for parallelization of the existing MCR option. The implementation of the PCT feature does require adding a MATLAB Toolbox dependency, but the WEC-Sim development team felt this option was a valuable addition for researchers hoping to better exploit their high-performance computing resources.

3.5 Improvements to PTO-Sim

The proper design and study of PTO systems is critical to improve the efficiency and power generation capabilities in WECs. Ignoring the PTO influence on the WEC dynamics could lead to misleading conclusions. WEC-Sim allows users to couple the PTO models into their simulation. There are two options to model PTO systems using WEC-Sim: The first option is to use a simplified parametric approach in which the PTO is modeled as a linear spring-damper system. In this case, the PTO force is calculated as a function of the spring-damper coefficients and the displacement, x, and velocity, $\dot{x}$, of the WEC, as shown in Eqn. (1).

$$F_{PTO} = k_{PTO}x + c_{PTO}\dot{x} \tag{1}$$

where k_{PTO} and c_{PTO} are spring and damping coefficients of the PTO, respectively.

The second approach available in WEC-Sim is to model in detail the dynamic interaction of the different parts of the PTO. There are different commercial software options that allow users to model complex PTO systems such as Simscape with all its packages like Simscape Fluids, Simscape Electrical, Simscape Driveline, etc. The PTO-Sim library was created within WEC-Sim to allow users to model complex PTO systems. PTO-Sim has a variety of blocks that can be used to simulate mechanical or hydraulic drivetrains commonly used in WEC applications. The current version of PTO-Sim is divided into three sub-libraries: Electric, Hydraulic, and Motion Conversion. The sub-libraries have blocks to model electric generators and hydraulic devices such as cylinders, rectifying valves, accumulators, and hydraulic motors, among others. Some of the blocks currently available in PTO-Sim are shown in Fig. 5, Fig. 6, and Fig. 7.

PTO-Sim can easily be coupled with the other WEC-Sim blocks, which allows the user to develop complete wave-to-wire models of the WEC systems. One of the main features of PTO-Sim in the v5.0 release is the versatility of the blocks, which can be used to simulate complex hydraulic or mechanical systems. For example, the hydraulic motor block allows the user to simulate a variable displacement hydraulic motor, including two dif-

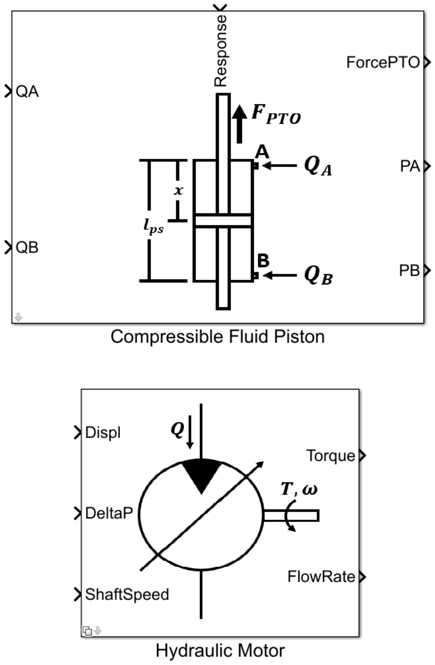


FIGURE 5. PTO-SIM HYDRAULIC BLOCKS

ferent options to consider the motor efficiency in the simulation. The first option is with an analytical model of efficiency, based on characteristics of the fluid and the hydraulic motor. The second option is with a tabulated efficiency that will interpolate the efficiency of the motor based on the current values of pressure and flow.

Additionally, users can implement control algorithms using the PTO-Sim blocks inputs as control signals. One of the inputs of the "Electric Generator Equivalent Circuit" block is the load resistance. The users could use this block input to adjust the load of the system depending on the specific requirements of their application. Different control algorithms could be used to adjust this value to achieve the desired behavior. Examples of using PTO-Sim can be found in the WEC-Sim Applications repository on GitHub [7].

