Post Access Report

Numerical Modeling & Optimization of the iProTech Pitching Inertial Pump (PIP) Wave Energy Converter (WEC)

Awardee: iProTech

Awardee point of contact: Nick Wynn

Facility: NREL

Facility point of contact: Stein Housner

Date: January 16th, 2024

TEAMER
Testing & Expertise for Marine Energy

EXECUTIVE SUMMARY

This project aims to develop an automated workflow between various Python packages and the WEC-Sim time-domain numerical simulation tool in MATLAB to efficiently evaluate the performance of WECs and use its time-domain outputs to inform subsequent design iterations. iProTech's Pitching Inertial Pump (PIP) WEC is used as the test WEC and is parameterized into a series of independent variables (design variables) and dependent variables that can be used to create a mesh of the WEC, calculate its hydrodynamic coefficients, and simulated in a time-domain environment. This process is wrapped into an optimization function to automatically calculate time-domain results and use those results to inform how the optimizer should adjust design variables. Various single variable optimization studies are performed to determine the relative influence that each variable has on the power output of the device. A complete design optimization of the entire PIP WEC was not included as this involves more advanced optimization algorithms and better software organization and architecture.

The PIP WEC uses an internal water coil to generate electricity through pressure differences. The outer hull can be parameterized into three adjoined circles: a nose, a bottom, and a stern, and the internal water coil can also be parameterized into variables such as the diameter of the water coil, or the stiffness and damping of the numerical PTO system. Each of these design variables is optimized to maximize the power output of the device over a range of regular wave periods (running irregular wave simulations increases the computation time dramatically). The diameter of the internal water coil proved to be the most influential, as the moment of inertia of the coil and the hull have a direct correlation to the relative pitch motion of the device (power output). The bottom circle's radius has the highest influence on power output among the geometric design variables. The remaining design variables have less influence on the device, mostly due to the initial design description of the PIP WEC.

Two out of the three main objectives of the project were met. The automation of the workflow between Python packages and WEC-Sim in MATLAB was completed and refined to where each creation and simulation of a design took on the order of 1-3 minutes. A sensitivity study of how different design variables affect the power output of the WEC was completed. A full system optimization, however, was not completed due to the need to integrate other more advanced optimizers and a reorganization of the design tool's software.

Regardless, this work set up the start of a design tool that can incorporate the effects included in a time-domain environment. Expanding on this tool can promote future design optimization studies of WECs that include the time-domain. For the PIP WEC, the developers can focus on certain design variables over others to continue optimizing their design. Continued work should involve the reorganization of the design tool's software to better integrate other optimizers and post-analysis scripts to fully optimize the design of any WEC of any function.

1 INTRODUCTION TO THE PROJECT

The iProTech Pitching Inertial Pump (PIP) concept features a single floating hull, slack moored, pitching in waves (Figure 1). Inside the system's hull is a novel inertial loop pump, which is rigidly attached to the hull's interior and is completely filled with water. The coil ends are connected to a valve manifold and hydraulic accumulator arrangement.

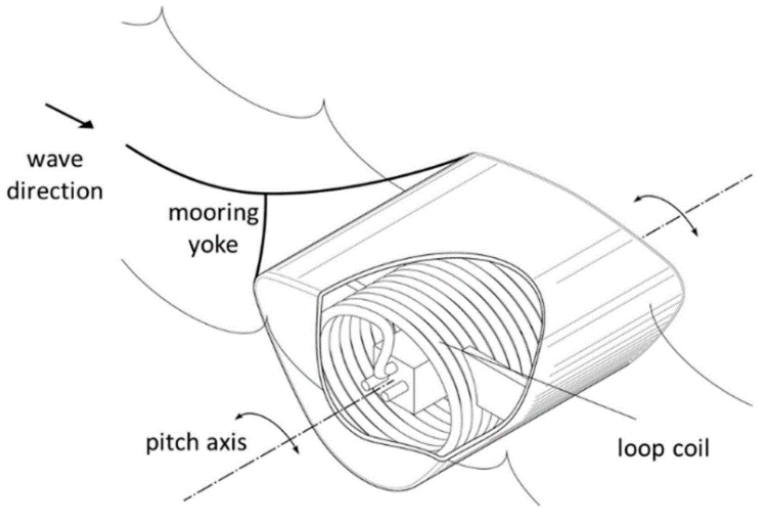

Figure 1. PIP Concept with Inertial Loop Coil.

Under passive control, the coil water content moves with the device. When the pitch induced angular acceleration is sufficient to overcome the accumulator pressure difference, the check valves open and water passes from the low pressure accumulator, through the coil, into the high pressure accumulator.

Under active control, a bypass valve is manipulated to permit the water coil and hull to move independently. Closing the valve initiates a water hammer event, which causes the check valves to open and water to pass through the coil.

Under TEAMER RFTS 1, tools were developed to automatically generate meshes of the hull and compute hydrodynamic coefficients. Two time-domain models were also developed with WEC-Sim; a faster model with a linearized PTO (spring-damper) and a more accurate model with a fully nonlinear PTO – featuring a detailed representation of the system's hydraulic circuit. Good agreement was observed between the two models when comparing power response curves – demonstrating the viability of using the faster linearized model for geometry and mass distribution optimization studies.

Because the device only absorbs power from the relative pitch motion of the hull and the internal water coil, it is critical to optimize the system for maximum relative pitch motion. In RFTS 1 several variables were identified that were highly influential on pitch motion – including the angle of attack of the PIP's hull nose.

However, only the meshing and BEM portions of the workflow could be automated (in Python), and once all of the geometric data and hydrodynamic coefficients for each design iteration were saved to

disk, they could be simulated in WEC-Sim as a batch run. In other words, there is currently no way to use the time-domain outputs from WEC-Sim to feedback into the meshing stage. Automating this step is critical for optimizing the system. Hence, one of the main goals of this project is to fully automate the existing workflow and wrap it with an optimizer to more rigorously explore the PIP's design space and help determine the PIP's optimal form. To our knowledge, this kind of optimization study has not yet been performed with WEC-Sim (although it is common in other engineering fields).

2 Roles and Responsibilities of Project Participants

2.1 Applicant Responsibilities and Tasks Performed

Nick Wynn: project coordination, oversight & technical advice

2.2 Network Facility Responsibilities and Tasks Performed

David Ogden: automation of meshing, BEM, and time-domain modeling workflow

Matt Hall, Stein Housner: selection of optimizer and performing optimization studies

3 Project Objectives

The objectives of this RFTS 6 project are to:

1 Determine the optimal geometry, mass distribution and operating parameters for the PIP device in a specific wave climate – to maximize annual power capture
   - 1.1 This will involve integrating numerical optimization tools with an automated modeling workflow
   - 1.2 Various open-source optimization frameworks will be used (e.g. NASA's OpenMDAO or NLopt) – to see what works best.

2 Improve the current WEC-Sim based models, in particular to automate the workflow and develop a 'design recipe' script (i.e. create a full system description from design variables).
   - 2.1 The current workflow exists in 2 siloes: meshing and BEM runs are automated with Python, whereas time-domain modeling is performed separately with WEC-Sim in MATLAB/Simulink and Simscape Multibody. There is currently no clear way to feed back WEC-Sim outputs to the meshing stage.
   - 2.2 In order to optimize the PIP, WEC-Sim results need to inform the mesher automatically; hence the whole workflow needs to be fully coupled and automated.
   - 2.3 This could be achieved by calling Python from MATLAB, and this approach will be tried first. If this proves too challenging, we will consider leveraging an open-source multibody dynamics solver (e.g. Chrono and its Python API, PyChrono) and automate the whole workflow in Python. Any change in approach will be reviewed/discussed fully before implementation.

3 Develop an understanding of where performance is most sensitive to design and operating parameters and how these interact

3.1 Building on RFTS 1 studies, the design space will be explored more extensively (parameter sweeps) to better understand the system's sensitivity with respect to different variables. In addition, a wider range of variables will be considered (with finer resolution), including:

- 3.1.1 Nose location in x
- 3.1.2 Nose radius
- 3.1.3 Stern location in x
- 3.1.4 Stern radius
- 3.1.5 Bottom location in z
- 3.1.6 Bottom radius
- 3.1.7 Mass and inertia of the water coil
- 3.1.8 Mass and inertia of the hull
- 3.1.9 C of G of the device

3.2 To better understand the impact of design space changes, intermediate parameters which are physically meaningful will be generated during the workflow, e.g (i) Angle of Attack (NoseDown Angle), (ii) Center of Buoyancy, (iii) Metacentric Height, (iv) Natural Pitch Period (both coupled and uncoupled with water coil). The relevance of these parameters can then be assessed by correlating with the performance objective function.

4 Test Facility, Equipment, Software, and Technical Expertise

NREL develops, maintains, and supports free, publicly available tools to help the wave energy community analyze technology designs. The award-winning Wave Energy Converter SIMulator (WEC-Sim), developed by NREL and Sandia National Laboratories, simulates WECs in the time-domain to capture nonlinearities in the system and assess device performance. In addition, NREL has expertise in automated mesh generation and supports the development of the open-source BEM code Capytaine, which will be leveraged in this project to compute hydrodynamic coefficients.

Efficient design exploration of WECs with WEC-Sim is a relatively unexplored space (most studies so far have focused on parameter sweeps). However, NREL has extensive expertise with optimization frameworks (e.g. OpenMDAO, NLopt) and has used them for wind turbine design, plant layout, building HVAC optimization and battery architecture design.

5 Test or Analysis Article Description

PIP Technology and Working Principles

The PIP WEC captures wave energy using a novel inertial loop pump, housed in a slack-moored floating body that pitches in waves. The body profile resembles the device shown in Figure 2 but the internal inertial loop pump PTO system is completely new.

Figure 2. Physical scale model testing of the Pitching WEC.

Inertial Loop Pump: Figure 3 shows schematically the inertial loop coil pump installed in PIP. The loop coil is rigidly attached inside the pitching body and filled completely with water.

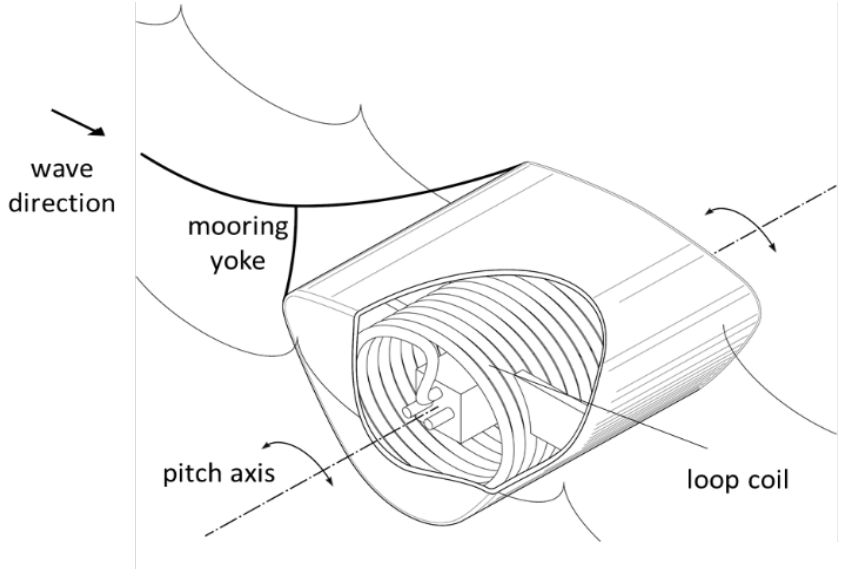

Figure 3. PIP concept with inertial loop coil.

PTO Valves, Hydraulic Accumulators, Control Options: Figure 4 shows how the coil ends are connected to a valve manifold and hydraulic accumulator arrangement. Under passive control, the bypass valve V_S is kept closed and lock valve V_L open, so the coil water content moves with the device. When the pitch induced angular acceleration is sufficient to overcome the accumulator pressure difference, the check valves open and water passes from the low pressure accumulator, through the coil, to the high pressure accumulator.

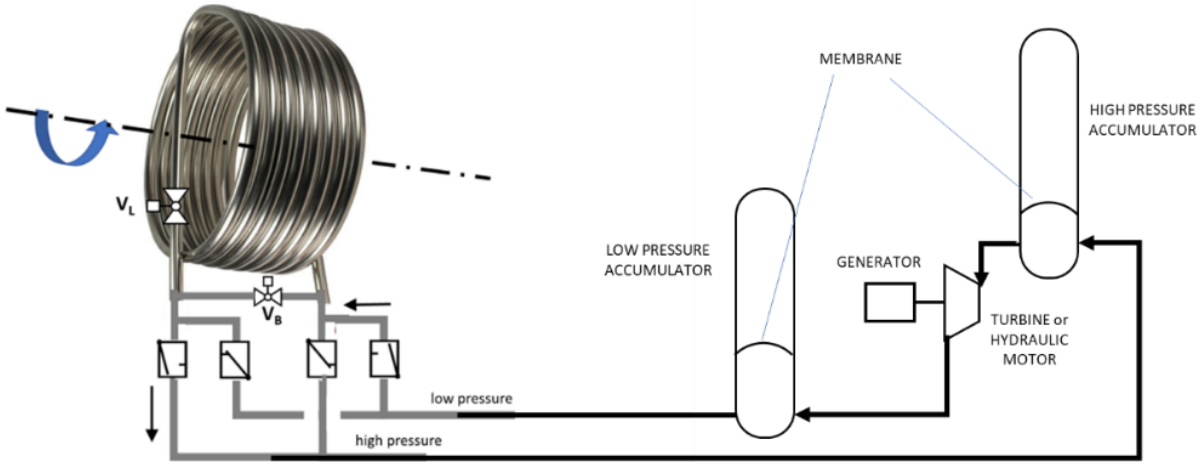

Figure 4. Flow schematic of valves and hydraulic accumulators.

Manipulating bypass valve V_S allows active control of the device. Running with V_L and V_S open allows the device body and water content to move independently. Closing V_S to stop relative motion will initiate a water hammer event, opening the check valves and also pumping water. These scenarios are displayed in Figure 5. Timing this opening/closing to maximize pitch motion will enhance power capture.

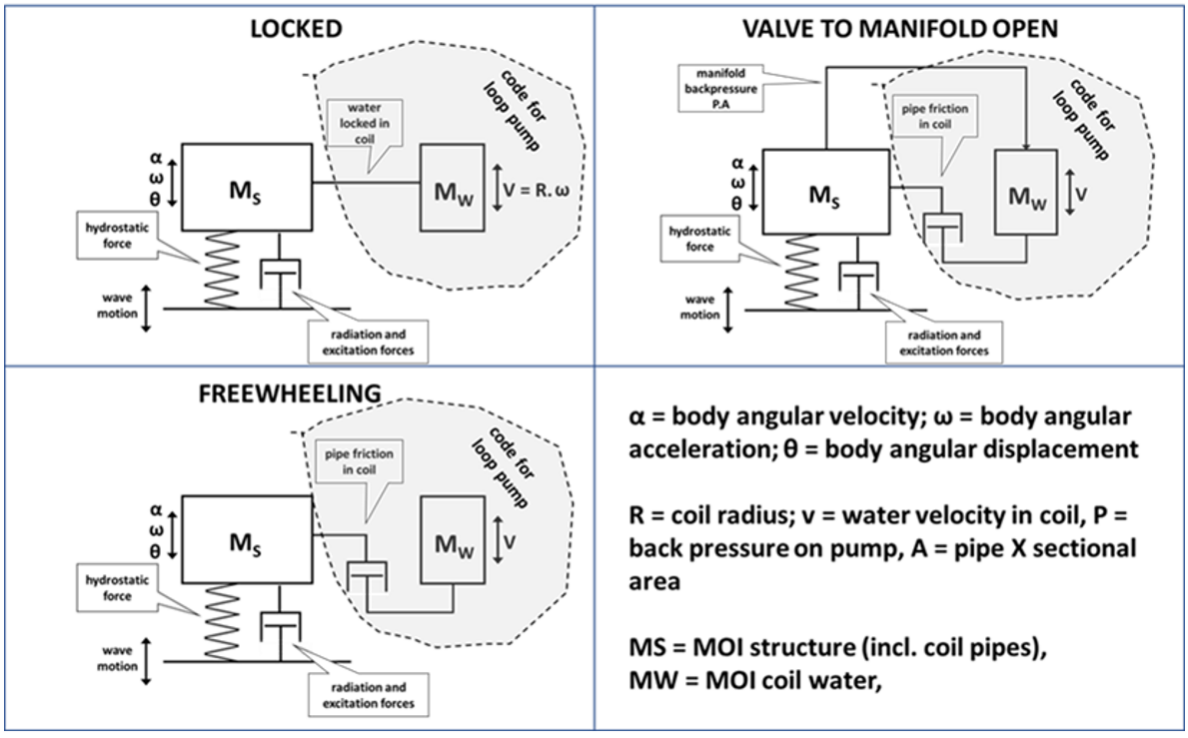

Figure 5. Overview of PIP control states.

Unique Attributes of the PIP device

Robustness: Storm forces on WEC's can be 50 x normal conditions. PIP is a two body (structure/coil and water) device with no joints, bearings or concentrated loads and is slack moored. Stresses from inertial and hydrostatic loads are distributed over the whole body.

Power capture: The system's pitch response can be optimized for the target scale and sea state. Active wave-to-wave control times valve opening/closing to maximize power capture. Hydraulic accumulators smooth power output over one or two wave cycles.

The above attributes promise low CAPEX and OPEX in many WEC applications. Hydraulic power capture can be further converted by conventional hydraulic amplifiers, hydraulic motors, or turbines, depending on the application.

6 Work Plan

The following subsections provide the methods used to achieve the results.

6.1 Numerical Model Description

The design tool developed consists of a combination of two primary numerical simulation tools, WEC-Sim and Capytaine, detailed below, as well as the integration of optimization algorithms.

WEC-Sim

WEC-Sim is a mid-fidelity WEC numerical modelling tool based on linear potential flow theory. Hence, the wave field is considered to be a linear superposition of incident, radiated, and diffracted regular wave components. Boundary Element Method (BEM) codes are used to compute a body's hydrodynamic coefficients (e.g. added mass, damping, excitation force) for a range of discrete frequencies. BEMIO is then used to pre-process the hydrodynamic coefficients and save them to a .h5 file that can be read by WEC-Sim.

WEC-Sim then uses these frequency-domain coefficients in time-domain formulations of the hydrodynamic forces. This conversion is required to model the WEC system in the time-domain, which is necessary to include non-linearities in the system - such as joints, PTOs, control systems, moorings etc.

A complete description of the code's theory is available on the WEC-Sim website: https://wec-sim.github.io/WEC-Sim/man/theory.html

The accuracy of WEC-Sim has been verified in code-to-code comparisons and validated against experimental data. A full list of relevant publications is also available on the WEC-Sim website: https://wec-sim.github.io/WEC-Sim/man/publications.html

Capytaine

Capytaine is Python package for the simulation of the interaction between water waves and floating bodies in frequency domain. It is built around a full rewrite of the open source BEM solver Nemoh for

the linear potential flow wave theory. Capytaine can compute the following hydrodynamic coefficients for rigid bodies:

- Added mass
- Radiation damping
- Diffraction force
- Froude-Krylov force

In addition, Capytaine has the ability to compute hydrostatic stiffness matrices for a given wetted surface. Hence, Capytaine requires a mesh input file – which can be generated automatically using NREL's bespoke Python scripts or using an open-source meshing tool such as pygmsh. Another open-source python package, meshmagick, was used to generate meshes of the WEC for the design tool.

The outputs of Capytaine can be read by BEMIO for use with WEC-Sim.

Optimization

Several candidates of optimization models have potential to be integrated with the design tool (e.g., SciPy, NLopt, and OpenMDAO) and the automated workflow is set up to easily switch between different optimizers to compare performance. The details of these optimization algorithms can be found in their respective documentations. The general optimization workflow is shown in Figure 6.

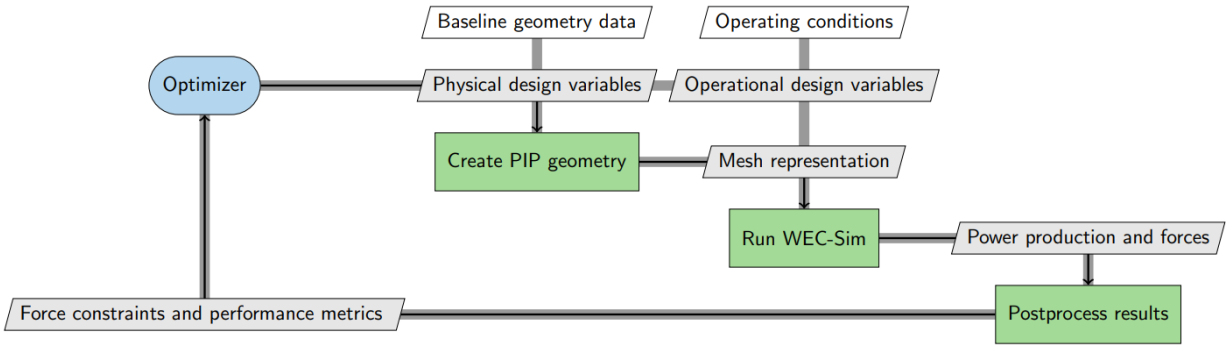

Figure 6. PIP optimization XDSM diagram.

6.2 Test and Analysis Matrix and Schedule

Task Names & Descriptions

Task		Description
1	Augment the existing workflow by calling WEC-Sim from Python	Currently all of the optimizers, meshing and BEM codes are written in Python and must be run separately from WEC-Sim. This task will develop an integrated workflow where WEC-Sim is called from Python. The first month of the project will focus on this integration. If this proves too challenging, we will consider leveraging an open-source multibody dynamics solver (e.g. Chrono and its Python API,

		PyChrono) and automate the whole workflow in Python. Any change in approach will be reviewed/discussed fully before implementation.
2	Automate the modelling process (from geometry creation to WEC performance metrics)	This largely involves building on the previous task to develop an integrated workflow for modelling the PIP. This would also involve parameterizing the model design and using those parameters to refine the pre-processing meshing scripts. Then, once the time-domain solver has finished, the post-processing analysis scripts would be integrated, and this whole process could be automated. Some parameters will be sea-state dependent – hence an initial goal will be to optimize the PIP's geometry and PTO parameters to maximize average power for the sea state containing the most energy at a particular location (e.g. PMEC).
3	Integrate numerical optimization frameworks into the automated design process	After the complete modelling workflow has been fully coupled, the process needs to be wrapped by an optimizer to automatically update the system's geometry and mass variables using the time-domain results.
4	Perform prototype design optimizations of the PIP geometry	Run the automated design process loop for various numerical optimization frameworks to design preliminary geometries of the WEC. Determine which parameters are the most sensitive by performing parameter sweeps. Interpret the resulting optimal designs and determine any formulation changes that are needed.
6	Add capability to the optimization framework needed to capture realistic design constraints	Develop the objective function and constraints of the optimization problem to represent the characteristic of the physical system. Investigate differences between different optimization methods or different combinations of design variables.
7	Produce final optimal design results for the PIP geometry and write paper and Post Access Report on the optimal designs and methodology used to obtain them	Using the improved capabilities from Task 6, rerun the optimization procedure to find the ideal geometry configuration of the PIP to maximize the relative pitch motion of the device, which will maximize the power output. Report all findings in a paper to be submitted

Schedule

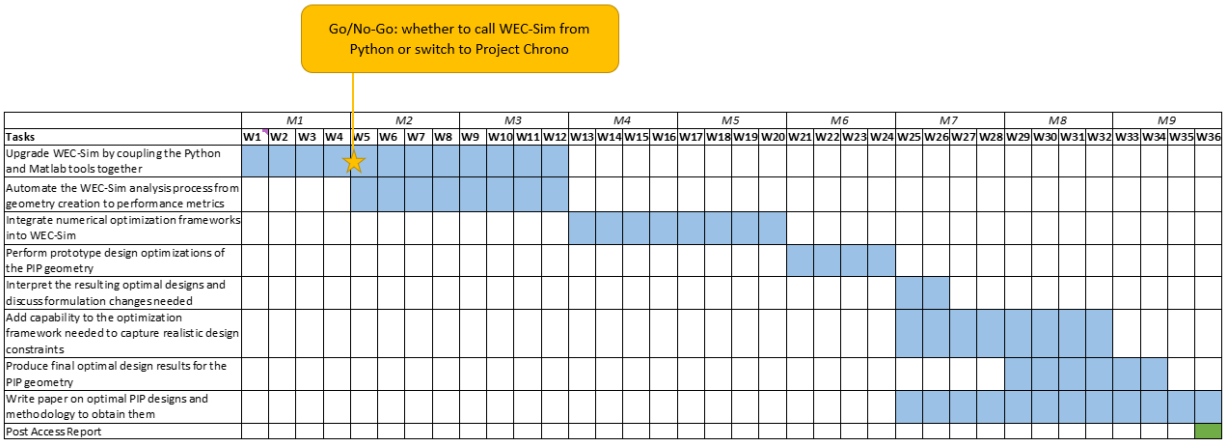

Figure 7. Project schedule.

6.3 Safety

Desk based study; all typical office safety measures will be adhered to.

6.4 Contingency Plans

The major first task of this project involves calling WEC-Sim from Python (in order to provide time-domain results to the optimizer).

If this task proves too challenging, we will consider two contingency plans:

1 Leveraging an open-source multibody dynamics solver (i.e. PyChrono) and automating the whole workflow in Python.
2 Building Chrono models from C++ as .exe files, and calling these using Python's subprocess module.

Core WEC-Sim functionality has already been re-written for use with Project Chrono and verified. Python API development is ongoing.

Any change in approach will be reviewed/discussed fully before implementation.

6.5 Data Management, Processing, and Analysis

6.5.1 Data Management

GitHub will be used for model version control and may be used to back up data as well. If the storage amounts become too high (currently there is a 5 GB limit on GitHub), OneDrive could be used to back up and share data.

For each design iteration, key files will include:

- System information file (.txt)
  - Records the geometry properties, mass properties, design name and all other relevant input data for a particular design iteration
- Mesh data file (.nemoh)
  - Standard Nemoh-convention mesh file for the design iteration
- Capytaine input file (.py)
  - Contains all of the relevant input data for the BEM code (frequency range, resolution, dofs, etc)
- Hydrodynamic coefficients (.nc and/or .h5)
  - .h5 generated by BEMIO
- Time-series data for power and kinematics (.mat file if using MATLAB, .txt if using Python/C++)

These data for each design iteration will be contained in separate directories with meaningful names.

6.5.2 Data Processing

From RFTS 1, an ideal mesh resolution (providing good balance between accuracy and speed) has been determined. Meshes generated automatically will be set to provide similar resolutions.

All meshes and hydrodynamic coefficients generated within the workflow will be saved to disk. Plotting scripts will enable convenient checking of hydrodynamic coefficients and radiation impulse response functions (RIRFs) – two common sources of error in time-domain results.

GitHub will be used back up the models, provide version control and backup/share results. If the 5GB limit is reached, results will instead be shared via OneDrive (1TB limit).

In order to build confidence in the results, the optimized design should be simulated using a higher fidelity approach (i.e. CFD or SPH). This is not included in the current scope of work but a potential strategy for this validation effort will be described in the final report.

6.5.3 Data Analysis

The time-domain results of the time-domain simulations will be analytically and graphically checked in each iteration of the optimization loop to inform the adjustment of the next iteration's design variables. The objective function—relative pitch motion of the PIP—and various constraints, such as maximum horizontal offset or maximum mooring line tensions, will be computed in each iteration and checked to ensure they meet criteria set in design standards such as IEC TS 62600-10 [1].

The parameterized variables of each PIP WEC design and the results of the time-domain simulations of each design iteration were analytically and graphically checked throughout the optimizations. The results of each iteration are organized into readable graphs to show the power output of each new design at various wave periods. Objectives and constraints are computed in each iteration and checked to make sure they meet static equilibrium, as well as criteria set in design standards such as IEC TS 62600-10.

7 PROJECT OUTCOMES

7.1 RESULTS

The following discusses the results gathered during the project execution timeline. A default PIP WEC description was provided from TEAMER RFTS 1, which was used to select an appropriate, initial optimizer, and then perform optimization studies of different design variables.

Numerical Design Tool

The modeling and optimization procedures of the PIP WEC were all contained within one program, hereby referred to as the "numerical design tool", but otherwise known as "TOP-WEC" (Time-domain Optimization Platform for WECs), with a graphical explanation of how it works provided in Figure 8.

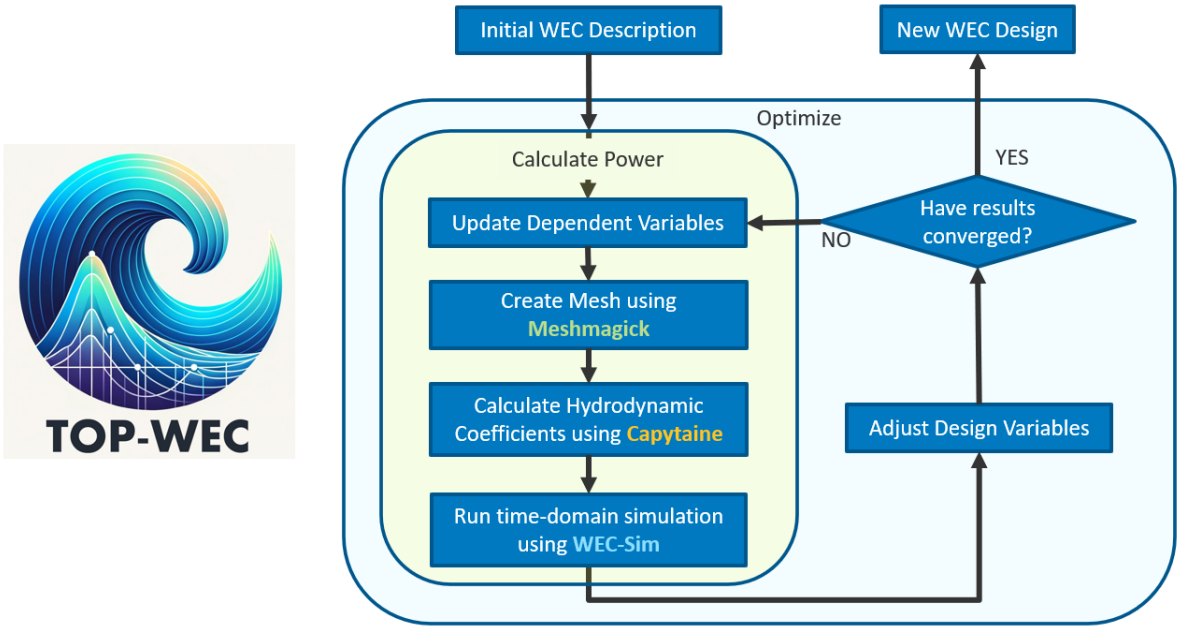

Figure 8. Graphical explanation of how the numerical design tool works.

The PIP WEC is initially parameterized into independent variables, which all have the option to be design variables if an optimization is desired. The main modeling functionality of the tool is to calculate the power output of the WEC, which involves a series of steps starting with: updating the dependent variables of the WEC, such as the mass of the internal coil, or the moment of inertia of the hull; creating a mesh from the geometry description using the Python-based meshing package, Meshmagick [2]; creating a Capytaine [3] object from the mesh to calculate the hydrodynamic coefficients, also in Python; and then sending the design description and hydrodynamic coefficients to WEC-Sim [4] in MATLAB to simulate the WEC design and output relevant parameters, such as average power. To optimize, the average power output can be considered the objective of the optimizer, where the optimizer can analyze the power output and adjust design variables to maximize that power, resulting in a new, optimized PIP WEC design (see page 16 for more detail on the optimization objective). These internal modules, such as Capytaine and WEC-Sim, are all verified and validated from previous studies and can be found in their respective documentations. Verification and validation was not required in this project as we were exploring different design options of the PIP WEC.

Default WEC Description

The default PIP WEC design is shown in Figure 9. This design is used as the starting point for different design variables to be optimized from. Other initial designs may produce different optimized results, but this design was chosen from the previous TEAMER project as a baseline.

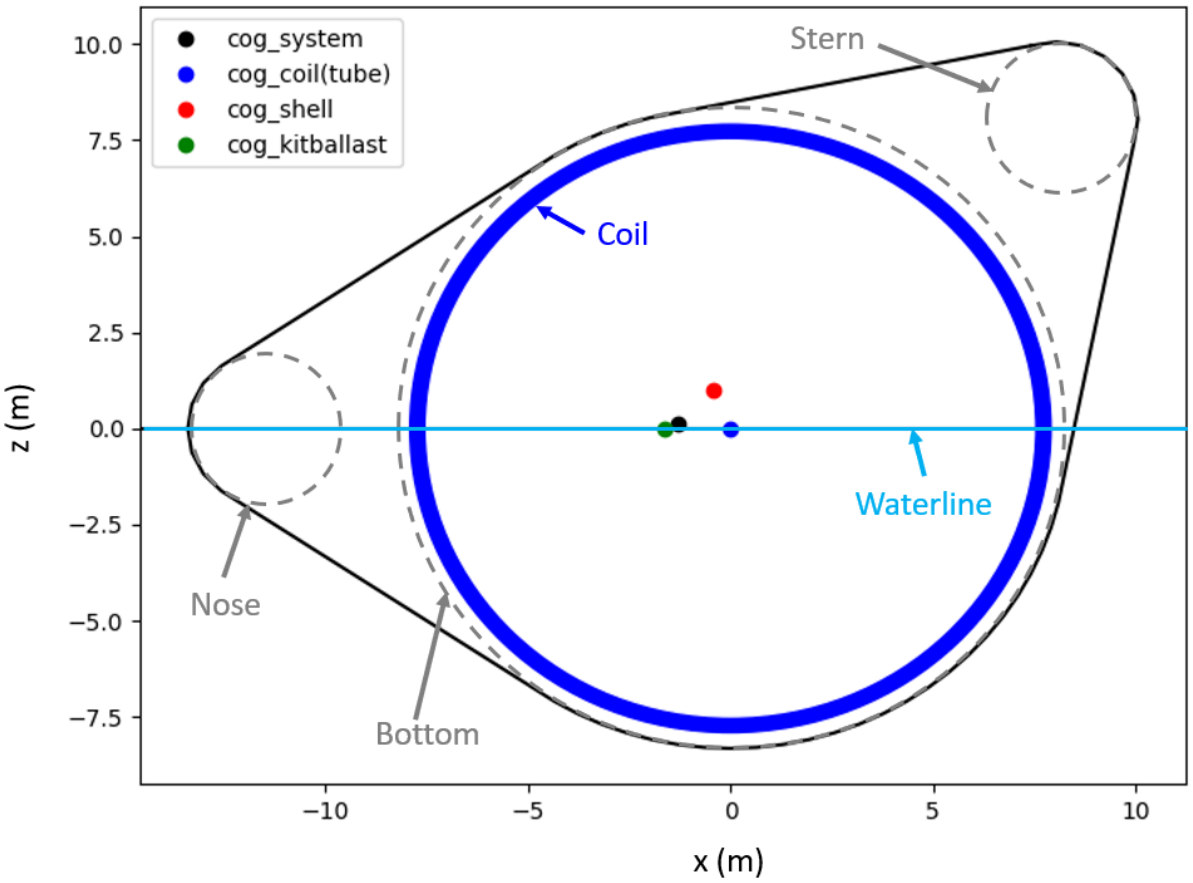

Figure 9. Default PIP WEC design.

The outer shape of the PIP can be defined by three circles: a nose, a bottom, and a stern, where the adjacent circles are connected by tangent lines. This shape is consistent along the width of the device, meaning that the 2D shape in Figure 9 can be extruded into or out of the page for a full 3D description. The bottom circle is assumed to always be centered around the origin (0,0), with a default radius of 8.325 m. The nose circle has a default x-position of -11.385 relative to the origin, and a default z-position of 0.0 m, on the waterline. The stern circle has a default angle relative to the origin of 45°, meaning that both the default x- and z-positions of the stern circle are 8.05 m. The default radii of the nose and stern circles are 2.01 m. Rather than include the bottom circle's x- and z-positions as design variables, we make the waterline a design variable, which can raise or sink the equilibrium PIP design in the water. The default width of the device is 24.48 m. Using these geometry descriptions, an outer shell thickness of 20 cm, and assuming a density of steel of 7,850 kg/m^3, the mass of the shell is 340 MT with a center of gravity of [-0.42, 0, 1.0] m.

The internal water coil of the PIP is represented as a series of hollow toruses (donut shapes) filled with water, where the water is able to flow through the coils and create pressure differentials to generate power. We have defined a default tube diameter of 0.42 m with a default tube thickness of 10% of the diameter, or 4.2 cm, and a default tube material density of 950 kg/m^3. The remaining volume within the tube is filled with water with a density of 1,000 kg/m^3. The radius of the coil, or the distance the outer

edge of the coil is from the bottom circle's center (the origin), has a default value of 7.9 m. The number of toruses that make up the coil is 55.9, which is dependent on the tube diameter and the width of the device (the mass and center of gravity calculations account for non-integers). The center of gravity of the coil is assumed to be equivalent to the center of the bottom circle, at the origin.

The total system mass is dependent upon the submerged volume of the device and the x-position of the total system center of gravity is always set to the x-position of the center of buoyancy, at -1.31 m from the origin (to always keep the device in equilibrium). Therefore, any changes to the default design description that affects the submerged volume, the total system mass and center of gravity in the x-direction will change. The z-position of the system center of gravity, however, is set to a default value of 0.11 m and can be used as a design variable.

Using the total system mass and center of gravity, and the shell and coil masses and center of gravity, the remaining mass that is required for mass equilibrium is defined as the "kit-and-ballast", to represent a "kit" of equipment near the coil needed for the power take off (PTO) systems, and any ballast within the device. The default total mass of the PIP is 3,154 MT, with the shell containing 340 MT, the coil tube containing 126 MT, and the water within the coil containing 235 MT. This leaves a significant mass of 2,452 MT to the kit-and-ballast for the default design.

Optimizer Study

One of the first steps towards optimizing the PIP design is selecting an optimization algorithm to automate the design process. Optimization algorithms are beneficial to reduce time spent designing and automatically finding an optimal design. However, different optimization algorithms work differently, and due to the computational time of using WEC-Sim "in-the-loop", an optimizer with excessive iterations will significantly increase the runtime of the design tool.

The initial work plan described testing multiple different optimization frameworks, such as SciPy [5], NLopt [6], and OpenMDAO [7], that each could employ different optimization methods. During project planning, it was deemed a higher priority to first produce different optimized designs of the PIP WEC, before comparing different optimization frameworks. As such, the SciPy framework was first integrated into the design tool given the low effort of integration complexity. Other frameworks like OpenMDAO would have involved a higher level of integration complexity (but could produce more optimized results).

Different SciPy optimization methods were run to optimize the x-position of the nose circle (noseX) to maximize the average power over time returned by the device in WEC-Sim for a given regular wave height and wave period. In other words, the optimizers found the design with a noseX that produced the highest power, calculated by WEC-Sim. Results are shown in Table 1.

Table 1: Optimization algorithm test results

	COBYLA	**Powell**	**SLSQP**	**Trust-constr**	**CG**	**Nelder-mead**	**TNC**
Optimal noseX (m)	-14.99	-14.96	-14.68	-13.77	-13.40	-14.95	-14.52
Average Power (kW)	251.6	251.7	251.2	235.5	224.8	251.6	249.9
Number of Iterations	11	46	70+*	44	32	42	39

Out of these seven SciPy optimization algorithms, all but two (trust-constraint and CG) produced similar objective results. There is not a significant difference in optimal noseX and average power between the remaining five optimizers, but there is a noticeable difference in the number of iterations it takes to achieve the optimal result. Starting at the default design and optimizing for noseX, the COBYLA optimization algorithm produced an optimal design in 11 iterations, while the other optimizers took significantly more time and iterations to end on a similar final result. Using WEC-Sim in-the-loop, each design iteration runs in order of minutes, which can significantly add up for detailed optimizers. Even though COBYLA is an optimizer that uses linear approximation and depends on the initial design variables, it is able to find an optimal result in a fraction of the time compared to other SciPy optimizers. This is the optimizer that will be used as the optimization algorithm in all following results. Other SciPy optimizers not in Table 1 were not included as they required other optimization parameters that weren't efficient, such as the calculation of a Jacobian, or they encountered other numerical errors during testing.

Wave Period Study and the Objective

Ideally, the design tool could simulate WEC-Sim in irregular waves to capture the more realistic sea states that the WEC would encounter. However, irregular waves in WEC-Sim would significantly add to the overall run time of the simulation, due to the calculation of the impulse response function. Instead, we simulate WEC-Sim in one wave height over a range of regular wave periods. An example wave period-average power curve is provided for the default PIP design in Figure 10.

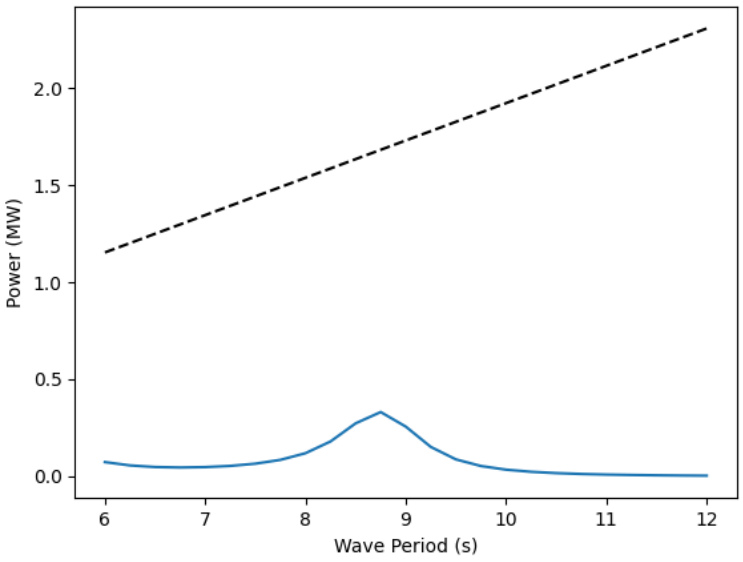

Figure 10: Average power vs. wave period for the default design

This curve defines the range of average power outputs of the PIP WEC for discrete wave periods. For the following optimization studies, the objective of the optimization is to maximize the area under this curve (referred to as the "period-power curve") to represent the design that can produce the most power over the range of wave periods. However, this area under the curve can vary depending on how many wave periods are simulated. Less wave periods can produce coarse and jagged curves, leading to inaccurate

areas under the curve but more wave periods can increase the computation time of the design tool significantly. We decided to simulate waves from 6 s to 12 s in increments of 0.25 s waves, meaning that each design is simulated in 25 regular wave periods, in one wave height (default value of 2 m).

The dashed black line shown in Figure 10 represents the total amount of power within the wave, to verify the accuracy of the numerical simulation results.

Single Variable Optimization Studies

The first step in optimizing the design of the PIP WEC is to start with single variable optimizations of different design variables. Based on the Optimization Study, the optimization algorithm, COBYLA, is employed. These single variable optimizations are relatively efficient and can provide certain insights into the influence that each design variable has on the objective. Starting at initial values defined by the default design, 12 design variables are optimized separately to produce designs that achieve the highest average power. Each design iteration is simulated in WEC-Sim in regular waves with a wave height of 2 m and wave periods ranging from 6-12 s. Bounds were applied to each design variable as to not produce irregular designs that could not be represented by a mesh or simulated in WEC-Sim. The final results are listed in Table 2 and further details are explained in the following sections.