3.6 Cable Implementation

For many WEC archetypes, particularly those with a large spatial separation between dynamic bodies, bodies are coupled together with cables or tethers. This is a distinct consideration from a cable mooring, for which the seabed-affixed cable end is static. The challenge of modeling a cable is that forces are only transmitted between coupled bodies while the cable is in tension: if it is not, then the motion between the bodies is not coupled by the cable. While Simscape Multibody includes cable/tether elements, these blocks are useful only if the cable is perpetually in tension. In fact, the simulation will report an error if this is ever not the case. This is problematic for WEC devices because cable slack events do occur and are of significance for many design problems.

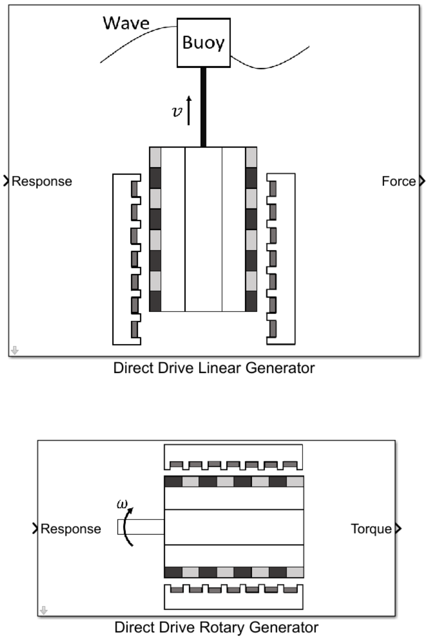


FIGURE 6. PTO-SIM ELECTRIC BLOCKS

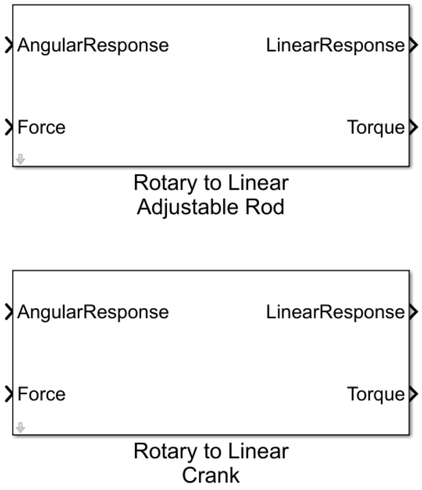


FIGURE 7. PTO-SIM MOTION CONVERSION BLOCKS

To address this modeling need, a simple cable implementation was created within WEC-Sim. This is not intended to capture the motion/dynamics of the cable itself but only the resultant forces exerted on the coupled bodies. Cable force acts only in tension in the direction of the cable and is calculated as:

$$F_{cable} = \begin{cases} [0,0,0] & \text{if } L \leq L_0 \\ -K(X_F - X_B) - C(\dot{X_F - X_B}) & \text{if } L > L_0 \end{cases} \tag{2}$$

where L is the instantaneous cable length based on the separation of the coupled constraints, and

$$L = |X_F - X_B| \tag{3}$$

where X_F and X_B are the [x,y,z] locations of the follower and base constraints, respectively, to which the cable is coupled. L_0 is the user-defined unstretched cable length (m), K is the cable stiffness (N/m), and C is the cable damping coefficient (N-s/m), which apply force through the cable if and only if the cable is stretched beyond its unstretched length (Eqn. (2)). By default, the cable itself is defined as neutrally buoyant and inviscid, such that the motions and mass of the cable itself do not affect coupled body motion. The cable implementation includes two drag bodies, located by default at either cable end, that allow for mass and fluid drag forces to be applied to the cable itself based on user-specific drag coefficients, characteristic areas, and cable mass. The resulting model is effectively a low-order (two-element) lumped capacitance model of the cable dynamics. In theory, this framework can be extended to more elements and resolve actual cable motion, but as it stands, it is only suitable to capture the resultant force on each coupled body.