Table 2: Single-variable optimization results using COBYLA

	Initial DV Value	Optimized DV Value	Optimized Objective (MW-s) (Area Under Curve)
Default Design	-	-	0.504
noseX (m)	-11.385	-9.333	0.781
noseZ (m)	0.0	2.677	0.687
noseR (m)	2.01	0.328	0.708
bottomR (m)	8.325	9.924	1.388
sternX (m)	8.05	6.618	0.521
sternZ (m)	8.05	*20.02	*0.635
sternR (m)	2.01	1.290	0.513
tilt (°)	0.0	*8.508	*0.908
zcg_system (m)	0.108	1.634	0.900
Dtube (m)		Not Optimized	
kPTO (N/m)		Not Optimized	
cPTO (Ns/m)		Not Optimized	

*Optimized design increases indefinitely past the bounds

Starting from the default design, the geometric design variable with the highest influence on the area under the curve is the bottom circle radius, with an increase from 0.504 to 1.388 MW-s area under the curve. This makes sense, as this has a large influence on the overall shape and size of the WEC, relative to other geometric variables. The noseX, noseZ, and noseR variables had the next highest influence on

power, as they define the initial part of the WEC that contacts the waves, while the sternX, sternZ, and sternR variables have less influence. The tilt of the device, as well as the center of gravity of the system in the z-direction had a higher influence on the power output of the device, other than the bottomR. However, the diameter of the tube that houses the internal water coil is by far the most influential variable on the device's performance. An optimization was not performed for Dtube, but small increases in the diameter result in large changes in power. The stiffness and damping of the PTO system in WEC-Sim were not as influential. Further details on these variables can be found below.

The following optimization studies provide a figure showing the change in geometry of the PIP throughout the optimization, a contour plot of the average power for each design iteration for each wave period, the wave period-average power plot with each line representing a different design, the average power over different designs with each line representing a different wave period, and the area under the curve of each line in the wave period-average power plot. Secondly, each design variable also underwent a parameter sweep (i.e., a manual optimization) that looped through incremental design changes to graphically show the true optimal designs.

noseX

The results of the noseX optimization using the COBYLA optimization algorithm with an initial value of -11.385 m are shown in Figure 11 and Figure 12.

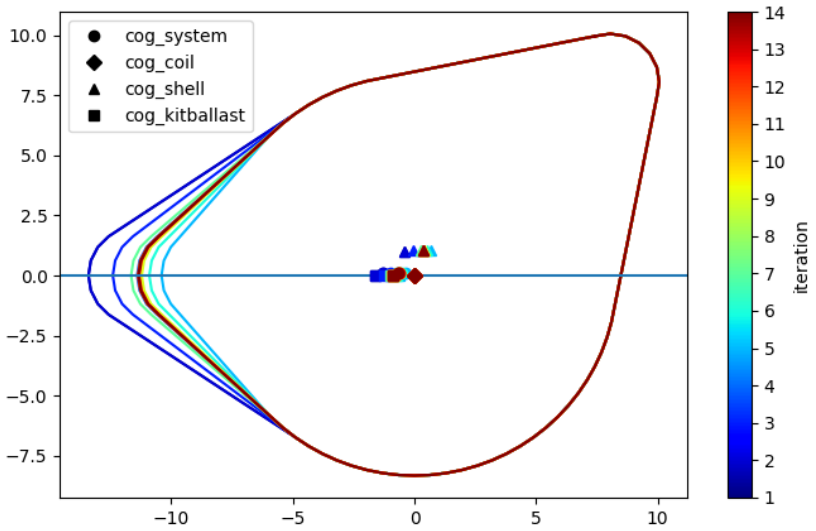

Figure 11: Iterated geometries for the noseX optimization

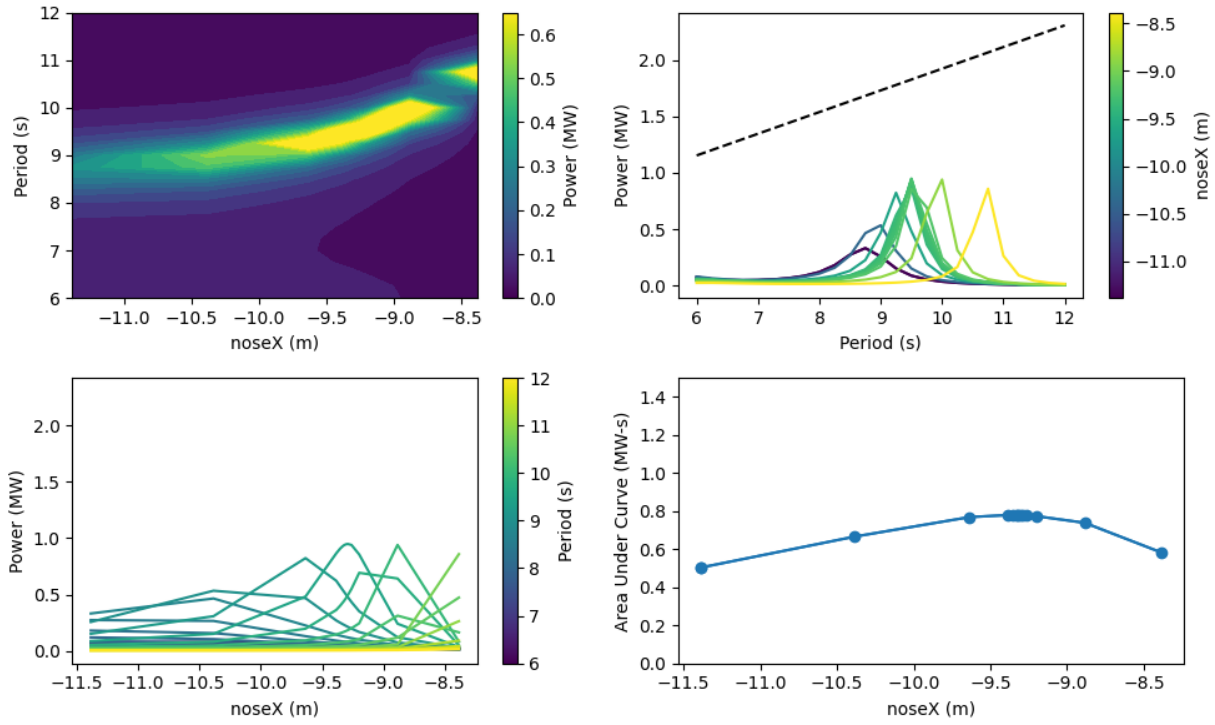

Figure 12: Power plots for noseX optimization

Using this wave period discretization and the default design parameters, it can be found that the optimal noseX position is at a value of -9.33 m. Intuitively, larger designs should produce higher power, but there happens to be a peak in area under the curve at a noseX value closer to the center than the initial noseX value. From the wave period-average power plot, noseX values between -10 and -8.5 all produce similar peak average powers, but at slightly different wave frequencies. Only at an intermediate value of -9.33 is there a maximum area under the curve over this range of wave periods. This only means that the noseX value of -9.33 m is only optimal for this range of wave periods, starting from the default design.

The natural frequency of the WEC can be seen to shift due to the change in geometry, which causes changes to the mass and hydrostatic stiffness matrices. Also, wave periods between 9 and 11 s seem to produce significantly more power than other wave periods. This can be seen by the yellow band of high power in the contour plot that hovers around 9 s for noseX positions farther than -9.5 m and then increases to 11 s for closer noseX positions. This is likely due to a matching of the wave number with the characteristic length of the device and shows that those wave period ranges produce significantly more power in the device than others.

A parameter sweep was performed of the noseX design variable to verify the optimal value of -9.33 m, shown in Figure 13.

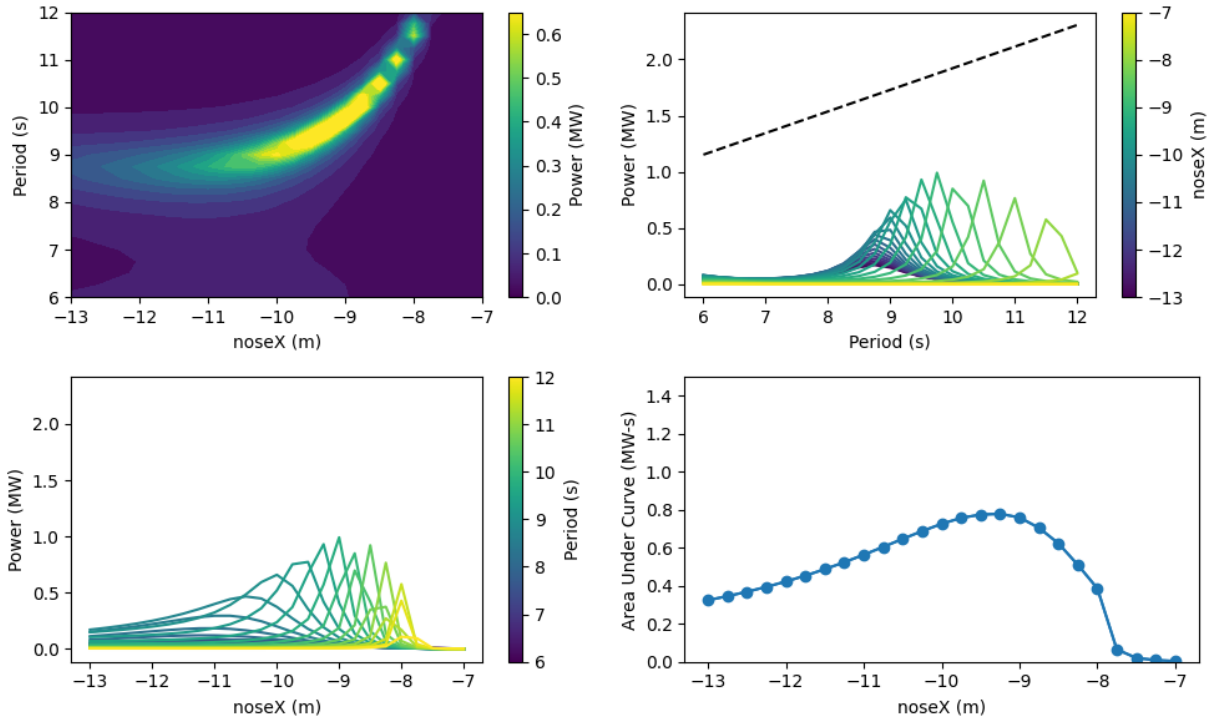

Figure 13: Power plots for noseX parameter sweep

This parameter sweep shows the same optimal noseX value for the given conditions and even shows a more refined band of high powers in the contour plot. The sharp decrease in area under the curve for closer noseX values is a factor of the range of wave periods simulated. As seen in the wave period-average power plot, the natural period slowly increases as the noseX gets closer to the PIP center and likely goes beyond 12 s for the close noseX values, where the design tool only considers wave periods up to 12 s.

noseZ

The results of the nose vertical position (noseZ) optimization using the COBYLA optimization algorithm with an initial value of 0 m are shown in Figure 14 and Figure 15.

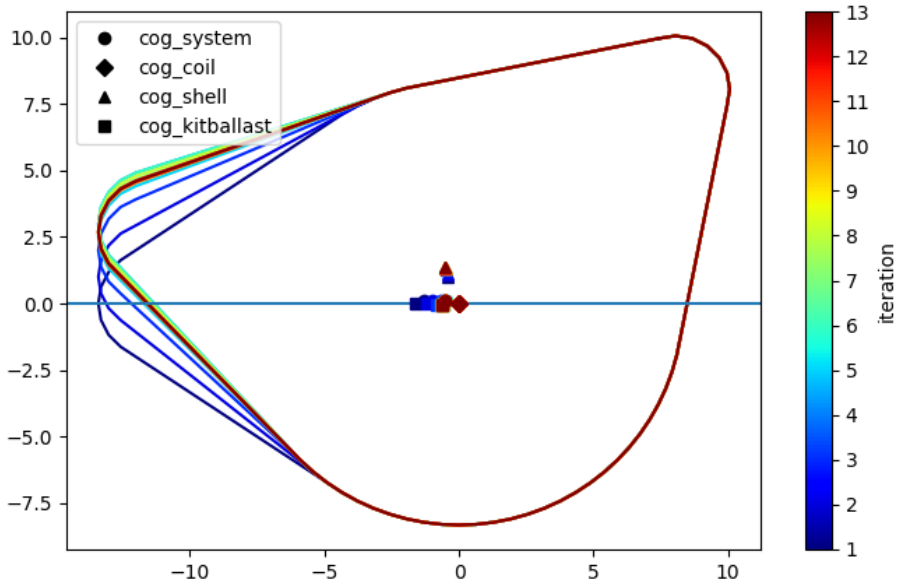

Figure 14: Iterated geometries for the noseZ optimization

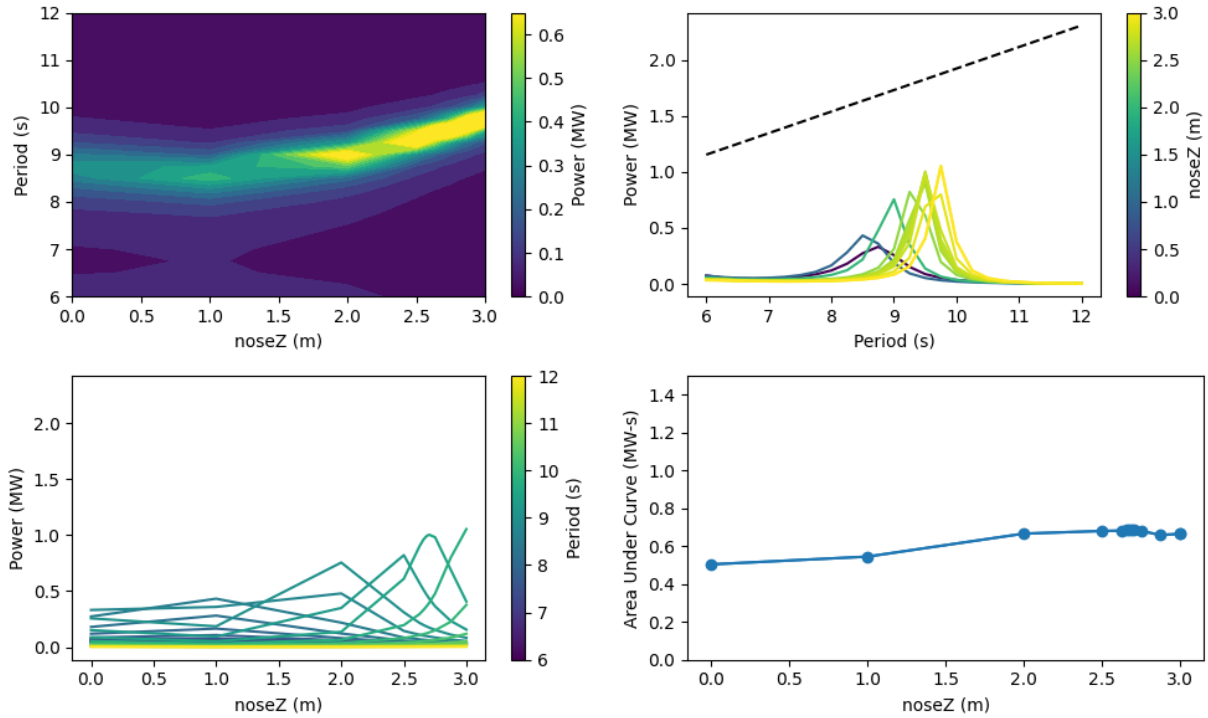

Figure 15: Power plots for noseZ optimization

Using this wave period discretization and the default design parameters, it can be found that the optimal noseZ position is at a value of 2.65 m. This makes intuitive sense, that a nose position above the water produces larger pitch motion due to the wave action on the underside of the PIP, relative to a nose position on the waterline. Larger noseZ values actually produced a higher average power for one wave period, but the overall area under the curve of the period-power plot at a noseZ of 2.65 m was

higher than the rest. The importance of the wave period discretization can also be seen here, as the design with a noseZ value of 2.8 m does not have a sharp peak in power, thus lowering the area under the curve. The natural period of the PIP can also be seen to increase as noseZ increases, but at a much lower rate than the noseX changes.

A parameter sweep was performed of the noseZ design variable to verify the optimal value of 2.65 m, shown in Figure 16.

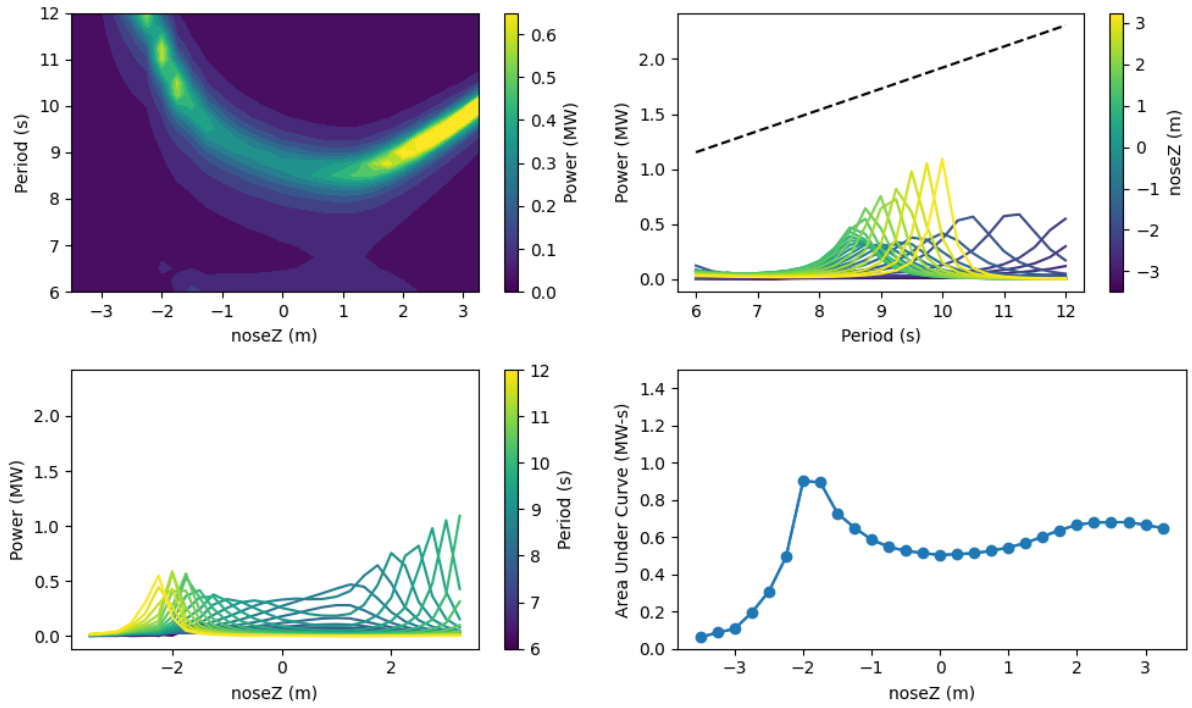

Figure 16: Power plots for noseZ parameter sweep

Interestingly, there is a design that the parameter sweep found that the optimizer did not find. A design with a noseZ position around -2 m has a higher area under the curve than the design at +2.65 m. This can be seen in the period-power plot to show the importance of the objective of the optimization. High noseZ positions produce average powers close to 1 MW for wave periods between 9 and 10 s, whereas the peak power produced by a design with a noseZ of -2 m is around 0.6 MW. However, the total area under the curve for the design with a noseZ of -2 m is larger than the area under the curve for a design with a noseZ of +2.65 m. The contour plot also shows this same information, but also shows the band of highest powers following a curved shape between the two optimal designs. This curve is likely based on the default design but shows what designs produce the highest power at which wave periods.

These optimization studies are useful to show the difference in power output for different designs over different regular wave periods, but an optimization for a realistic sea state with many superimposed wave periods would involve breaking the assumptions of this analysis and likely increase computational time significantly.

noseR

The results of the nose radius (noseR) optimization using the COBYLA optimization algorithm with an initial value of 2.01 m are shown in Figure 17 and Figure 18.

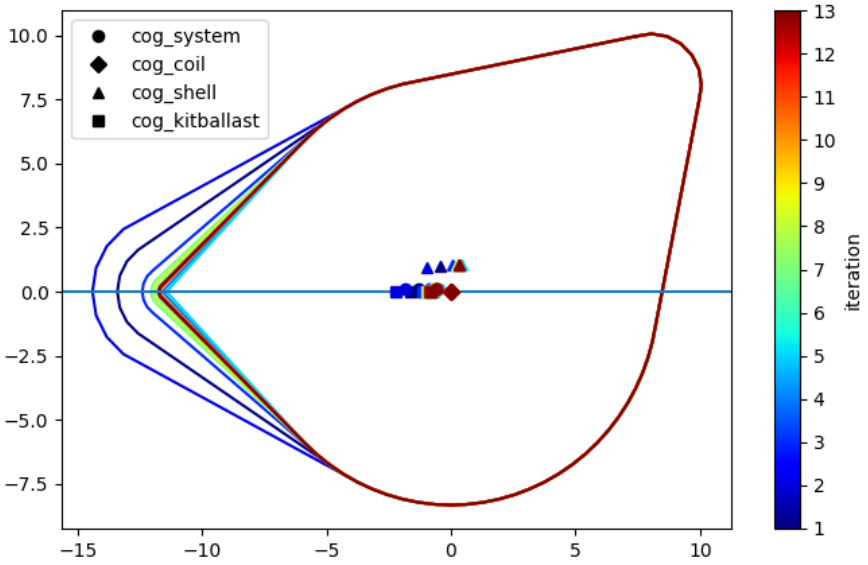

Figure 17: Iterated geometries for the noseR optimization

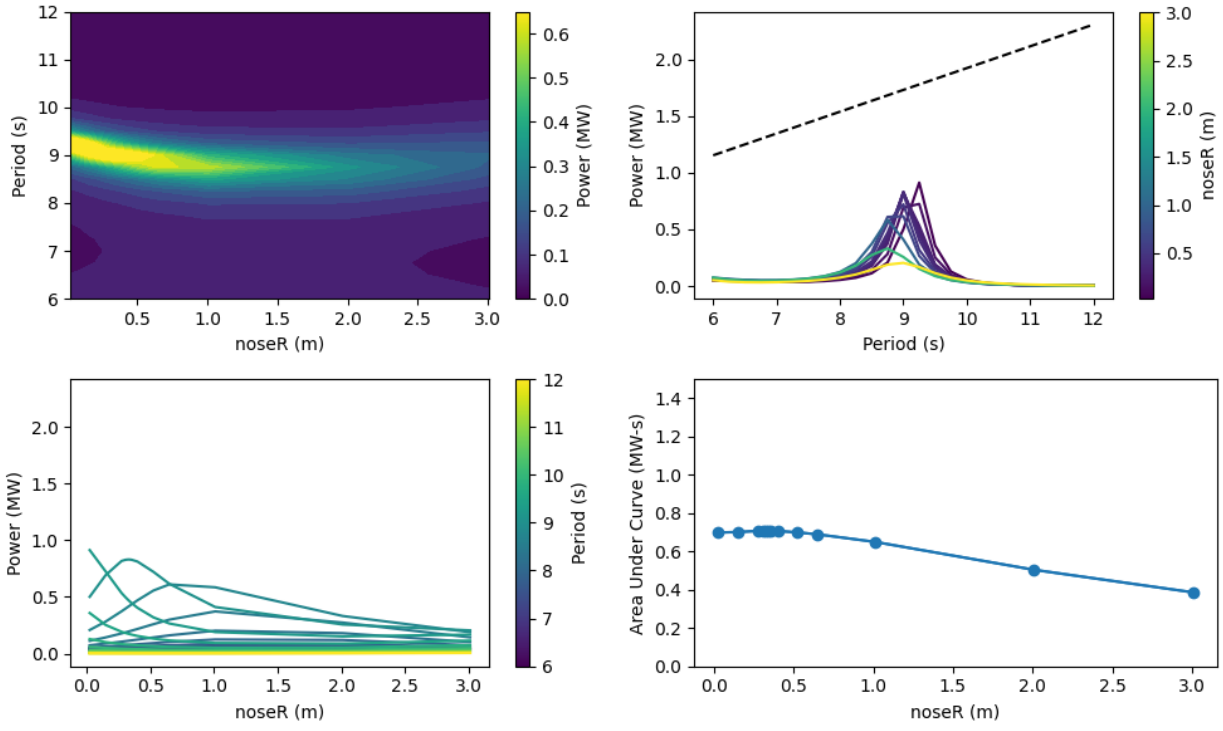

Figure 18: Power plots for noseR optimization

Using this wave period discretization and the default design parameters, it can be found that the optimal noseR is 0.4 m. Unintuitively, smaller nose radii produce more relative pitch motion than larger nose radii. The main changes in design are the steepness of the hull shape between the nose and the bottom of the bottom circle, and the roundness of the end of the nose. These changes are likely more advantageous to produce higher power. Again, the peak power outputs happen around the 9 s regular wave and have a clear high power band in the contour plot, but that band would likely move for a different default design. However, the trend of smaller noseR values producing higher powers would likely remain constant.