To support realistic cable coupling, a spherical constraint and spherical PTO block were also introduced. These blocks fix translational degrees of freedom and allow rotation in all three rotational degrees of freedom. In the constraint block, these rotational degrees of freedom move without resistance. The spherical PTO block allows a stiffness and a damping to be applied to these motions.

3.7 BEMIO improvements and Support for Capytaine

In addition to already supporting the BEM software such as WAMIT, Nemoh, and AQWA, BEMIO was extended recently to also support Capytaine (itself a major rewrite of Nemoh with many modifications, including significant changes to the format of the output data, now saved as a binary NetCDF file). The hydrostatics-related text files (i.e., KH.dat and Hydrostatics.dat) can be provided separately in the same way as Nemoh. At present, Meshmagick can be used to compute these properties. In future versions it may be possible to have all required data computed solely by Capytaine and saved to a single NetCDF file to be read by BEMIO.

One of the purposes of BEMIO is to compute the impulse response functions (IRFs) from the hydrodynamics coefficients. These functions were recently sped up on the order of 10x by use of vectorization. While in previous versions of BEMIO, the added mass at infinite frequency term was taken as the value computed by the BEM software for the highest frequency defined in the user's input file. A recent improvement has been to apply a more robust approach by calculating the infinite frequency added mass at each input frequency, and taking the mean average of these values.

To implement this approach, WEC-Sim continues to use the inverse Fourier transform of the radiation wave damping coefficients to calculate the impulse response function [20] (see Eqn. (4)), as for most cases these coefficients decay to zero well before reaching infinite frequency. With better confidence in the calculation of the impulse response function, BEMIO can then perform the Fourier transform on the impulse response function to estimate the infinite frequency added mass as shown by Eqn. (5). Theoretically for any chosen wave angular frequency, Eqn. (5) should provide the same value for infinite frequency added mass; however, depending on which frequencies are output by the BEM software and the length of time used in the integration the calculation can result in variations in the output. After performing statistics on the variation in the calculation of the infinite frequency added mass, the development team decided taking the average across all frequencies would provide the most robust solution.

$$K_{r,ii}(t) = \frac{2}{\pi}\int_0^\infty \lambda_{ii}(\omega)\cos\sigma t d\sigma \tag{4}$$

$$\mu_{ii}(\infty) = \mu_{ii}(\sigma) + \frac{1}{\sigma}\int_0^\infty K_{r,ii}(t)\sin\sigma t dt \tag{5}$$

For Eqns. (4) and (5) the variable definitions are as follows: $K_{r,ii}$ is the radiation impulse response function for the ith direction, λ_{ii} is the radiation wave damping, σ is the wave angular frequency, t is time, μ_{ii} is the radiation added mass. Note that this option is only used if the infinite frequency added mass is missing from the hydrodynamic data or the BEM software cannot calculate the infinite frequency added mass directly. For example, WAMIT has the capability to solve for the infinite frequency added mass directly, and those values were used to verify this approach.

3.8 Spatial Phase Lag for Excitation Force

By default, WEC-Sim precalculates excitation force based on the initial X-Y position of the simulated WEC(s) from the waveClass definition to define local wave height and the excitation coefficients from BEM to calculate force from these heights. This excitation force can be inaccurate if a body undergoes large X-Y displacements over the course of the simulation because the precalculation presumed a consistent body location.

To better handle simulation of these cases, a simulation option to bodyClass was added that recalculates the phase of the exciting wave as a function of current displacement. As an example, consider the wave height calculation for an irregular wave:

$$\eta(x,y,t) = \sum_{i,j} \frac{H_{i,j}}{2} \cos\left(\omega_{i,j}t - k_{i,j}(x\cos\theta_{i,j} + y\cos\theta_{i,j}) + \phi_{i,j}\right) \tag{6}$$

where H is the wave height, ω is the radian frequency, t is time in seconds, k is wave number, θ is the wave direction, ϕ is the phase, and x and y define the location on the nominal free-surface plane of interest. The indices i and j denote the frequency and direction of the wave component, respectively. The latter two terms of the cosine argument can collectively be considered a phase and by default are calculated in advance for all wave components; in fact, the default body location is initially (0,0), implying that only ϕ determines wave component phasing, as the coefficient of k is thus zero. If the "largeXYDisplacement" option is selected for the body, however, this coefficient of k is included with x and y updated at every time step. An identical argument applies for regular plane waves, although in this case the indices i, j can be dropped, as there is only one wave component.