A parameter sweep was performed of the noseR design variable to verify the optimal value of 0.4 m, shown in Figure 19.

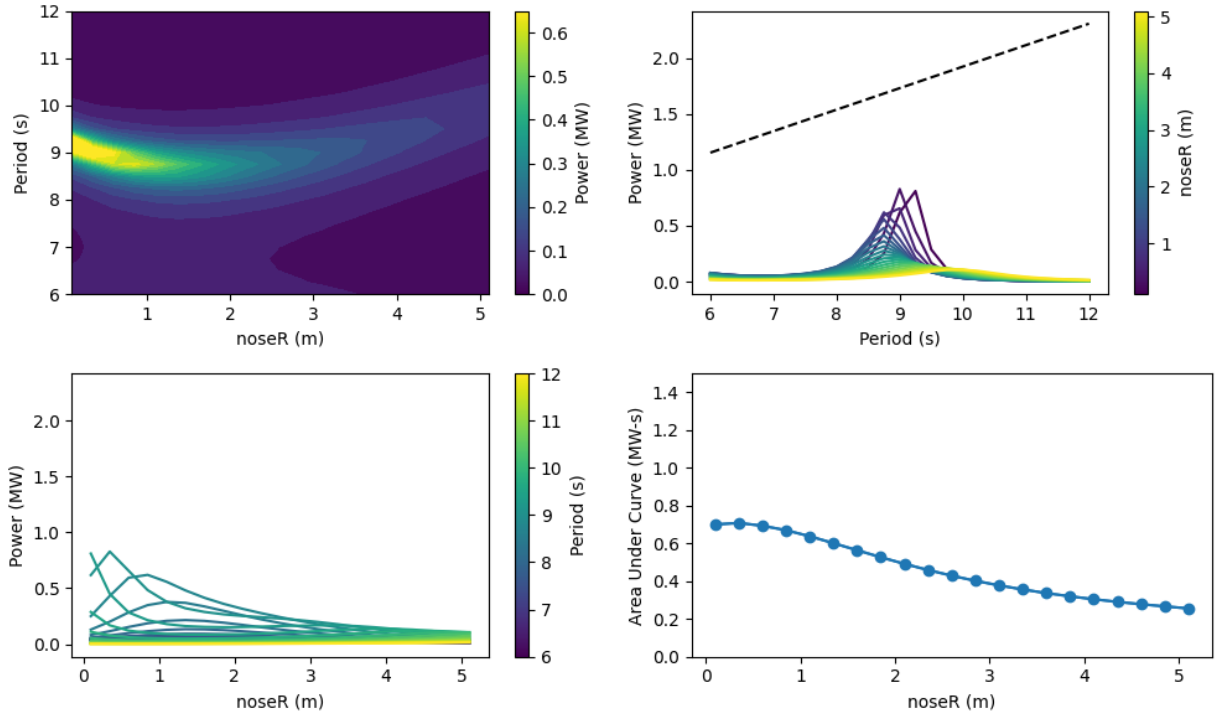

Figure 19: Power plots for noseR parameter sweep

No major differences were found in the noseR parameter sweep and the smaller noseR values still produce larger areas under the curve than larger noseR values. There still is a peak in area under the curve at about 0.4 m, where designs with a noseR value less than 0.4 m have slightly lower areas under the curve. This could be due to the wave period discretization while calculating area under the curve, the change in hydrodynamic coefficients of really small noseR values, or the relative difference between the noseR and the bottomR.

bottomR

The results of the bottom radius (bottomR) optimization using the COBYLA optimization algorithm with an initial value of 8.325 m are shown in Figure 20 and Figure 21.

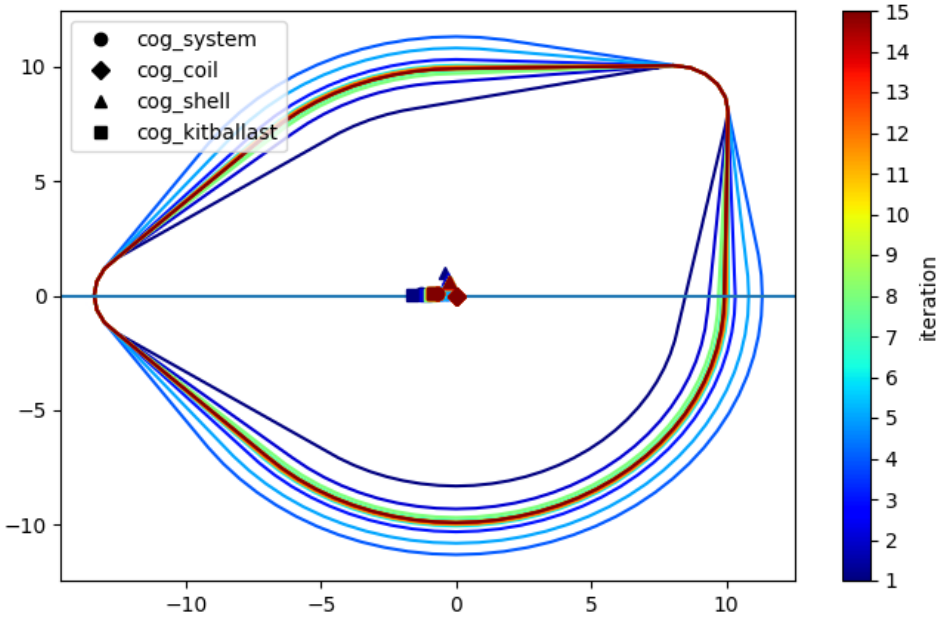

Figure 20: Iterated geometries for the bottomR optimization

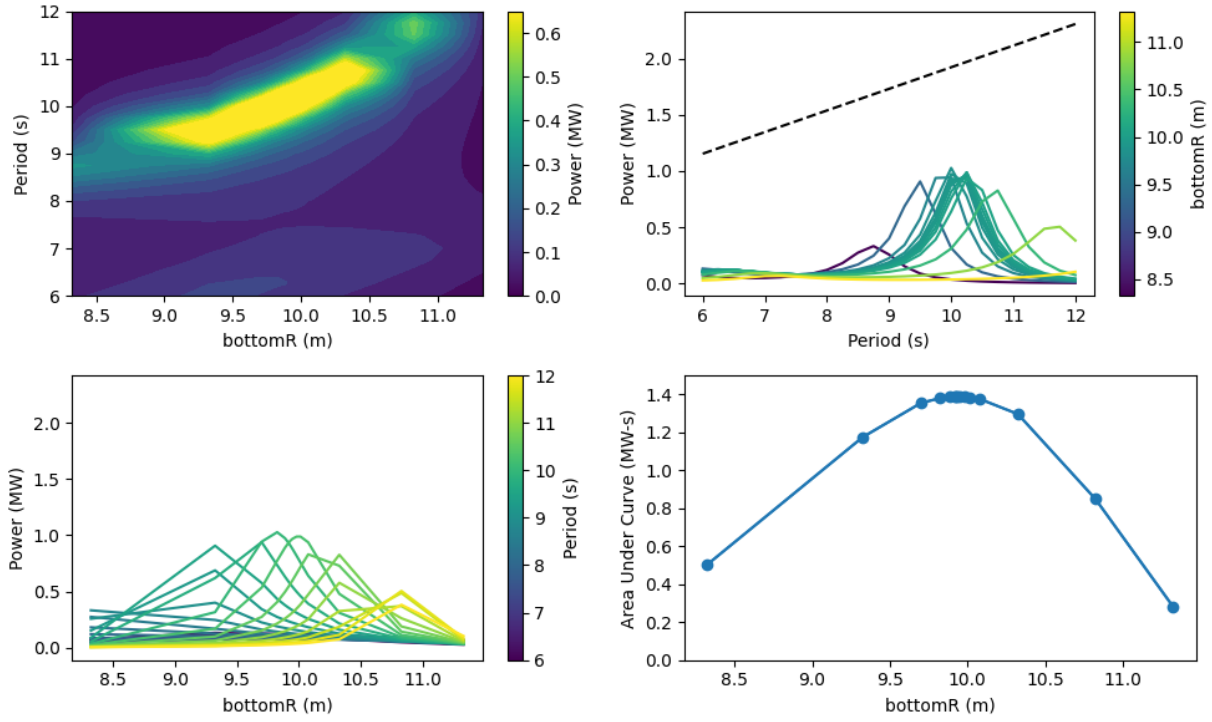

Figure 21: Power plots for bottomR optimization

Using this wave period discretization and the default design parameters, it can be found that the optimal bottomR is 9.96 m. There is a peak in the area under the curve for bottomR values ranging from 8.325-11.325 m, contrary to the assumption that larger bottomR values will increase the power output. A similar yellow band of high power again appears in the contour plot, but the band width is wider than

the contour plots for the nose variables and extends over more wave periods than the nose variables. The natural period of the device seems to significantly increase as bottomR values increase, which is likely due to the significant changes in geometry. As such, the larger bottomR values likely produce high powers in greater than 12 s waves. Therefore, for this optimization with regular wave periods between 6 and 12 s, there is an optimal bottomR value that produces the highest area under the curve.

A parameter sweep was performed of the bottomR design variable to verify the optimal value of 9.96 m, shown in Figure 22.

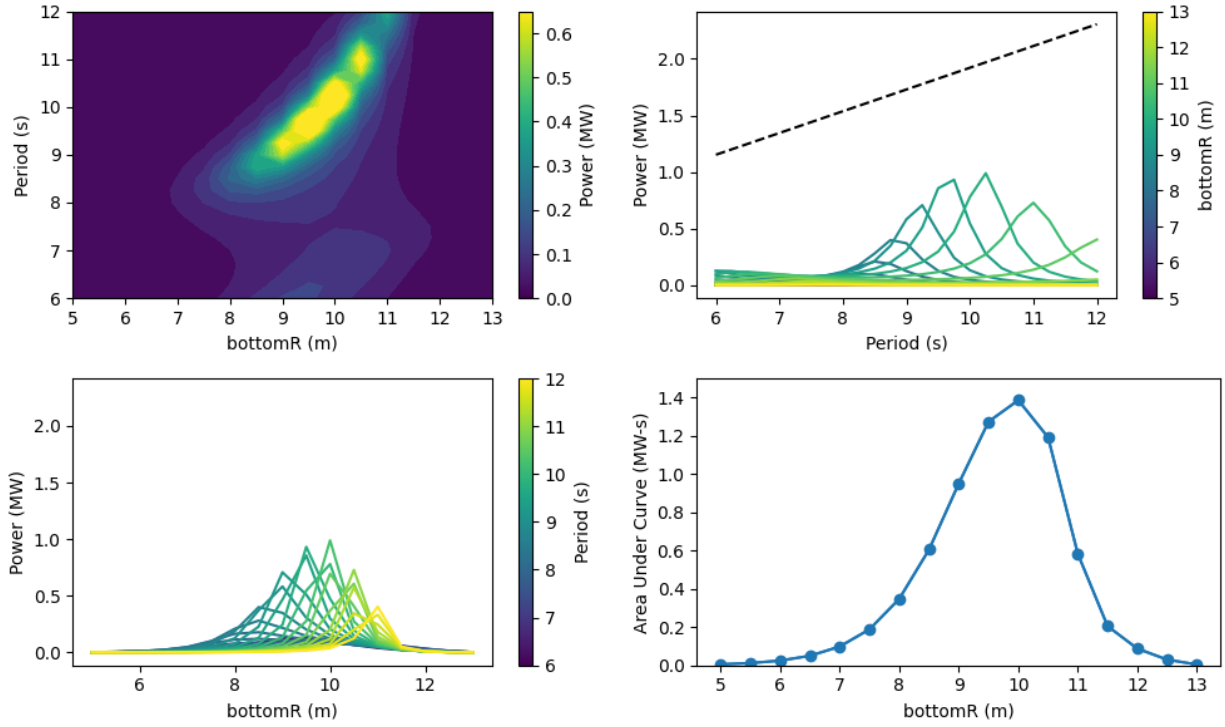

Figure 22: Power plots for bottomR parameter sweep

The same conclusions can be drawn about the bottomR variable after running the parameter sweep. The contour band is similar, the design with the highest area under the curve is around 10 m, and the only differences come from the increments of bottomR changes through the sweep and the wave period discretization. Again, it can be seen that higher power outputs for larger bottomR values can likely be found at higher wave periods, but to keep this analysis consistent, we only consider wave periods between 6 and 12 s.

sternX

The results of the stern x-position (sternX) optimization using the COBYLA optimization algorithm with an initial value of 8.05 m are shown in Figure 23 and Figure 24.

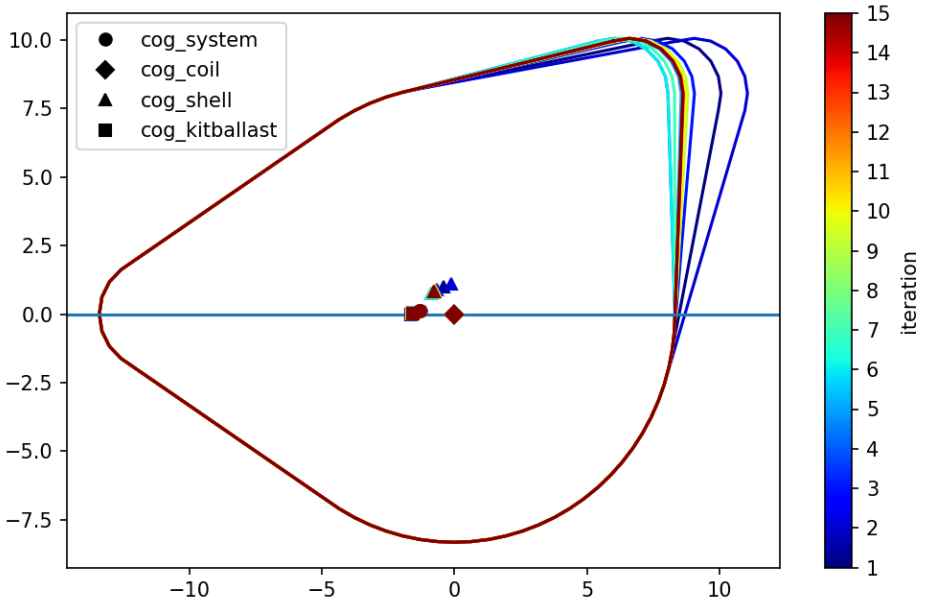

Figure 23: Iterated geometries for the sternX optimization

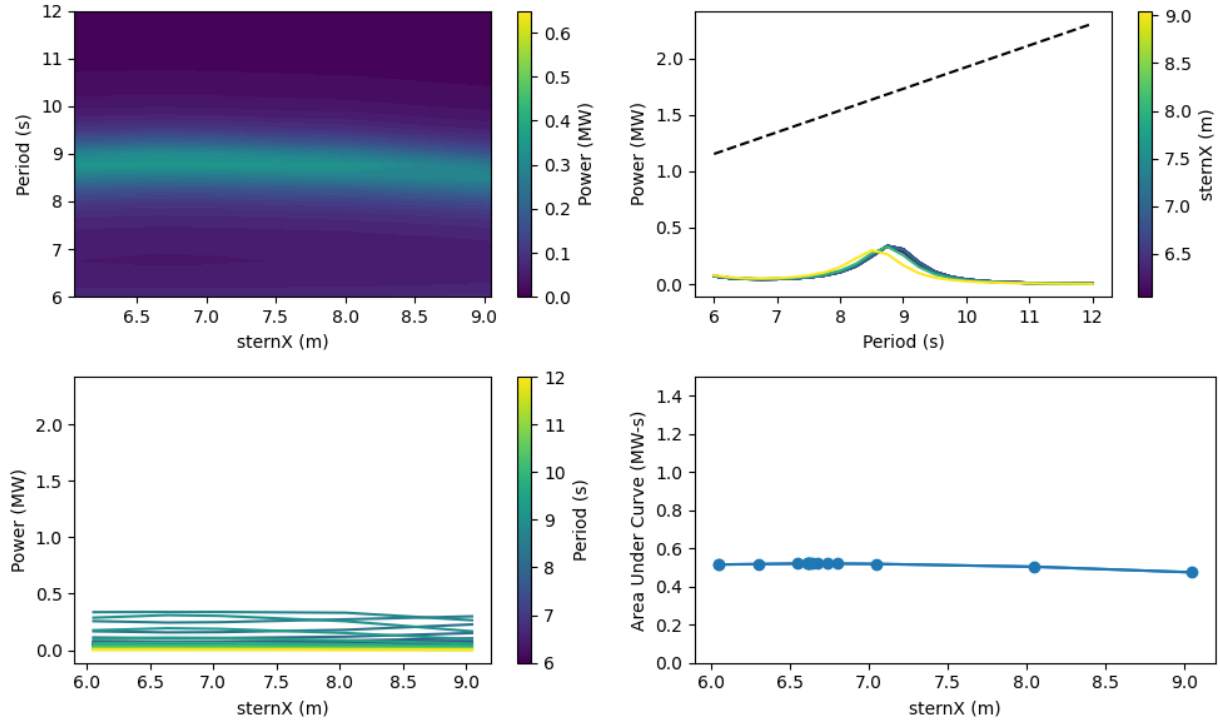

Figure 24: Power plots for sternX optimization

Using this wave period discretization and the default design parameters, it can be found that the optimal sternX is 6.62 m. The sternX is not as influential on power output as the other design variables are. All sternX designs have similar period-power curves and there is little relative difference in area under the curve between each of them.

A parameter sweep was performed of the sternX design variable to verify the optimal value of 6.62 m, shown in Figure 25.

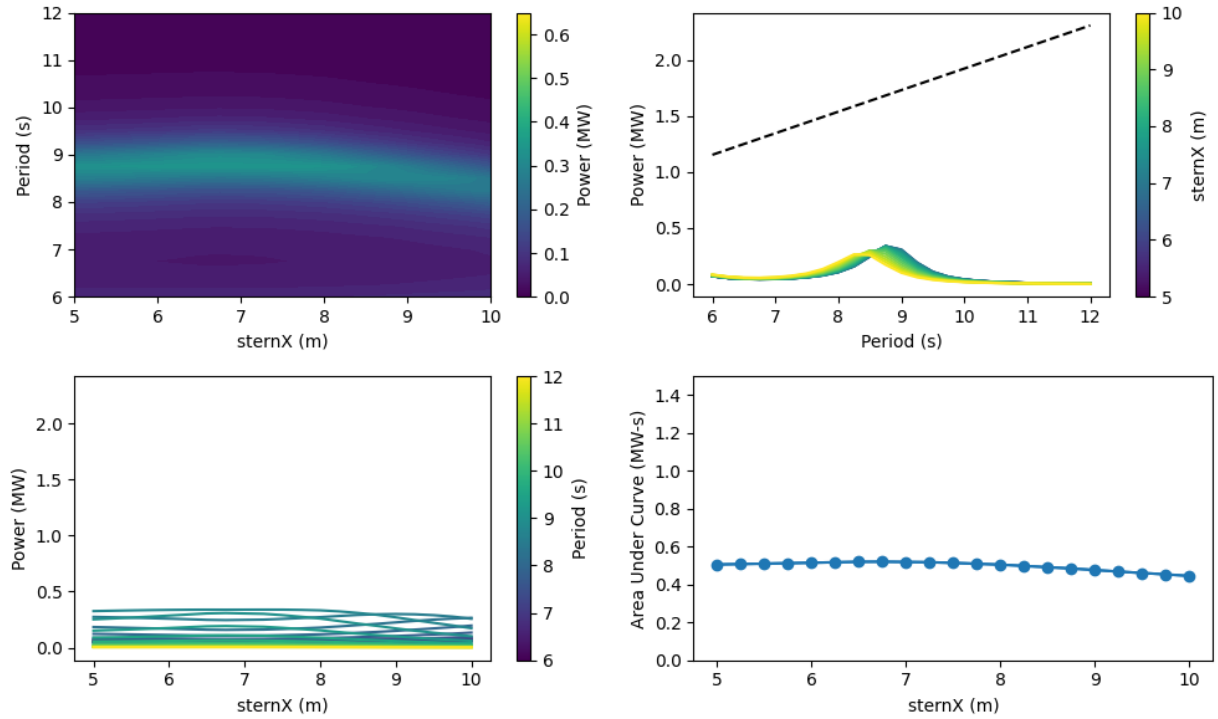

Figure 25: Power plots for sternX parameter sweep

The parameter sweep of sternX verifies the results of the optimization and shows us the relatively small effect this variable has on the performance of the PIP WEC. An optimal value can be found slightly closer to 7 m, but still has almost a negligible difference between areas under the curve for other designs. The natural periods slightly decrease as sternX increases, showing that the underwater shape of the device and the mass distribution properties do not change significantly.

sternZ

The results of the stern z-position (sternZ) optimization using the COBYLA optimization algorithm with an initial value of 8.05 m are shown in Figure 26 and Figure 27.

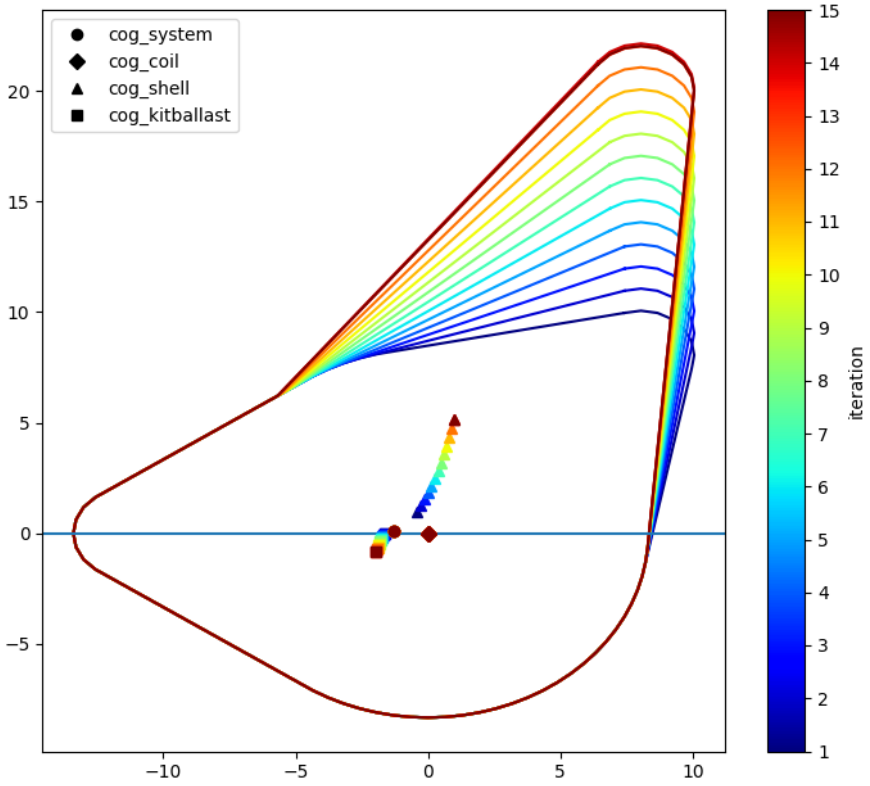

Figure 26: Iterated geometries for the sternZ optimization

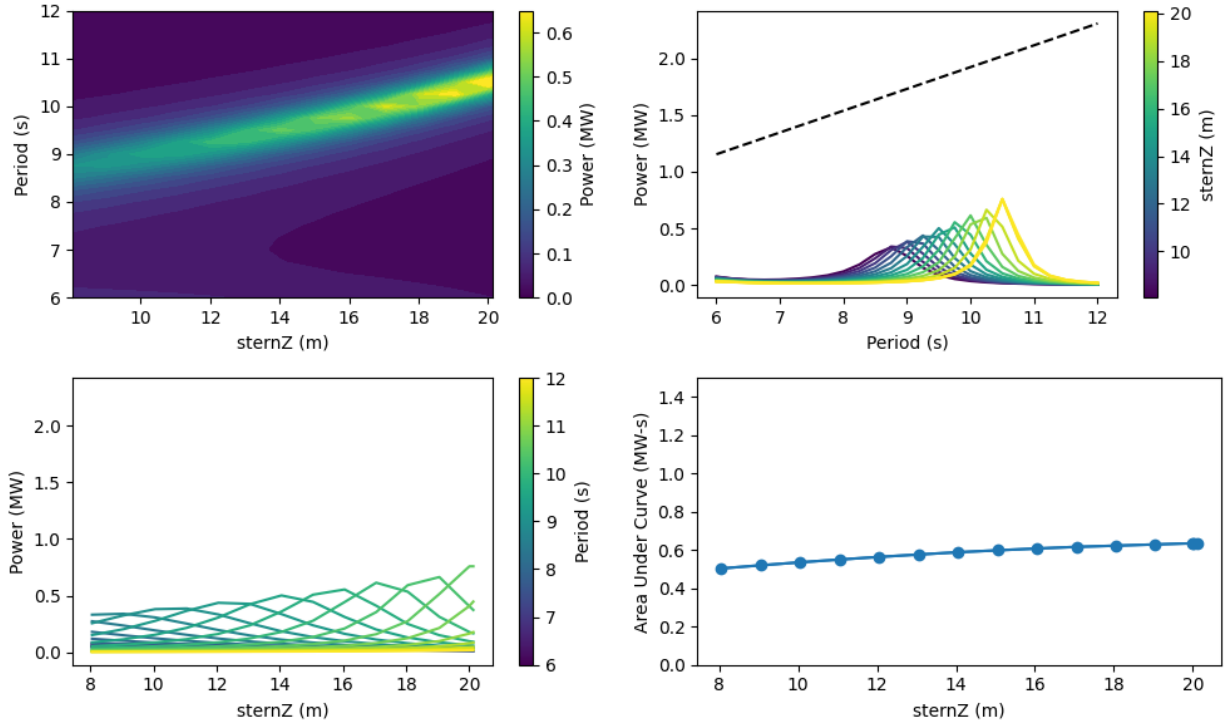

Figure 27: Power plots for sternZ optimization

Using this wave period discretization and the default design parameters, the area under the curve increases as noseZ increases, indefinitely. There are negligible changes in the underwater geometry of the device, meaning that the hydrodynamic coefficients and effects between different designs do not change. However, the change in centers of gravity of different components of the WEC change significantly as sternZ increases. Figure 28 shows how these changes in centers of gravity affect the remaining mass distribution of the device.

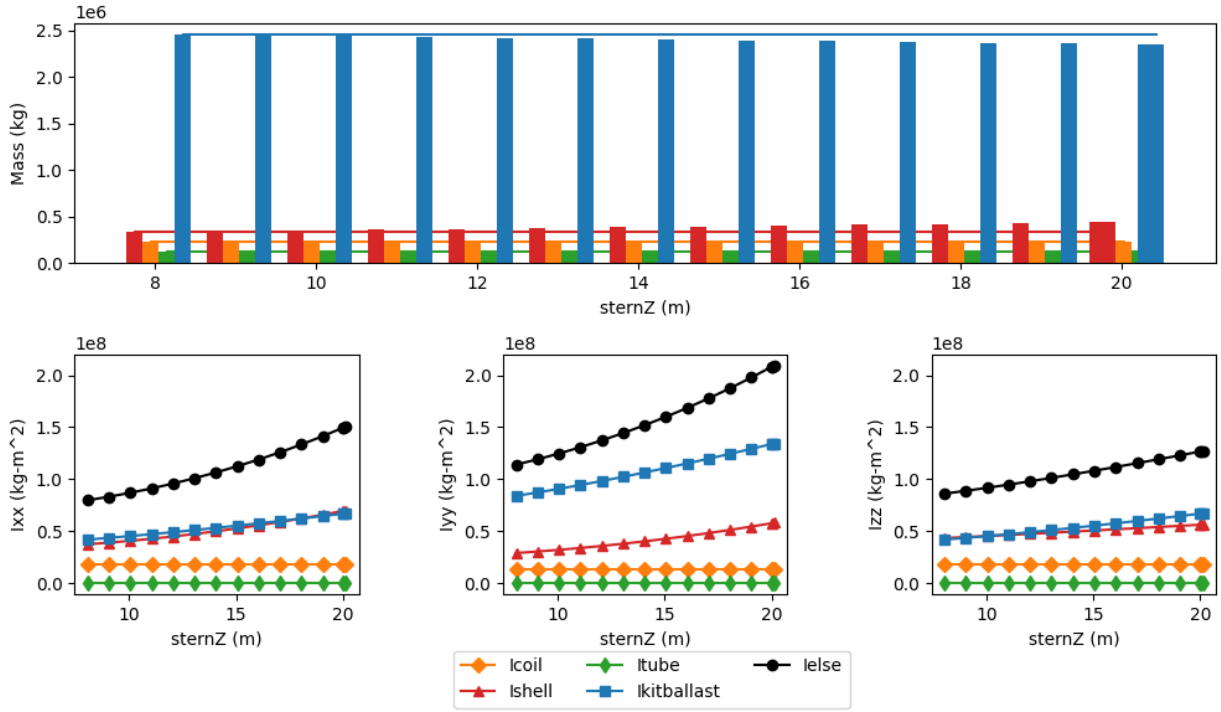

Figure 28: Mass distribution plots for sternZ parameter sweep

Blue colors represent the “kit-and-ballast”, red colors represent the outer shell, orange colors represent the internal water coil, green colors represent the internal tube, and black colors represent the “else” component of the device, which is everything but the internal water coil. As the sternZ increases, the overall shape of the hull increases, which increases the shell mass and decreases the kit-and-ballast mass to keep the device at mass equilibrium. Regardless, the moments of inertia for both the shell and the kit-and-ballast steadily increase as sternZ increases. The moment of inertia of “else”, which is the moment of inertia of everything except for the internal water coil, more rapidly increases and can be considered the direct correlation in the increase in area under the curve for larger sternZ values. The larger the moment of inertia of the device, the larger the peak average powers of the device and the subsequent areas under the curve.

This trend can be seen in the period-power curve of Figure 27 as larger sternZ values produce larger peak power outputs and larger areas under the curve. The natural period again slightly increases and sternZ increases, and there is again a band of peak powers in the contour plot ranging from 9 to 11 s. As stated before, the default design likely influences where this band of peak powers occurs, but the trend of higher powers for higher sternZ values would likely remain constant.

A parameter sweep was not performed for the sternZ variable as it produced nearly identical figures as Figure 27.

sternR

The results of the stern radius (sternR) optimization using the COBYLA optimization algorithm with an initial value of 2.01 m are shown in Figure 29 and Figure 30.

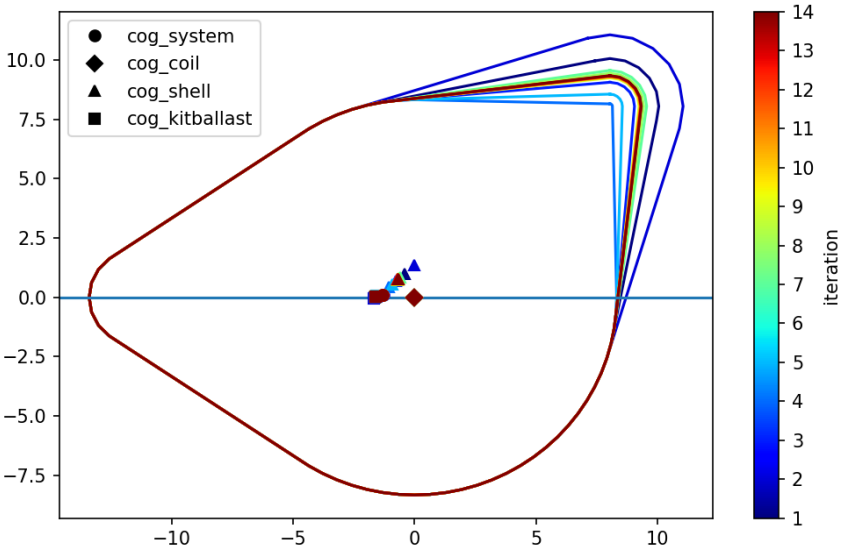

Figure 29: Iterated geometries for the sternR optimization

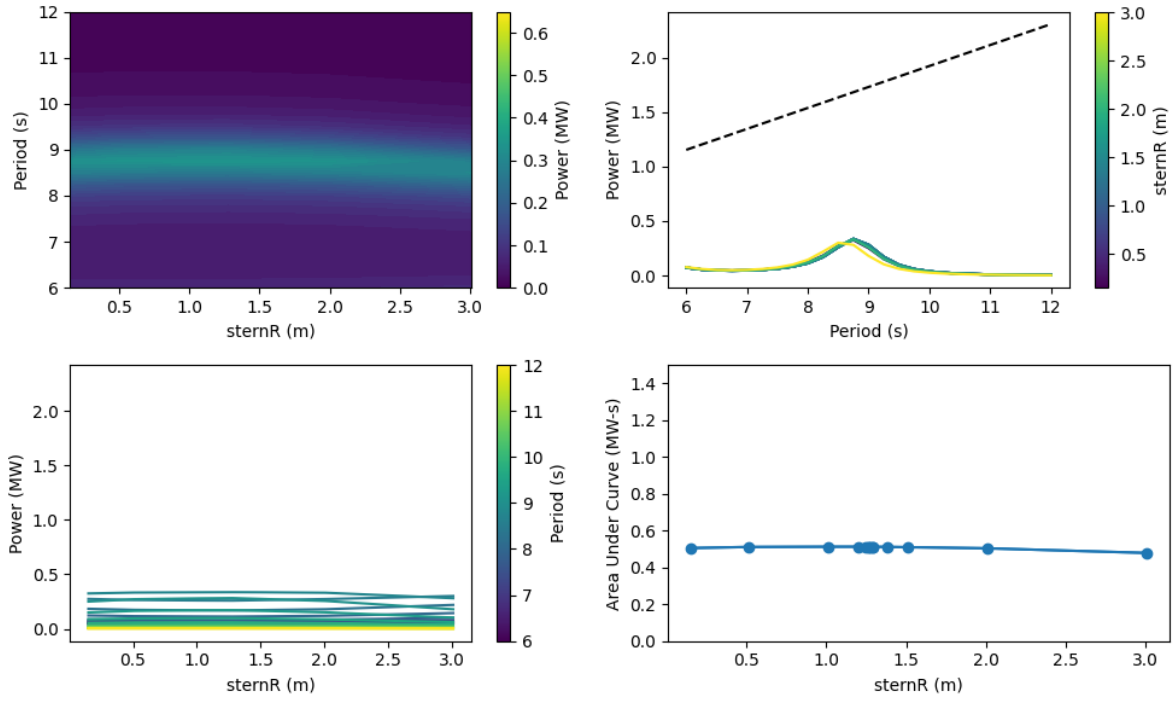

Figure 30: Power plots for sternR optimization

Using this wave period discretization and the default design parameters, it can be found that the optimal sternR is 1.25 m. Similar to sternX, the sternR variable is not as influential on power output as the other design variables are. All sternR designs have similar period-power curves and there is little relative difference in area under the curve between each of them. The difference between areas under the curve between different designs is small.

A parameter sweep was performed of the sternR design variable to verify the optimal value of 1.25 m, shown in Figure 31.

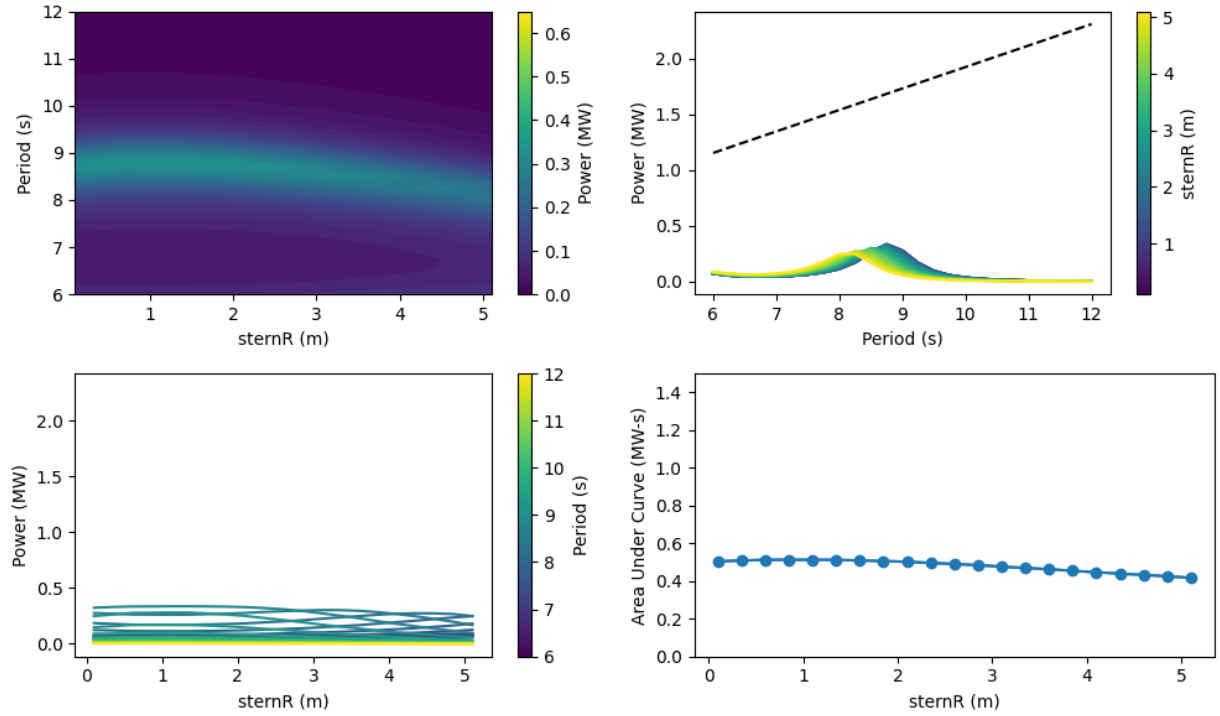

Figure 31: Power plots for sternR parameter sweep

The parameter sweep of sternR verifies the results of the optimization and shows us the relatively small effect this variable has on the performance of the PIP WEC. An optimal value can be found slightly closer to 1 m, but still has almost a negligible difference between areas under the curve for other designs. The natural periods slightly decrease as sternR increases, showing that the underwater shape of the device and the mass distribution properties do not change significantly.

Tilt

The results of the tilt (rotation of the entire device relative to the default design, positive counterclockwise) optimization using the COBYLA optimization algorithm with an initial value of 0° are shown in Figure 32 and Figure 33.

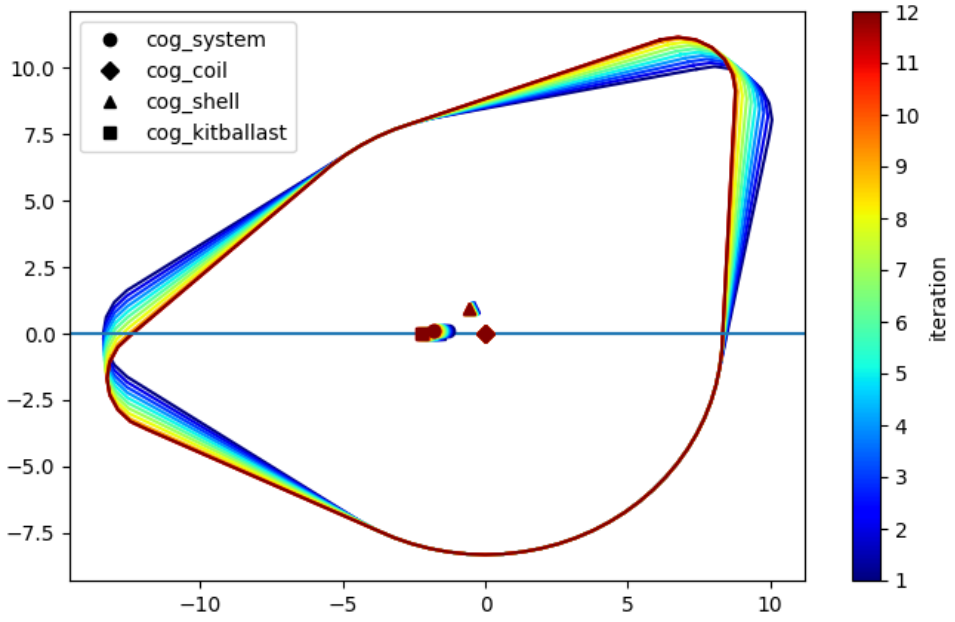

Figure 32: Iterated geometries for the tilt optimization

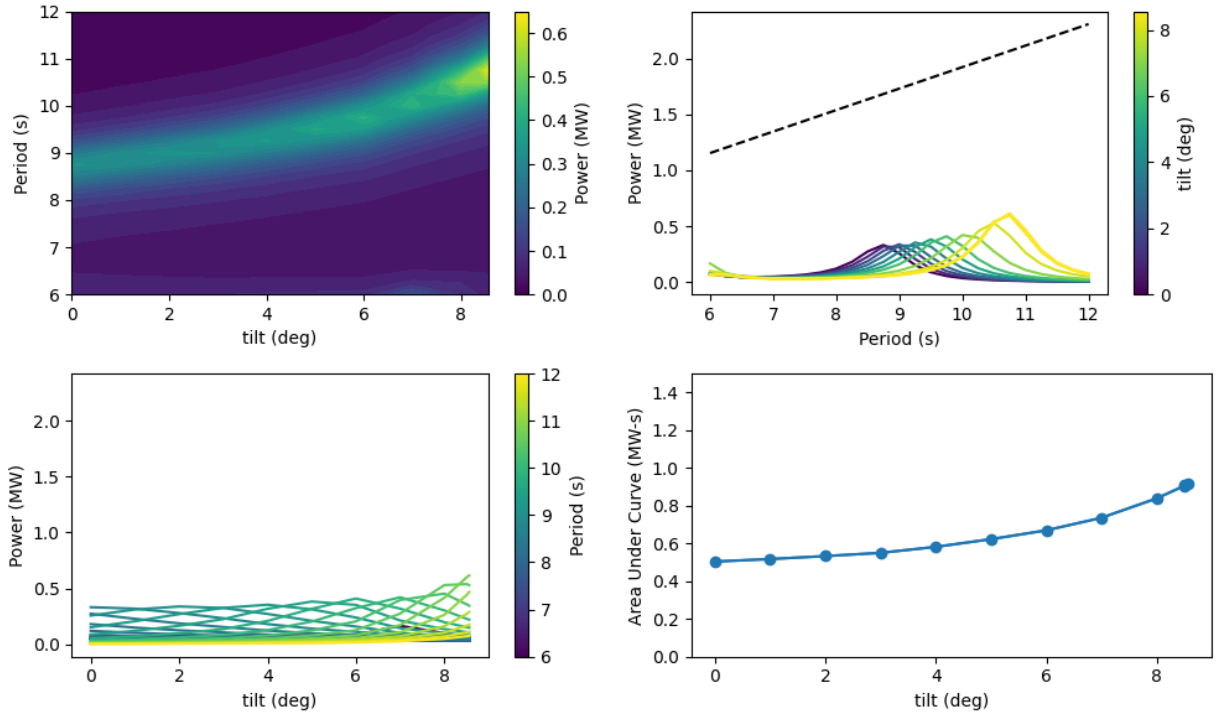

Figure 33: Power plots for tilt optimization

Using this wave period discretization and the default design parameters, the area under the curve increases as the tilt angle increases, indefinitely. This means that the “optimal” tilt angles are the ones with the nose down in the water, contrary to intuition. This is likely due to the change in underwater mesh and the hydrodynamic coefficient calculations, as the mass distribution of the device does not

change significantly for different tilt angles. The natural period of the device increases as tilt angle increases and even produces higher peak powers at larger tilt angles. These higher powers lead to higher areas under the curve. However, the difference in area under the curve for each design is relatively small, meaning that there is still significant power being produced by the lower tilt angles.