3.9 Improved Wave Visualization Features

In previous versions of WEC-Sim, wave visualization could be completed by exporting the results to another software known as Paraview. Although Paraview can be a useful visualization and analysis tool, it is an entirely separate software from WEC-Sim that must be downloaded and installed to visualize the wave-body interaction. Moreover, learning to use Paraview and the exportation of data are each further steps that are not simple and have led to confusions from WEC-Sim users. In order to avoid the dependency on Paraview for wave visualization, two methods have been developed for visualization, both of which can be directly called by WEC-Sim. The first method utilizes MATLAB's 3D plotting feature to plot the wave and body position over time, and the second method adds wave elevation markers to the mechanics explorer feature.

3.9.1 Three dimensional plots in MATLAB The saveViz() function is applied as a postprocessing step of WEC-Sim and can be added to the userDefinedFunctions file to create a 3D plot of the wave conditions and body response. MATLAB's 3D plotting feature is ideal for visualizing the wave-body interactions modeled by WEC-Sim due to its ability to plot multiple surfaces and bodies and relatively simple saving/exporting function. As an example of the wave visualization, a plot of the RM3 and wave conditions are shown in Fig. 8. The wave elevation is indicated by the location of the surface as well as a color spectrum, and the response of the body is plotted in terms of both translation and rotation. The saveViz() function allows inputs for the axis limits, time steps per animation frame, the visualization start and end time, and a setting to specify whether to save it as a .gif or .avi file.

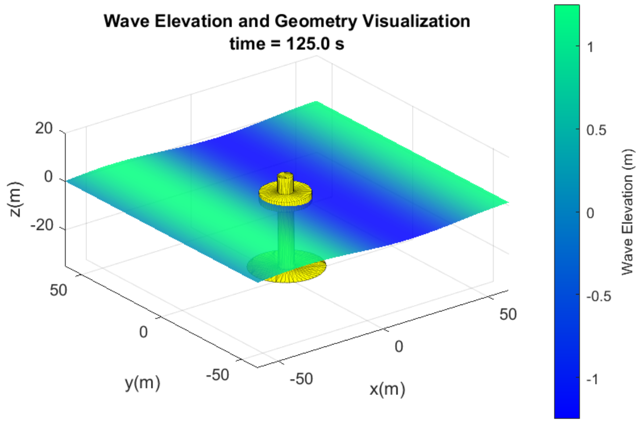


FIGURE 8. WAVE-BODY INTERACTION VISUALIZATION FOR RM3 EXAMPLE.

3.9.2 Wave visualization using markers New visualization capabilities were introduced that can represent wave elevations $\eta(x,y,t)$ at user-defined locations using animated markers within Simscape Mechanics Explorer. In terms of user inputs, if the user adds information about the marker locations, then in the Simulink model, the visualization subsystem blocks are programmatically added in the background. The user defines a set of locations, marker size, and selects a marker type—a spherical, cuboid, or a local coordinate-frame tripod. These visualization markers represent the time histories of the undisturbed wave elevations and are animated in the Simscape Mechanics Explorer alongside the simulation of the time-domain dynamics of the WEC. Note, the visualization markers only use the incoming wave spectra to generate wave elevation at a particular location and are not informed by the wave interactions with the floating bodies being simulated, as though representing the wave spectra with no floating bodies present.

Wave elevations at a given location in the $X-Y$ plane are calculated using Eqn. (6), where the necessary wave parameters are extracted from the *properties* of the *wave* object. The wave object can be described as a container that stores the wave parameters (*properties*) and the necessary functions (*methods*). New *properties* and *methods* were introduced in the *wave* object, to generate the time histories of the wave elevations $\eta(x,y,t)$ at each user-defined location.

The wave visualization markers are generated in two stages: (1) generation of the $\eta(x,y,t)$ time histories at the user-defined locations, and (2) generating corresponding blocks in the Simulink model that can then be visualized in the Simscape Mechanics Explorer. The first stage is accomplished using the new *properties* and *methods*. The second stage involves

- Passing the locations of the user-defined $\eta(x,y,t)$ time histories to Simulink
- Programmatically adding marker subsystems in Simulink, based on the user-defined marker locations and characteristics.

Additionally, it was important to have mechanisms to programmatically remove visualization blocks, especially if a previous run saved the Simulink model with visualization blocks that may not be needed for the current or a subsequent run. This requirement was fulfilled by checking the number of user-defined locations for every new run and programmatically removing the extra visualization blocks. The programmatic removal of extra blocks was also important for backwards compatibility with previous versions of WEC-Sim. If the user did not define any marker locations, and if the Simulink model does contain marker subsystems from a previous run, they would be programmatically removed. This removal will require no additional input from the user and will be performed in the background.

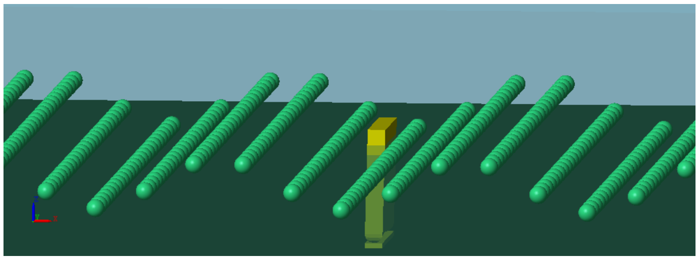


FIGURE 9. THE OSWEC EXAMPLE FROM THE WEC-SIM REPOSITORY WITH SPHERICAL MARKERS REPRESENTING THE INCOMING WAVE.

Figure 9 shows an example of the visualization blocks representing an incoming wave alongside an OSWEC device. One of the advantages of this new feature is that it works within the MATLAB-Simulink ecosystem. It is user-friendly, as the user only needs to define an array of *[X,Y]* coordinates in the *X*-*Y* plane. Following which the user can extract the corresponding wave elevation $\eta(x,y,t)$ time histories and visualize them.

3.10 Nonlinear Hydro Updates

Unless otherwise specified, WEC-Sim linearized the hydrostatics calculation of a body so that the restoring force acts as a simple spring, the product of a constant stiffness and a displacement. This is inappropriate if the orientation of the body affects the hydrostatic restoring force. As a simple example, consider a pitching vertical cylinder: above a certain pitch displacement the cut water plane area and therefore the restoring force, will increase sharply. For these cases, where a linear treatment of hydrostatics is inappropriate, WEC-Sim offers a nonlinear hydrostatics option [21]. Similarly, BEM estimates of excitation coefficients are calculated assuming a static body position. If the body geometry is appropriate and displacement sufficiently large, excitation forces are likely to depend strongly on the instantaneous body position. An example of this is a bottom-mounted pitching flap. At high pitch displacements, the area of the flap subjected to an energetic wave field is sharply reduced. For these cases, WEC-Sim can calculate the non-linear excitation forces based upon the instantaneous body position.

Both calculation methods depend on a refined *.stl mesh defining the body surfaces, or more specifically, the z-position relative to the instantaneous free-surface, individual panel areas and panel normal vectors. From this, the hydrostatic and excitation forces can be calculated for each panel and then summed to reflect the total force exerted on the bodies.

Previously, this calculation method was defined per simulation, so that all hydrodynamic bodies would have these forces calculated. This has been updated to be a property of each individual body: in this way, the user can perform the higher order calculation on only the bodies where these nonlinearities will be significant, speeding calculation time while retaining accuracy. At this time, the large XY displacement option cannot be combined with either nonlinear hydrodynamics option; this is planned for future development.