A parameter sweep was performed of the tilt design variable to explore other areas of the design space, shown in Figure 34.

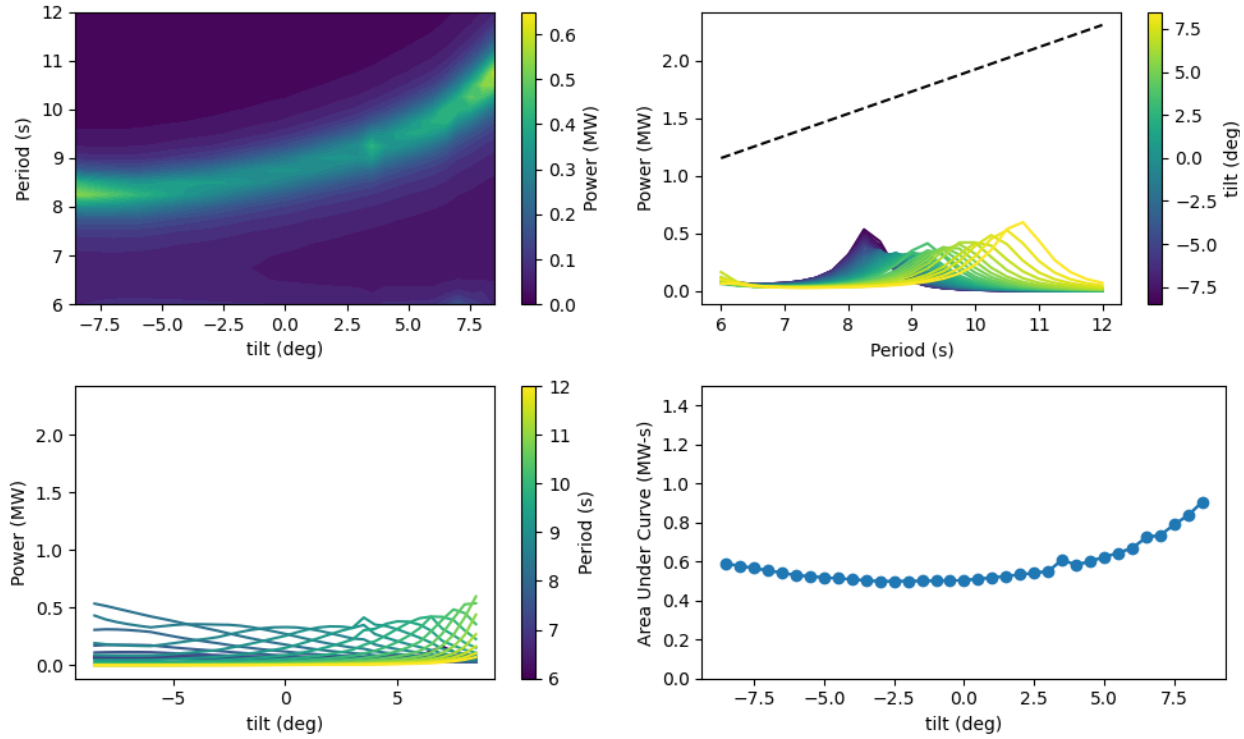

Figure 34: Power plots for tilt parameter sweep

The parameter sweep of tilt angle shows that there is a clear minimum of area under the curve at close to 2.4° clockwise, whereas larger tilt angles, clockwise or counterclockwise, produce higher areas under the curve. Furthermore, the peak power at higher clockwise angles appears to match the peak power seen in the higher counterclockwise tilt angles, but has less power at other wave periods, relatively. These results only mean that the higher tilt angles counterclockwise produce a higher area under the curve. Other optimal tilt angles can be calculated for different objectives or a different range of wave periods.

zcg_system

The results of the zcg_system (total system center of gravity in the z-direction) optimization using the COBYLA optimization algorithm with an initial value of 0.11 m are shown in Figure 35 and Figure 36.

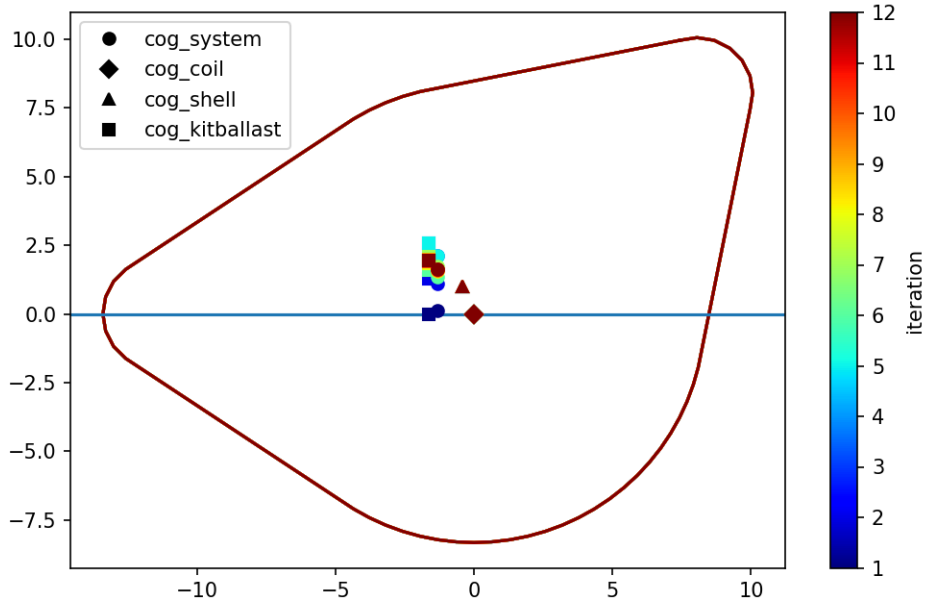

Figure 35: Iterated geometries for the zcg_system optimization

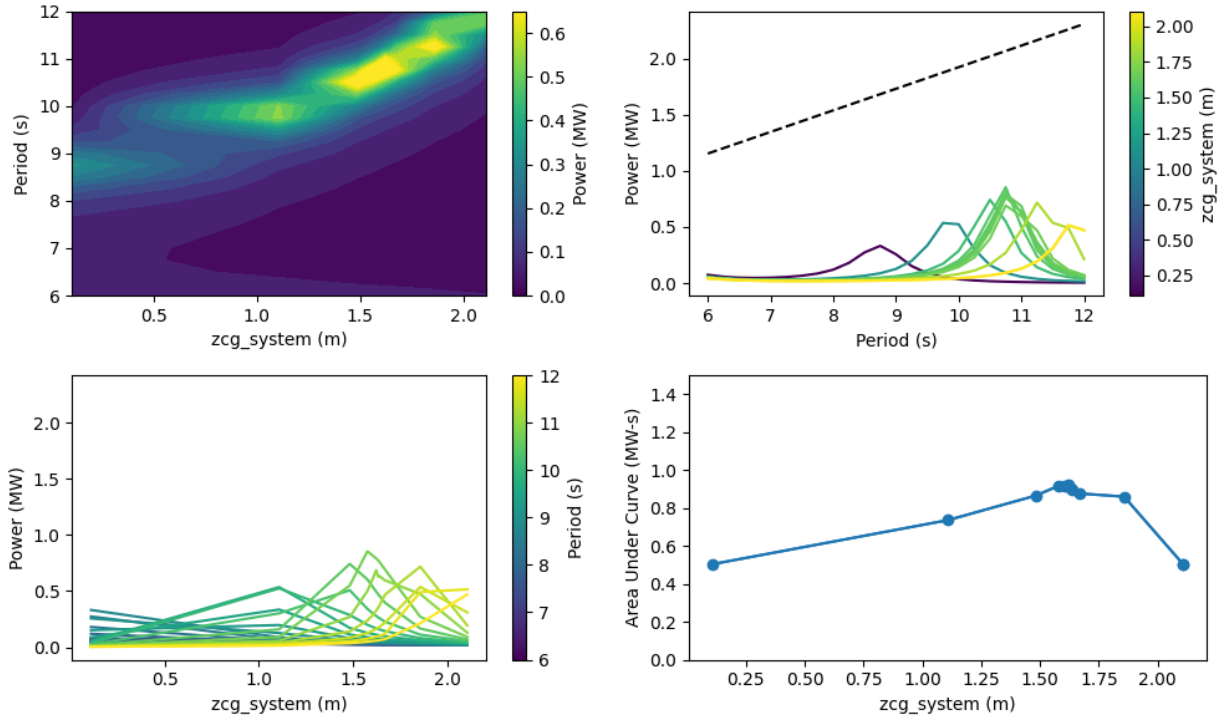

Figure 36: Power plots for zcg_system optimization

Using this wave period discretization and the default design parameters, the optimal zcg_system is 1.63 m above the waterline. It does appear that the natural period of the device changes significantly for small changes in zcg_system, shown by the period-power curve, starting around 9 s for a 0 m

zcg_system and moving to near 11 s for a 1.8 m zcg_system. This is also shown by the yellow band in the contour plot. There are some irregularities in the period-power plot that affect the area under the curve calculations, which do not affect the optimal result, but show the relative importance of the objective function and how it is calculated for an optimizer.

A parameter sweep was performed of the zcg_system design variable to explore other areas of the design space, shown in Figure 37. This parameter sweep was done with additional wave periods, extending up through 14 s waves, since the optimization results looked like the area under the curve calculations may have been cut off.

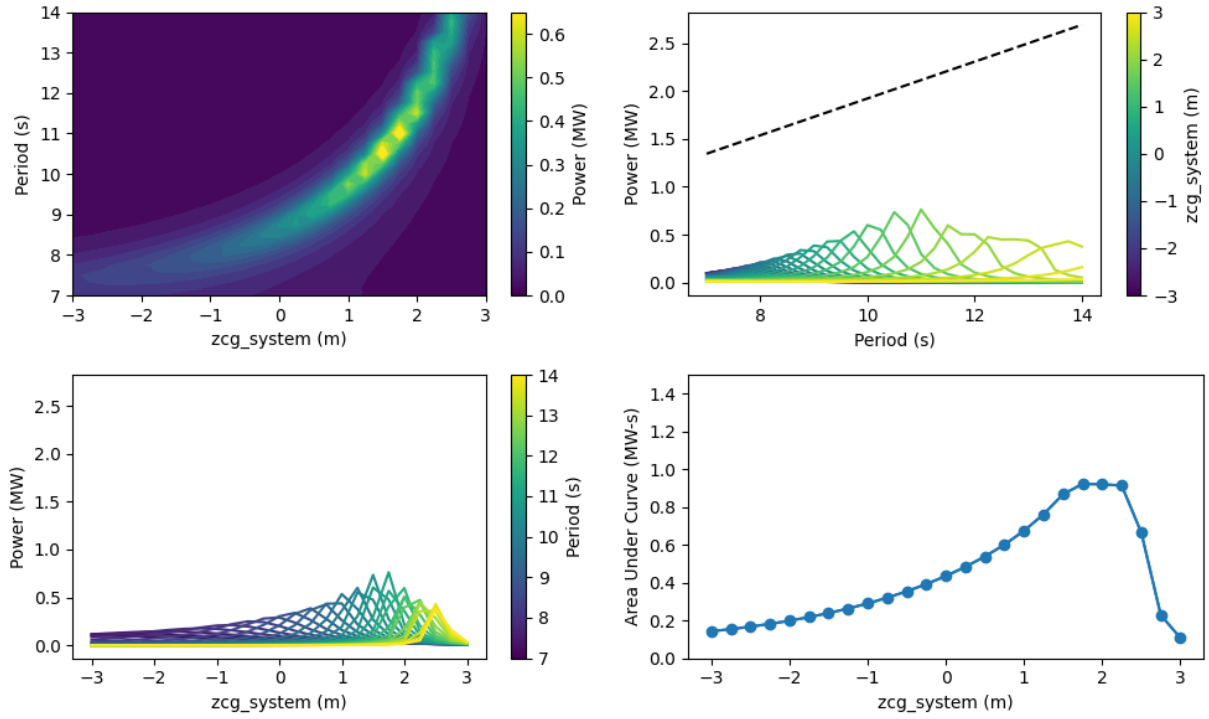

Figure 37: Power plots for zcg_system parameter sweep

The parameter sweep of zcg_system more clearly shows how the increase in zcg_system increases the peak average power, as well as the area under the curve. In this parameter sweep, we included wave periods up through 14 s and we can clearly see how the "optimal" zcg_system changed, which produced the highest area under the curve at a value of about 2 m, as opposed to the original 1.63 m. It can likely be concluded that maximizing the zcg_system maximizes the area under the curve, but not necessarily the peak power, since it appears that the peak powers start to decrease at higher zcg_system values and higher wave periods. The transverse and longitudinal metacentric heights for these runs all remained positive, ensuring the stability of the device.

Dtube

The diameter of the tube (Dtube) was another notable design variable that we investigated. In the interest of time, an optimization was not performed for Dtube, while only a parameter sweep was

performed to understand how different Dtube amounts affect the power output of the device, shown in Figure 38.

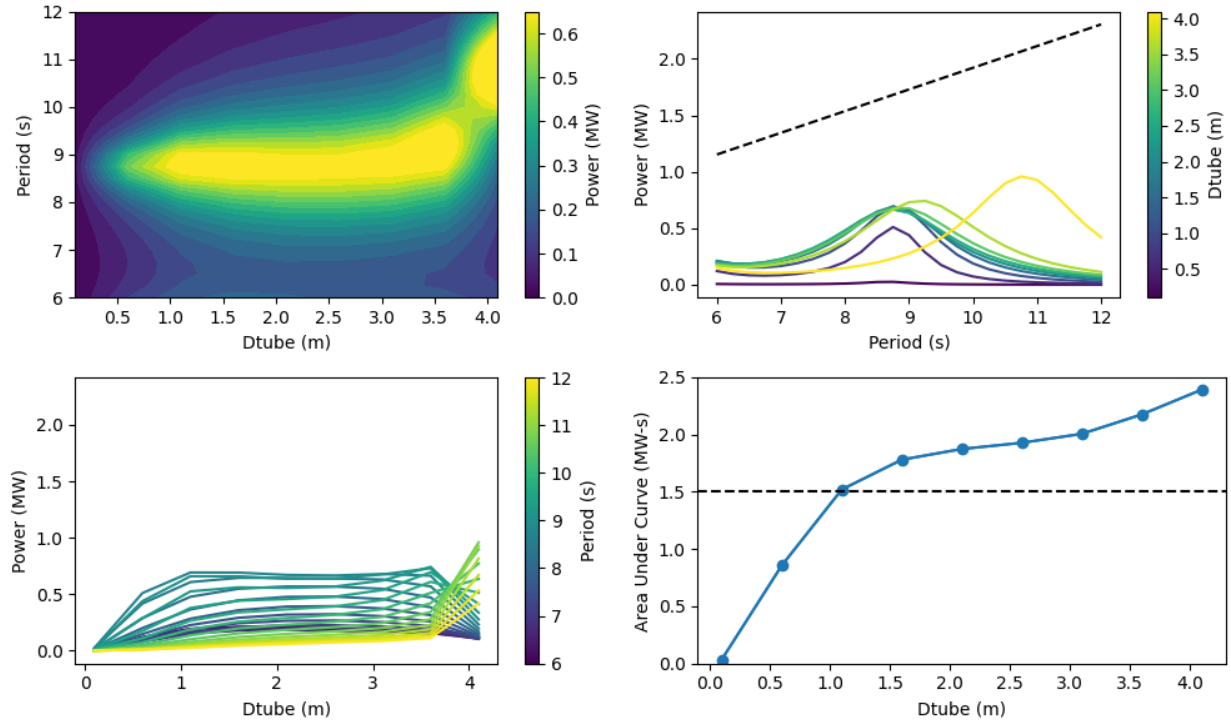

Figure 38: Power plots for Dtube parameter sweep

Compared to the previous geometric variables, Dtube has a significantly larger influence on the power output of the design. The maximum area under the curve that was seen in the previous optimizations and parameter sweeps was 1.5 MW-s, where a Dtube of 4 m produced nearly 2.5 MW-s of area under the curve. Interestingly, the natural period of the designs remained the same for all designs except the one with a 4-m Dtube. This may be a numerical inconsistency. Similar findings can be seen in a plot of the mass distributions (Figure 39).

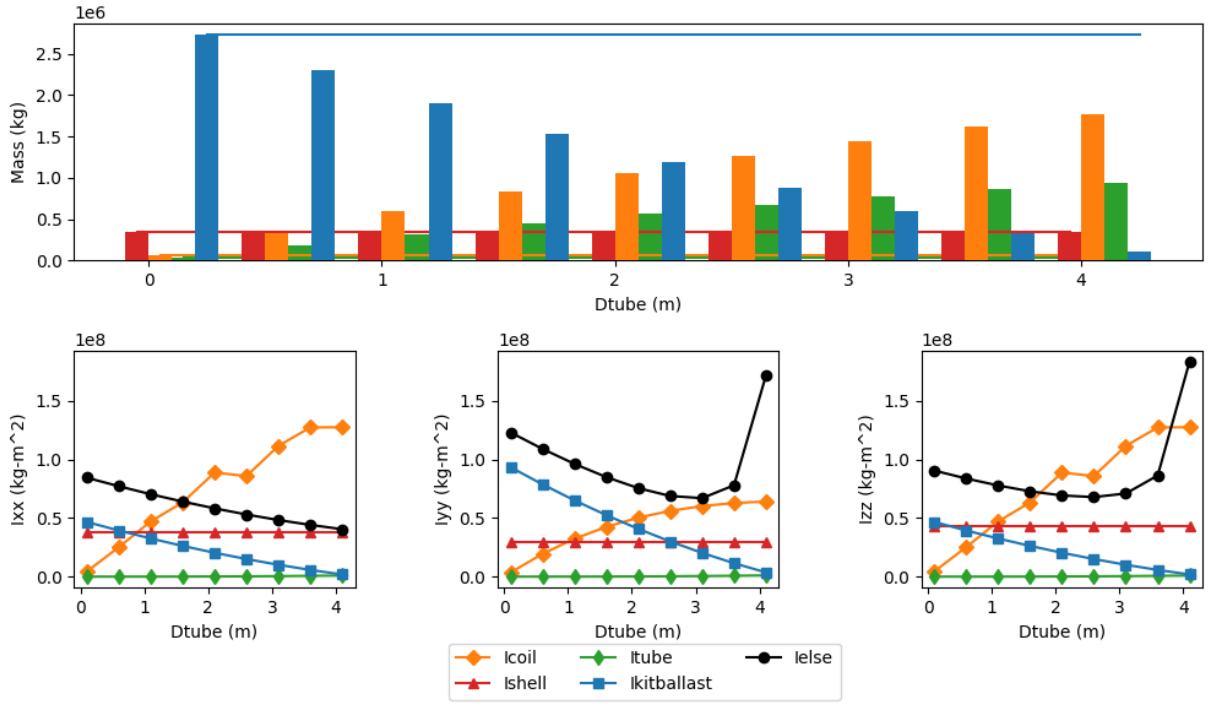

Figure 39: Mass distribution plots for Dtube parameter sweep

The increase in water coil mass automatically decreases the mass of the kit-and-ballast as well as decreases the moments of inertia of all components except for the water coil. The water coil moment of inertia is a direct input to WEC-Sim, showing the correlation between coil moment of inertia and relative pitch motion of the device.

Even without the design with a Dtube value of 4 m, it can clearly be seen that an increase in the diameter of the coil greatly increases the average power of the device and the areas under the curve.

kPTO

A parameter sweep was performed for the stiffness of the PTO system (kPTO) to determine whether this variable was influential on the power outputs. The results of the parameter sweep are shown in Figure 40.