3.11 Morison Element Updates

WEC-Sim first introduced Morison element forces [22] with the release of v1.3 [23]; however, there were limitations in that the forces along the x-, y-, and z-axes were handled independently, which would lead to certain theoretical inaccuracies depending on the wave conditions and selection of Morison element coefficients. In an effort to be more theoretically accurate, WEC-Sim added a second implementation option for the calculation of Morison element forces that followed a normal and tangential formulation. In mathematics, a vector can be uniquely decomposed into the sum of its tangential and normal components. The tangential component of the vector, $v_{\parallel}$, is parallel to the surface while the normal component of the vector, $v_{\perp}$, is perpendicular to the surface and is used in relation to the central axis to the Morison element. The WEC-Sim input file was altered to consider a tangential and normal component for the drag coefficient [$C_{d\perp}$, $C_{d\parallel}$], added mass coefficient [$C_{a\perp}$, $C_{a\parallel}$], characteristic area [$A_{\perp}$, $A_{\parallel}$], and the central axis unit vector of the Morison element is given by $\vec{z} = [z_x, z_y, z_z]$.

A general vector, $\vec{k}$, can be decomposed into the normal component as a projection of vector k on to the central axis $\vec{z}$ as follows:

$$\vec{k}_{\parallel} = \frac{\vec{z} \cdot \vec{k}}{||\vec{z}||} \frac{\vec{z}}{||\vec{z}||} \tag{7}$$

As the vector is uniquely decomposed into the sum of its tangential and normal components, the normal component can be defined as the difference between the vector and its tangential component as follows:

$$\vec{k}_{\perp} = \vec{k} - \vec{k}_{\parallel} \tag{8}$$

With the normal and tangential vectors of the fluid and body velocity and acceleration, the Morison element equation can be used to calculate the resulting force vector and along with the lever arm from the center of gravity will generate the resulting moment from the corresponding cross product. An illustrative test case is shown in Fig. 10 which calculates the Morison element force decomposition for a vertical element with a constant uniform flow rotating 360 degrees about the element. The resultant magnitude of the force should be constant, which is reproduced by Option 2, but there are variations in Option 1 based on the flow orientation. This is worst at 45 degrees when the flow is equally distributed between both *x*- and *y*-directions and by only squaring the individual flow components reduces the total force on the cylinder. Rather than remove Option 1, despite the theoretical discrepancies, both options were kept, as often Morison elements are used as a tuning parameter to account for fluid effects not captured by linear hydrodynamic theory, and thus Option 1 may still meet the modeler's needs.

4 CONCLUSION

Development and support of the WEC-Sim open-source software would not be possible without the continued support from the U.S. Department of Energy Water Power Technologies Office. This sustained support has allowed WEC-Sim to rapidly grow its user community over the past eight years from the first open-source release back in 2014. Since the initial release, WEC-Sim has focused on adding new features and capabilities to be able to simulate as wide a variety of device archetypes as possible [24]. The ability to model a wide variety of WEC concepts has made WEC-Sim one of the most popular TEAMER facilities to date with continued interest in obtaining direct support from the WEC-Sim development team. In addition, the WEC-Sim team has continued to develop a diverse set of open-access applications [19] highlighting the variety of use cases possible with WEC-Sim.

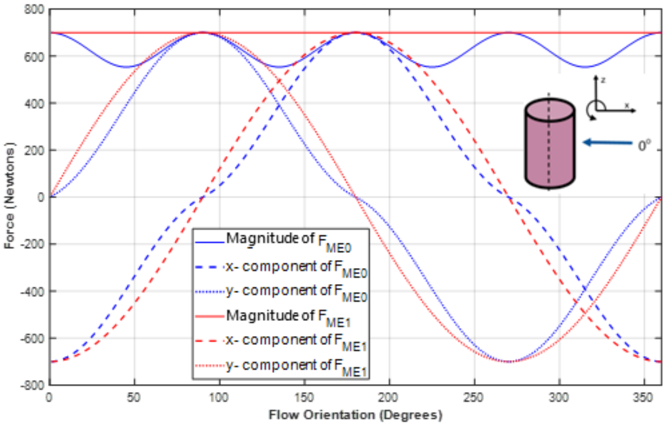


FIGURE 10. GRAPHICAL REPRESENTATION OF THE COMPARISON BETWEEN MORISON ELEMENT OPTION 1 AND OPTION 2 WITHIN WEC-SIM.

The software management updates presented in the paper have introduced improvements to the readability and usability of WEC-Sim. Building the unit test suite for WEC-Sim is ongoing work. Refactoring of the source code that was not written with unit testing as a priority is a challenging undertaking. The benefits, nonetheless, are multifaceted with improvements in test efficiency, code coverage, and working practices. Acceptance tests still run regularly to ensure a smooth user experience and record accuracy improvements from changes in the WEC-Sim software.

The software technical updates presented in the paper include improvements ranging from additional visualization capabilities to the inclusion of a spatial phase lag in the wave excitation forces. These improvements provide additional options for running WEC-Sim, like running from Simulink directly rather than the MATLAB command line, which is a key feature to simplify hardware-in-the-loop applications. The updates to PTO-Sim and a new cable implementation allow users to better represent their power conversion systems to increase the fidelity of their wave-to-wire models. The updates to WEC-Sim nonlinear hydro and Morison element functions have improved the fidelity and accuracy of existing capabilities. Updates to BEMIO allow users to utilize a wider range of BEM software and improved calculation of the hydrodynamic radiation coefficients.

On top of new technical and management code development, the WEC-Sim team at NREL and Sandia continue to support the user community by maintaining the GitHub repository, improving code usability, responding to user issues, resolving identified bugs, and improving code documentation. The WEC-Sim team is happy to see the use of the software to model diverse WEC archetypes to date but believes that there is still room for significant code improvement to further expand the application space. These improvements may include allowing for the calculation of quadratic transfer functions (i.e. sum and difference frequency forces) and for modeling multiple and distinct wave fronts to better capture wind and swell seas.

ACKNOWLEDGMENT

Sandia is managed and operated by NTESS under DOE NNSA 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.

REFERENCES

[1] Babarit, A., Hals, J., Muliawan, M. J., Kurniawan, A., Moan, T., and Krokstad, J., 2012. “Numerical benchmarking study of a selection of wave energy converters”. *Renewable Energy,* ***41***, pp. 44–63.

[2] Topper, M. B. R., 2011. “Identifying the Frontier of Knowledge for Marine Renewable Energy Research”.

[3] Yu, Y.-H., Lawson, M., Ruehl, K., and Michelen, C., 2014. “Development and Demonstration of the WEC-Sim Wave Energy Converter Simulation Tool”. In 2nd Marine Energy Technology Symposium, METS.

[4] Ruehl, K., Michelen, C., Kanner, S., Lawson, M., and Yu, Y.-H., 2014. “Preliminary Verification and Validation of WEC-Sim, an Open Source Wave Energy Converter Design Tool”. In 33rd International Conference on Ocean, Offshore and Arctic Engineering, OMAE.

[5] Cummins, W., 1962. “The impulse response function and ship motions (no. dtmb-1661)”. *David Taylor Model Basin Washington DC*.

[6] Lee, C. H., and Newman, J. N., 2006. “Wamit user manual”. *WAMIT, Inc*, p. 42.

[7] NEMOH. https://github.com/LHEEA/Nemoh. Accessed: 12-2022.

[8] Capytaine. https://github.com/capytaine/capytaine. Accessed: 12-2022.

[9] ANSYS-AQWA. https://www.ansys.com/. Accessed: 12-2022.

[10] Ruehl, K., Keester, A., Tom, N., Topper, M., Forbush, D., Grasberger, J., Husain, S., Ogden, D., and Leon, J., 2022. “Wec-sim/wec-sim: Wec-sim v5.0.1”.