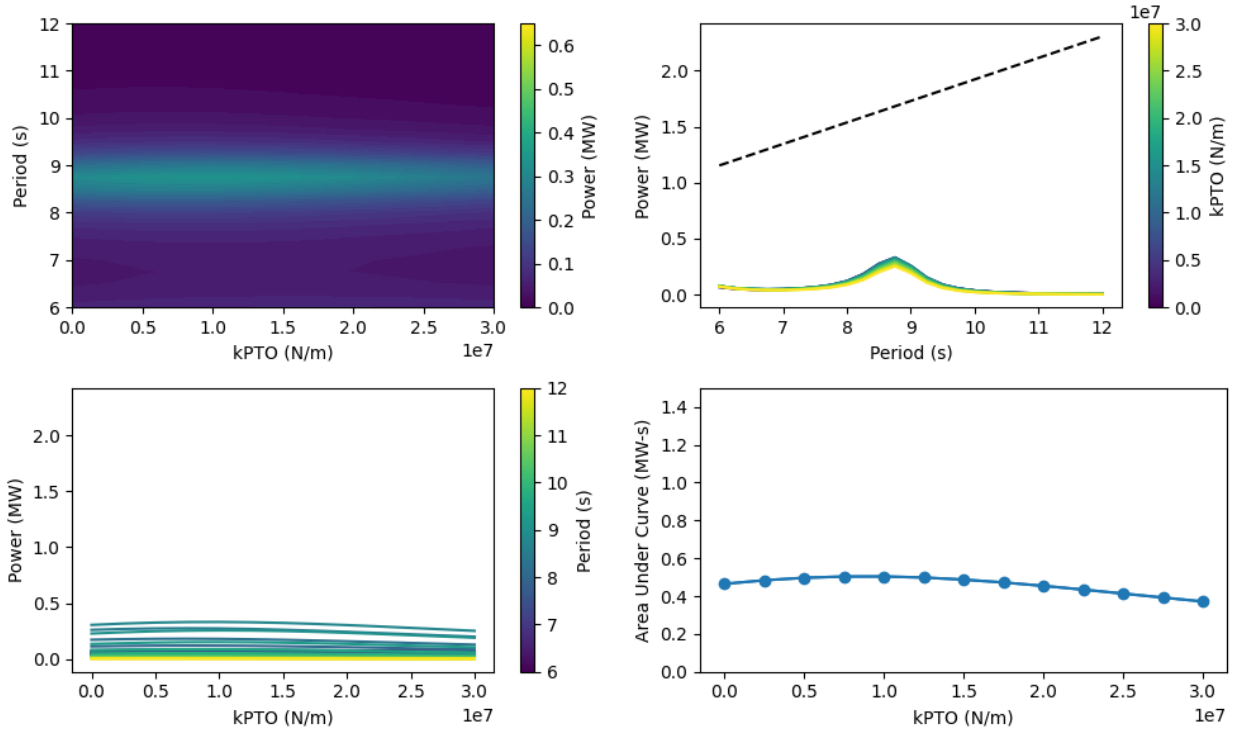

Figure 40: Power plots for kPTO parameter sweep

Unlike the diameter of the tube, the stiffness of the PTO does not seem to have an effect on the power output of the PIP WEC. Ranging from values of 0 N/m to 3e7 N/m, the area under the period-power curve stays relatively constant and no major differences are found.

cPTO

A parameter sweep was performed for the damping of the PTO system (cPTO) to determine whether this variable was influential on the power outputs. The results of the parameter sweep are shown in Figure 41.

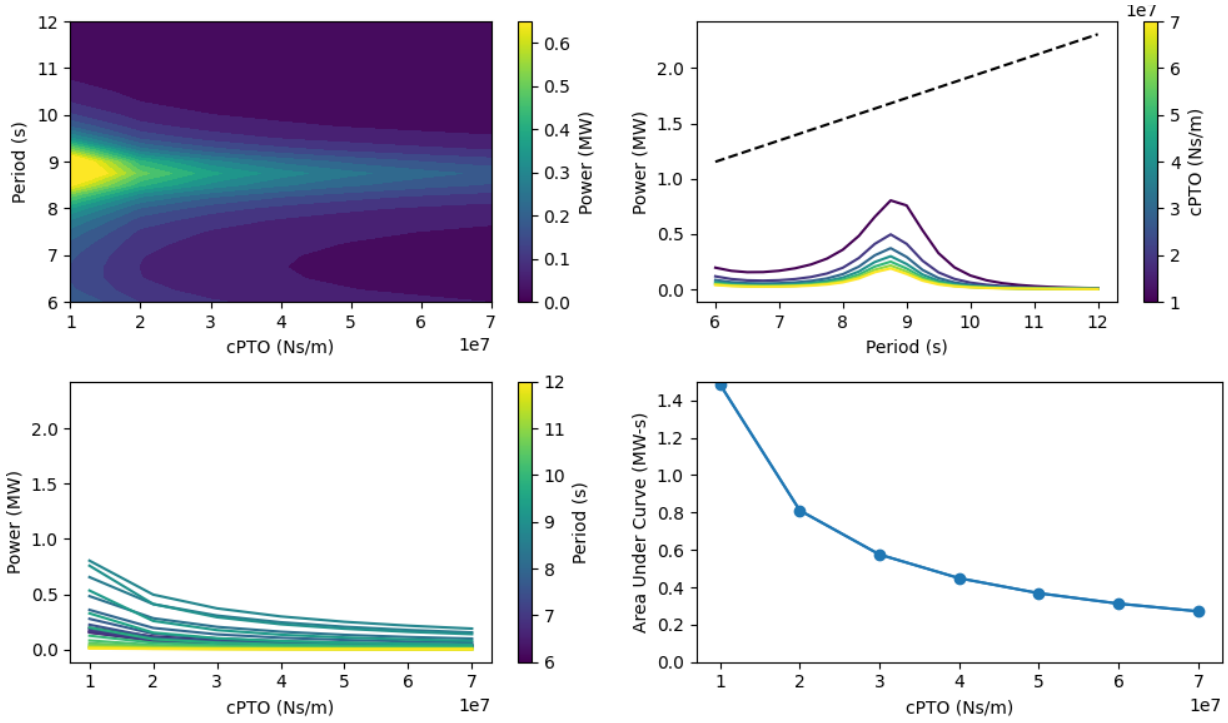

Figure 41: Power plots for cPTO parameter sweep

Contrary to the stiffness of the PTO system, the damping of the PTO system can be seen to have a significant effect on the power output and areas under the curve. Intuitively, as damping approaches zero, the relative pitch motion of the PTO model is allowed to move more freely, thus generating more power. This explains why low damping values produce higher power outputs and higher areas under the curve. For future design iterations, minimizing the PTO damping should produce higher power. An investigation into the modeling and effects of PTO damping can be done in the future.

Multi-Variable Optimization Studies

Ideally, all design variables would be used in an optimization algorithm to optimize the full design of the PIP WEC. To build up to this, we first optimized individual variables to understand which parameters were most influential to the objective. We then included multiple variables in the design optimizations, discussed in this section.

Using multiple variables in a design optimization naturally increases the number of iterations of the optimization, demonstrating the necessity of an efficient optimization method, such as COBYLA. However, these efficient optimization methods, like COBYLA, are sometimes more suited to fewer design variables than other optimizers. The following results are performed using the COBYLA optimization method, but a further investigation into other optimizers that are more suited for larger numbers of design variables might provide better results.

Each design iteration was simulated in WEC-Sim in regular waves with a wave height of 2 m and wave periods ranging from 6-12 s. Bounds were applied to each design variable to ensure feasible designs were always run in WEC-Sim. In addition to the bounds for each design variable, an equality constraint on was also applied so that the underwater volume of each PIP WEC design remained constant. The following sections detail the results from various multi-variable optimizations.

noseR, bottomR

The first multi-variable optimization uses the radius of the nose circle (noseR) and the radius of the bottom circle bottomR. This optimization is performed using the COBYLA optimizer and calculates the average power of each design at a series of wave periods ranging from 6 s to 12 s by increments of 0.25 s. The following figures showcase how the optimizer changes the radius of the nose circle and the radius of the bottom circle through progressive iterations (Figure 42). The area under the wave period-average power curve is calculated as the objective and is plotted against noseR and bottomR (Figure 43), as well as in a contour plot (Figure 44).

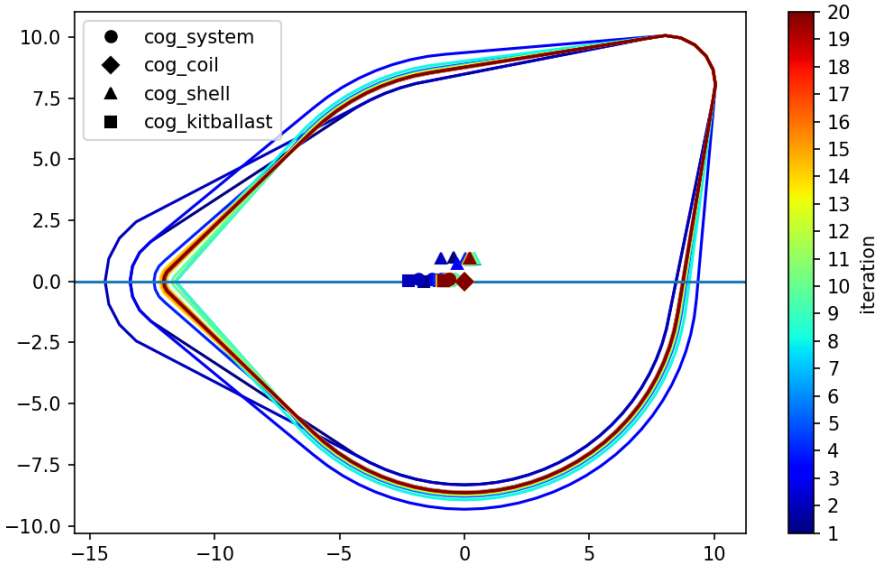

Figure 42: Iterated geometries for the noseR, bottomR optimization

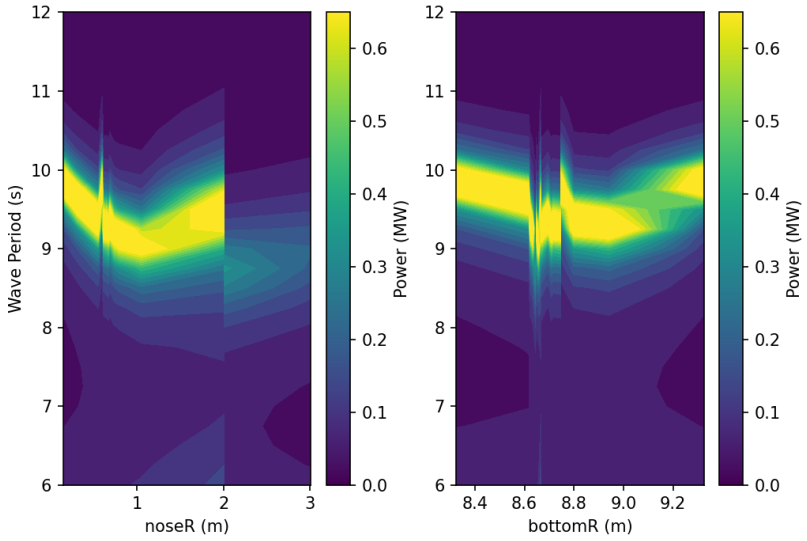

Figure 43: Power plots for the noseR, bottomR optimization

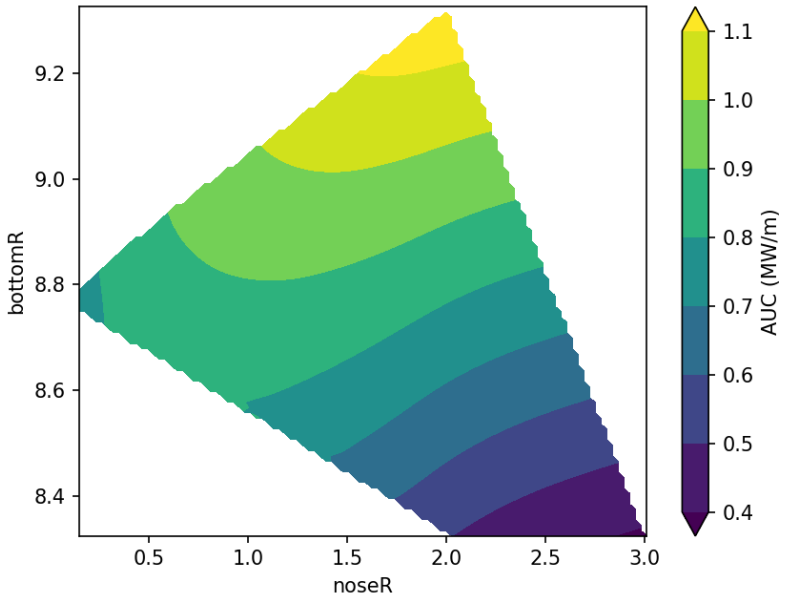

Figure 44: Area under the curve for noseR, bottomR optimization

The noseR variable is constrained between values of 0.15 and 15.0 m and the bottomR variable is constrained between values of 1.0 and 15.0 m. Data produced through the single-variable optimization informed the decision of bounding values. Additionally, the volume constraint was applied to maintain a constant underwater volume throughout the iterations. To determine the best combination of noseR and bottomR values, the area under the period-power curve was analyzed. The optimal combination of noseR and bottomR values may not produce the highest power at every wave period but will produce the highest average power across wave periods from 6 to 12 s. Based on the results, a noseR value between approximately 1.6 and 2.2 m combined with a bottomR value between approximately 9.18 and 9.3 m will result in the maximum area under the curve. However, this is purely an approximation as the

optimizer evaluates the area under the curve at specific points and interpolation between those points was conducted to produce Figure X. The optimal combination found by the optimizer was a noseR value of 0.66 m and a bottomR value of 8.65 m, which lies within the above stated region.

Based upon the above graphs, it appears that decreasing the noseR dimension from its initial length will lead to maximizing the area under the curve. Alternatively, increasing the bottomR value will ultimately lead to a greater area under the curve. Interestingly, the maximum power output of the PIP WEC occurs between 9 and 10 s waves across the range of noseR and bottomR combinations. Also, by increasing noseR, the PIP WEC will never reach the maximum power under any wave period, in contrast to bottomR, that reaches the maximum power at almost every bottomR value. This design performs poorly at wave periods above 10.5 s and below 8.5 s regardless of the noseR and bottomR values.

To verify the optimization, a "parameter sweep" (i.e., a manual optimization) is performed over incremental noseR and bottomR values to visually determine if the optimal value calculated by the optimization algorithm was correct. A similar contour plot is shown below (Figure 45). The parameter sweep was conducted from noseR values of 1 to 3 m with a step size of 0.25 m and bottomR values of 8.5 to 9.5 m with a step size of 0.2 m. To calculate the area under the curve, each combination of noseR and bottomR was simulated in WEC-Sim for wave periods from 6 to 12 s with a step size of 0.25 s. The power for each of these periods was calculated and the area under the period-power curve was found. These results indicate that noseR values between approximately 1.6 and 2.8 m combined with a bottomR value between approximately 9.35 and 9.5 m will maximize the area under the curve. Both the contour plots and the optimal combinations of noseR and bottomR values are similar for the optimization and the parameter sweep, thus verifying the optimization algorithm and successfully maximizing the area under the curve.

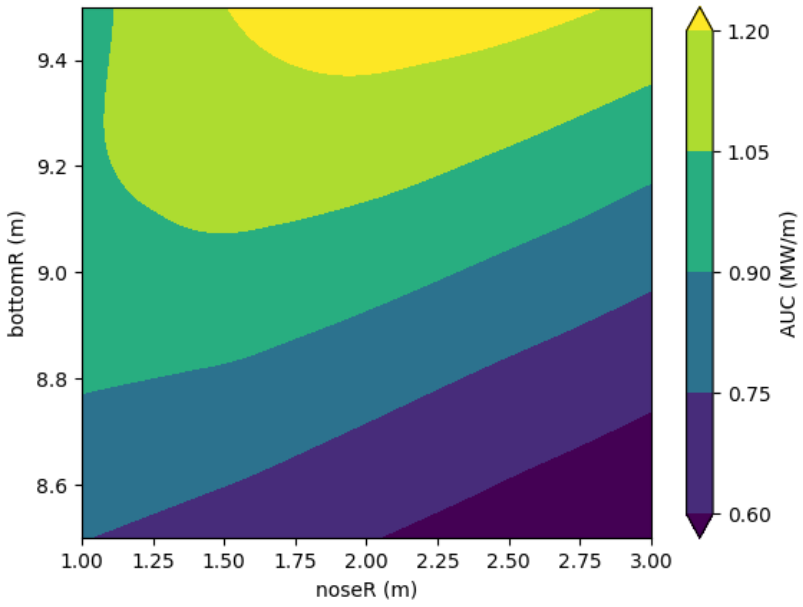

Figure 45: noseR, bottomR parameter sweep

noseX, sternZ

This optimization is between the x-position of the nose circle (noseX) and the z-position of the stern circle (sternZ). This optimization is performed using the COBYLA optimizer by calculating the average power of each design at a series of wave periods ranging from 6 s to 12 s by increments of 0.25 s. The following figures showcase how the optimizer changes noseX and sternZ through progressive iterations (Figure 46). The area under the curve results of an optimization for different configurations of noseX and sternZ parameters are plotted (Figure 47, Figure 48) as well as a parameter sweep of the same results across the range of noseX and sternZ values (Figure 49).

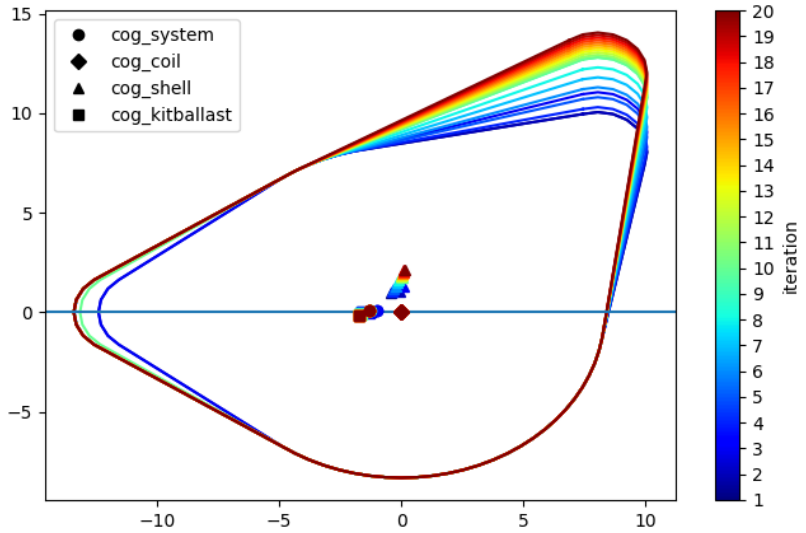

Figure 46: Iterated geometries for the noseX, sternZ optimization

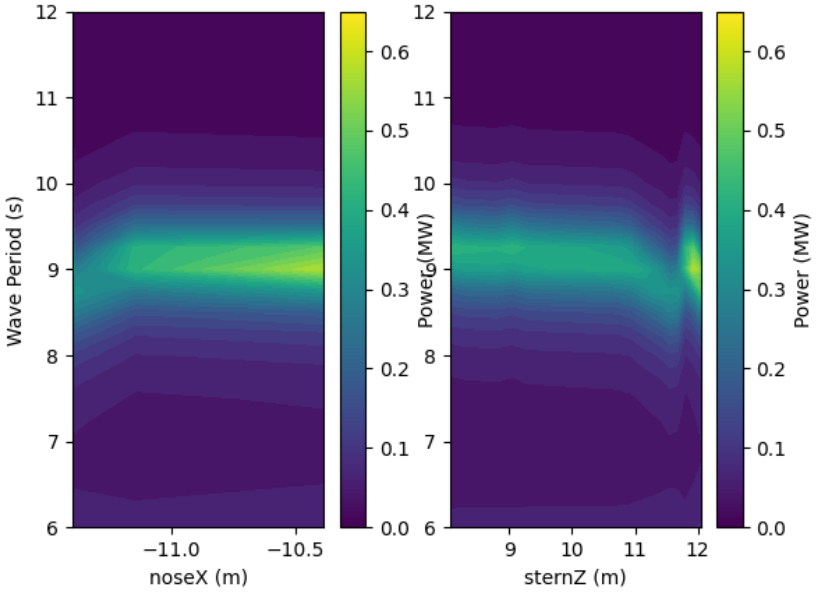

Figure 47: Power plots for the noseX, sternZ optimization

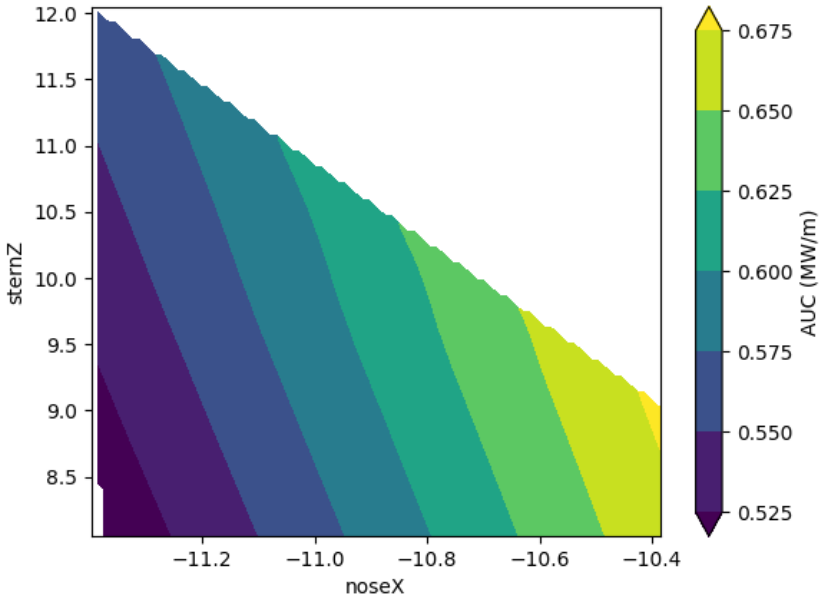

Figure 48: Area under the curve for noseX, sternZ optimization

For the noseX, sternZ optimization, bounds were applied to maintain a constant underwater volume. Additionally, the noseX dimension was constrained between -20 and -8.325 m, while the sternZ dimension was constrained between 0.2 and 15 m. Based on the contour plot, a noseX value greater than -10.4 m and a sternZ value between approximately 9.25 and 8.75 m will maximize the area under the period-power curve. The exact dimensions that the optimizer found to maximize the area under the curve was a noseX value of -10.385 m and a sternZ value of 9.05 m. In the single variable optimization studies, a "shorter" design produces greater power, which can again be seen in this multi-variable optimization. Based on the Figure 1, the sternZ value was increased to maximize the area under the curve. This can also be seen in the area under the curve plot as the sternZ value was increased from its initial value of 8.05 m to reach the optimal z-position. This relationship between a greater sternZ value and higher power produced is evident in the single variable optimization, offering some validation to these multi-variable optimization results.