[11] WEC-Sim – Wave Energy Converter SIMulator. http://wec-sim.github.io/WEC-Sim/. Accessed: 1-2023.

[12] Lockard, A., Vanzwieten, T., Mann, J., Keenan, M., Behrend, C., and Connell, B., 2021. “NASA’s Orion Crew Module Seakeeping Simulation and Test Comparisons in Ocean Wave Environments”. In Oceans 2021 Conference and Exposition, IEEE.

[13] US DOE Water Power Technologies Office, 2022. 2022 Peer Review. https://www.energy.gov/sites/default/files/2022-08/wpto-foundational-rd-day3-wec-modeling.pdf. Accessed: 1-2023.

[14] R&D World, 2021. R&D 100 Award in Software and Services. https://www.rdworldonline.com/rd-100-2021-winner/wec-sim/. Accessed: 1-2023.

[15] Li, L., Gao, Y., Yuan, Z., Day, S., and Hu, Z., 2018. “Dynamic response and power production of a floating integrated wind, wave and tidal energy system”. *Renewable Energy,* ***116***, pp. 412–422.

[16] Ringwood, J., Ferri, F., Tom, N., Ruehl, K., Faedo, N., Bacelli, G., Yu, Y.-H., and Coe, R. G., 2019. “The Wave Energy Converter Control Competition: Overview”. In Proceedings of the 38th International Conference on Ocean, Offshore and Arctic Engineering, OMAE 2019, no. June 9-14.

[17] TEAMER, 2022. Testing & Expertise for Marine Energy. https://teamer-us.org/. Accessed: 12-2022.

[18] Wendt, F., Nielsen, K., Yu, Y.-H., Bingham, H., Eskilsson, C., Kramer, M., Babarit, A., Bunnik, T., Costello, R., Crowley, S., Gendron, B., Giorgi, G., Giorgi, S., Girardin, S., Greaves, D., Heras, P., Hoffman, J., Islam, H., Jakobsen, K., Janson, C., Jansson, J., Kim, H., Kim, J., Kim, K., Kurniawan, A., Leoni, M., Mathai, T., Nam, B., Park, S., Rajagopalan, K., Ransley, E., Read, R., Ringwood, J., Rodrigues, J., Rosenthal, B., Roy, A., Ruehl, K., Schofield, P., Sheng, W., Shiri, A., Thomas, S., Touzon, I., and Yasutaka, I., 2019. “Ocean energy systems wave energy modelling task: Modelling, verification and validation of wave energy converters”. *Journal of Marine Science and Engineering,* ***7***(11), p. 379.

[19] WEC-Sim_Applications. https://github.com/WEC-Sim/WEC-Sim_applications. Accessed: 1-2023.

[20] Ogilvie, T., 1964. “Recent progress towards the understanding and prediction of ship motions”. In The Fifth Symposium on Naval Hydrodynamics, pp. 3–79.

[21] Lawson, M., Yu, Y. H., Nelessen, A., Ruehl, K., and Michelen, C., 2014. “Implementing nonlinear buoyancy and excitation forces in the WEC-SIM wave energy converter modeling tool”. *Proceedings of the International Conference on Offshore Mechanics and Arctic Engineering - OMAE.*

[22] Morison, J. R., Johnson, J. W., and Schaaf, S. A., 1950. “The force exerted by surface waves on piles”. *Petroleum*

Technology, ***2***(5), pp. 149–154.

[23] Tom, N., Lawson, M., and Yu, Y.-H., 2015. "Demonstration of the Recent Additions in Modeling Capabilities for the WEC-Sim Wave Energy Converter Design Tool". In 34th International Conference on Ocean, Offshore and Arctic Engineering, OMAE, ed.

[24] Ogden, D., Ruehl, K., Yu, Y.-H., Keester, A., Forbush, D., Leon, J., and Tom, N., 2022. "Review of WEC-Sim Development and Applications". *International Marine Energy Journal,* ***5***(3).