In this optimization, the PIP WEC only performs well and produces considerable power at a wave period of 9 s. Additionally, very little power is produced at wave periods outside of 8 to 12 s. In the single variable optimization studies, changes to the noseX and sternZ dimensions altered the resonant period of the device. However, likely due to the small range of noseX and sternZ values used in the multi-variable optimization, this trend is not evident.

A parameter sweep of the noseX and sternZ variables was completed by sweeping through the range of noseX values from -11 m to -10 m with a step size of 0.2 and the range of sternZ values from 8 m to 10 m with a step size of 0.4. The results of this parameter sweep can be seen in the plot below. This plot shares some similarities to the optimization plot as the maximum area under the curve occurs at lower noseX values and higher sternZ values. However, the parameter sweep plot shows a "band" of maximum area under the curve values that occur at various combinations of noseX and sternZ. It is likely that increasing sternZ and decreasing noseX further outside of the range of this plot would result in areas under the curve values greater than those displayed below.

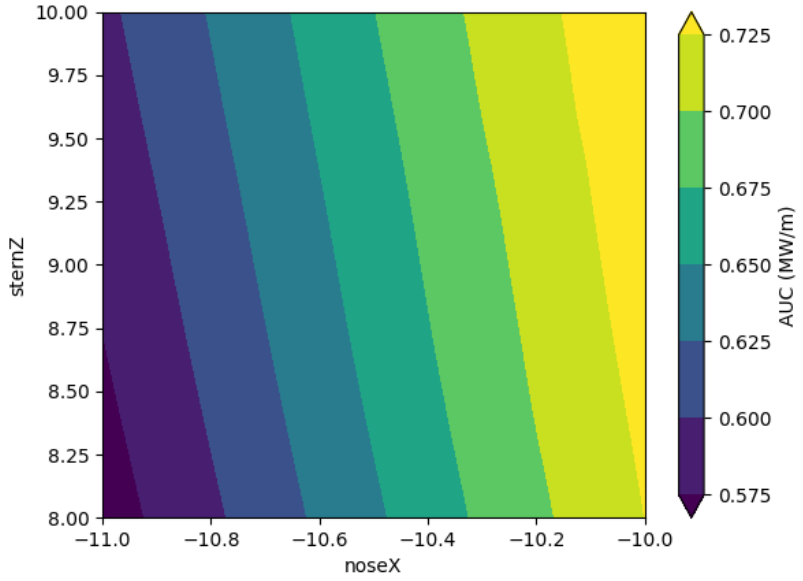

Figure 49: noseX, sternZ parameter sweep

sternX, sternZ, sternR

The first three design variable optimization is between the x-position, z-position, and radius of the stern circle (sternX, sternZ, sternR). This optimization is performed using the COBYLA optimizer by calculating the average power of each design at a series of wave periods ranging from 6 s to 12 s by increments of 0.25 s. As three design variables are being optimized instead of two, additional iterations are required. The following figures showcase how the optimizer changes these three design variables through progressive iterations (Figure 50). The effect of each design variable on the power output of the PIP WEC under different wave periods is all displayed in the figures below (Figure 51, Figure 52).

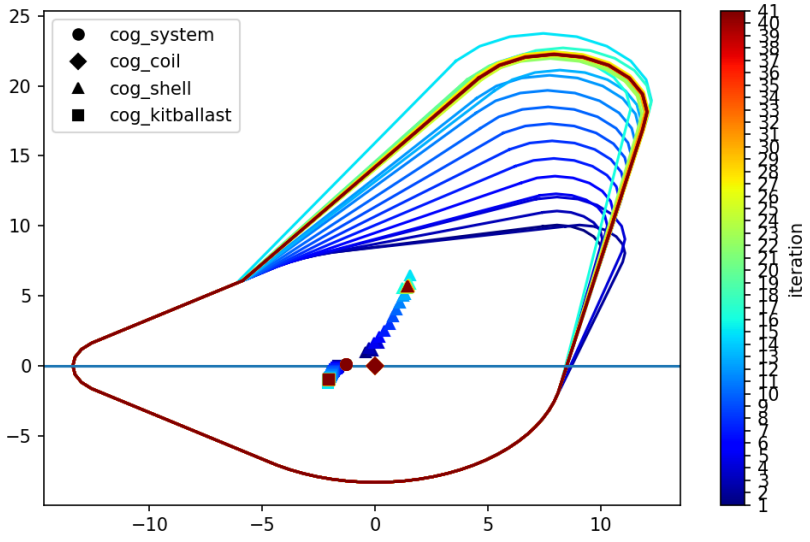

Figure 50: Iterated geometries for the sternX, sternZ, sternR optimization

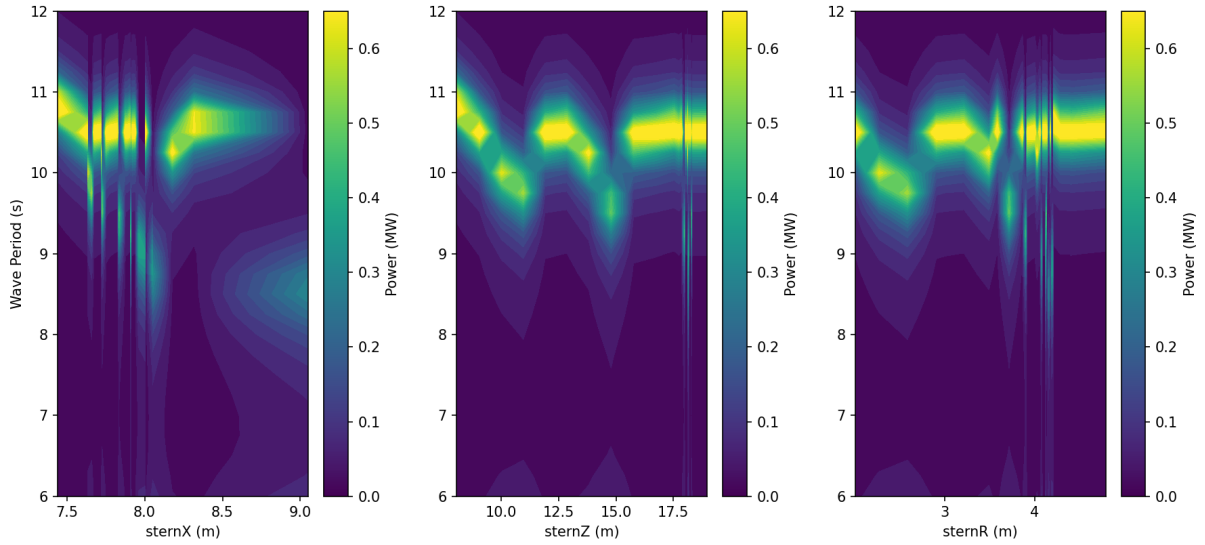

Figure 51: Power plots for the sternX, sternZ, sternR optimization

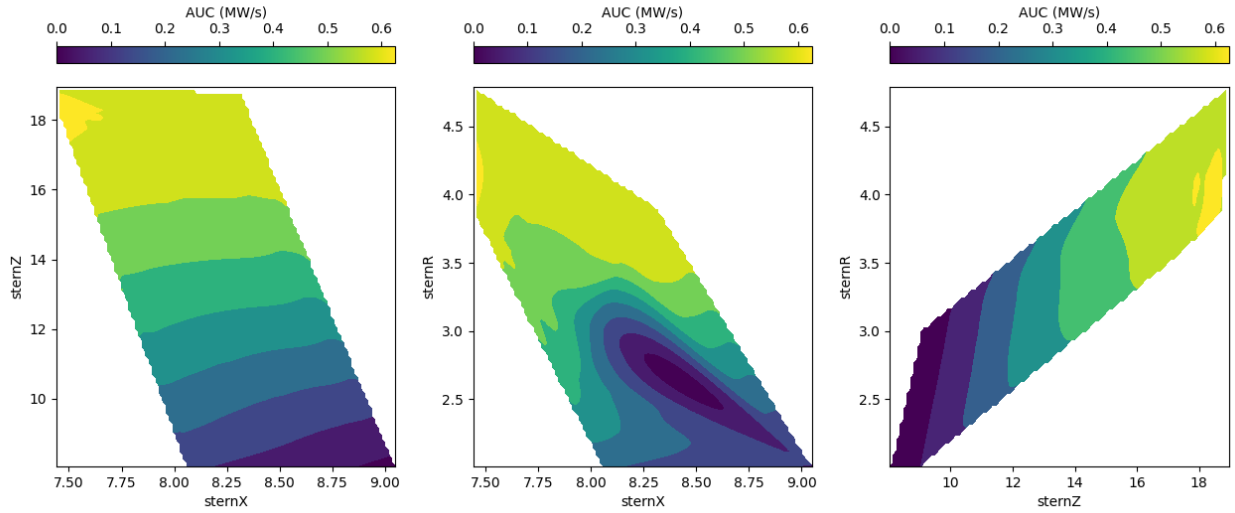

Figure 52: Area under the curve plots for the sternX, sternZ, sternR optimization

For this optimization, each of the design variables were given the same bounds between 0.1 and 20. Small dimensions lead to errors in the optimization, and bounds with large ranges did not significantly increase the time to perform the optimization. Additionally, previous results indicated that power output continuously increased for greater values of sternZ, so a significantly high upper bound value was required. A volume constraint was applied to maintain a constant underwater volume.

As seen in the cross-section plot, the optimizer altered the sternZ more severely than sternX and sternR, since the sternZ has a larger effect on the power output of the PIP WEC. Additionally, increasing the sternZ value leads to a greater area under the curve, which is consistent with the single variable study. Changing the center of gravity of the outer shell changes the center of gravity of the kit-and-ballast, which increases the moment of inertia of the WEC, which produces higher power outputs.

The optimizer increased the sternR value to maximize the area under the curve, which contrasts with the single variable optimization study where the sternR value was decreased to maximize the area under the curve. The sternX value was decreased to reach the maximum area under the curve, which also occurred in the single variable optimization.

Based on the power plot, the various designs of the PIP WEC within this optimization produced the most power within wave period from 10 to 11 s. It is once again evident that within this optimization, increasing the sternZ and sternR values and decreasing the sternX value increased the power output of the PIP WEC.

noseX, noseZ, noseR

This multi-variable optimization of three design variables is between the x-position, z-position, and radius of the nose circle (noseX, noseZ, noseR). This optimization is performed using the COBYLA optimizer by calculating the average power of each design at a series of wave periods ranging from 6 s to 12 s by increments of 0.25 s. As three design variables are being optimized instead of two, additional iterations are required. The following figures showcase how the optimizer changes these three design variables through progressive iterations (Figure 53). The effect that each design variable has on the power output of the PIP WEC under different wave periods is all displayed in the figures below (Figure 54, Figure 55).

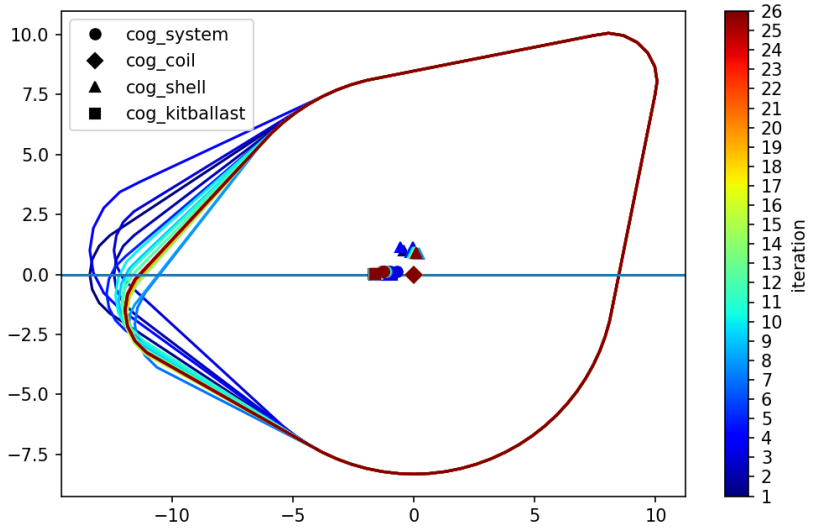

Figure 53: Iterated geometries for the noseX, noseZ, noseR optimization

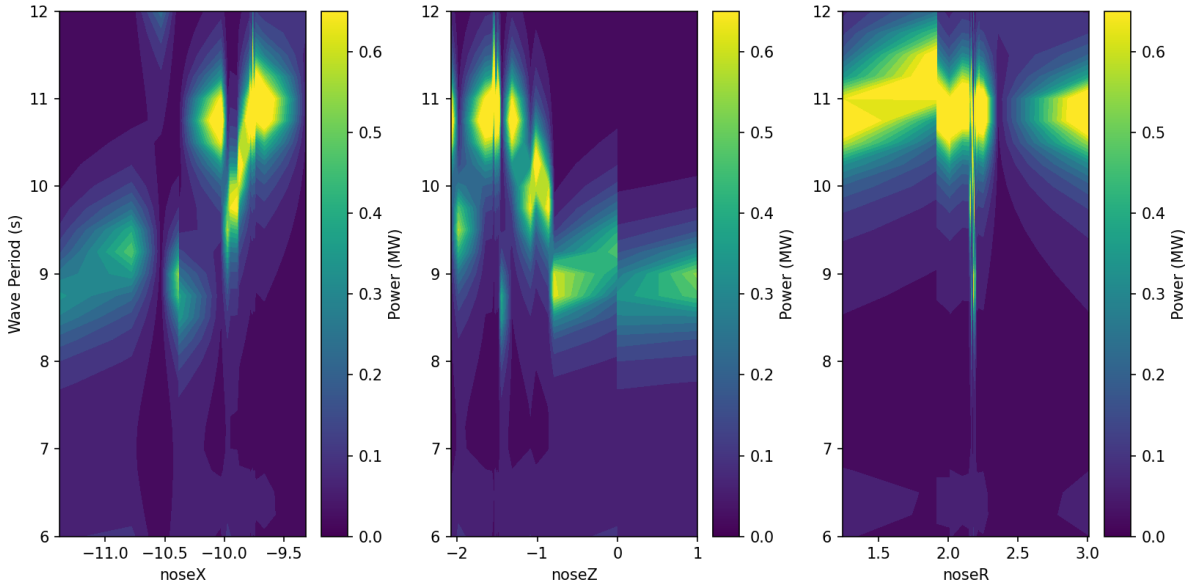

Figure 54: Power plots for the noseX, noseZ, noseR optimization

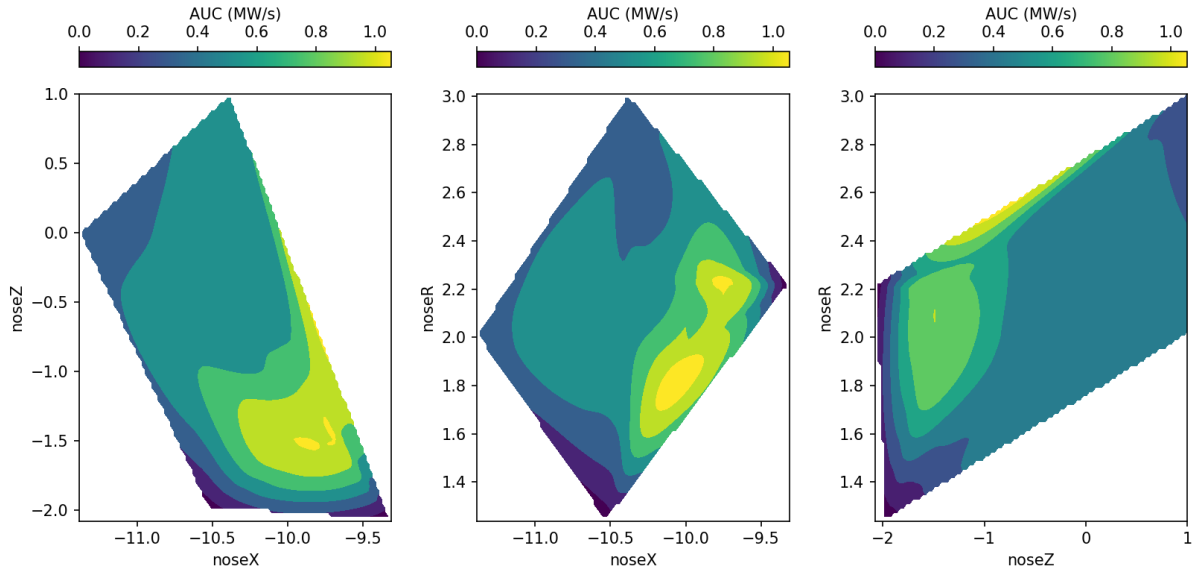

Figure 55: Area under the curve plots for the noseX, noseZ, noseR optimization

Bounds were applied to the design variables for this optimization to constrain the optimization within an appropriate range of values. The noseX was constrained within x-positions of -20 and -8.325 m. The noseZ dimension was constrained within z-positions of -2.5 and 3.0 m and the noseR dimension was constrained within radii of 0.1 to 3.0 m. These bounds were selected based on previous results found through single variable optimization. A volume constraint was also applied to maintain a constant below waterline volume of the PIP WEC.

To maximize the area under the curve, the optimizer decreased the noseX from -11.385 to -9.754 m, decreased the noseZ from 0 to -1.501 m, and slightly increased the noseR from 2.01 to 2.18 m. Decreasing the noseX value to maximize the area under the curve matches the results from the single variable noseX study, reaffirming that a "shorter" design produces more power. In the single variable optimization, the optimizer increased the noseZ value from an initial value of 0.0 m. However, a parameter sweep indicates that a noseZ value around -2.0 m would result in a greater area under the curve. Therefore, it appears appropriate that this multi-variable optimization reached a negative noseZ value to maximize the area under the curve. Decreasing the noseR value led to a maximum area under the curve in the single variable optimization, however the multi-variable optimization increased the noseR value. Based upon these multi-variable optimization results, the PIP WEC can be designed to maximize power output through a provided range of regular wave periods.

7.2 Lesson Learned and Test Plan Deviation

The following describes the lessons learned during the execution of the project that could help future TEAMER projects.

Lessons Learned

This project involved many changes to the design approach and model parameterization, which were made after learning more about how the numerical design tool worked. It was beneficial to create a numerical design tool to effectively encompass the different numerical modelers and dependencies and efficiently call their functions to switch between Python and MATLAB. However, as part of that software development, the design parameterization and the overall model approach kept changing, thereby pushing deadlines, and not allowing the full scope of objectives to be achieved.

If we could go back in time, we still keep the versatile numerical design tool, but we would try and modularize certain functions and capabilities of the program so that automation between Python and MATLAB could happen more independently from the design parameterization. This would also allow for earlier planning of how the optimization algorithms could be integrated. For future TEAMER projects, we would recommend earlier planning and testing of the architecture of the software to keep components modularized and allow for a setup where changes could be made efficiently without affecting other parts of the software.

Another lesson learned was to have a better contingency plan. The only contingency plan we had was whether WEC-Sim could be coupled with the Python modules, and we did not include any contingency plans for potential problems later in the project, such as changes to design parameterizations, or if more time was needed to achieve the objectives. Having those plans, however extreme, may have provided extra security and knowledge of what to do when plans changed later on in the project.

A third lesson learned was related to the transfer of information. During the numerical design tool development, it was asked that the code be explained and ready to use, even though it had not been finished or well-tested. This part of the software development phase also involves a decent knowledge of how to use various programming languages and development tools to simply run the program, which makes explaining how to use the design tool difficult to communicate to other users without that knowledge. In the future, I would set clearer tasks in the Test Plan on data transfer and user learning to explain the tool to a wider audience.

Test Plan Deviation

The following describes what was different in the final project compared to the initial Test Plan.

Roles and Responsibilities of Project Participants

Sal Husain was brought on to help David Ogden in the automation of meshing, BEM, and time-domain modeling workflow, as well as provide oversight and technical advice on the interpretation of WEC-Sim, Capytaine, and meshing results.

Andrew Stricklin was brought on as an intern to help with the selection of optimization algorithms and performing optimization studies, as well as contribute to final reporting.

Matt Hall did not participate in the project execution.

Stein Housner provided the primary software development for the automation of the meshing, BEM, and time-domain modeling workflow, led the selection of an optimizer, performed optimization studies, and contributed to the final reporting.

Project Objectives

In summary, the project objectives were to:

- Determine the optimal geometry, mass distribution, and operating parameters of the PIP WEC
- Automate the workflow between python and time-domain modeling in WEC-Sim
- Understand where performance is most sensitive to design and operating parameters

All objectives were achieved except for the first one. The optimal geometry, mass distribution, and operation parameters of the PIP WEC involves tens of design variables, which can involve many design iterations, each of which take on the order of minutes. The first and most logical step was to address the second objective and automate the workflow between Python mesh generation scripts and WEC-Sim in MATLAB. This provided the foundation for all future design optimizations and allowed for simple inputs and outputs to be made. This objective was achieved relatively early in the project timeline.

To verify the automation and integrate different optimizers, simple single-objective optimizations were performed to tackle the third objective, which taught us what parameters are most influential to the design performance. Throughout this process, we learned more about the PIP design through individual variables, but we also learned how to properly parameterize the design within the design tool to efficiently produce these optimization results. We were able to capture “dependent” variables, such as pitch natural period and metacentric height to better understand the PIP WEC’s behavior, in response to changes in “independent” variables, such as the nose location in x or the bottom radius.

The next step in the process would be to integrate different optimizers to the automation workflow to allow for different design variables to be used in multi-variable optimizations. However, there was not enough time available at the end of the project to do so and attempt to attain the first objective. For future projects, we would recommend keeping components modularized (as explained in “Lessons Learned”) to test different optimization methods without interfering with the Python-MATLAB workflow automation.

Data Management, Processing, and Analysis

The key output files from the numerical design tool are contained in the GitHub repository.

- System information, including the geometry, mass, and other properties, is contained within a text file in the repository
- Mesh data are stored in .nemoh files.
- Capytaine hydrodynamic data are stored in .nc files
- Hydrodynamic data for WEC-Sim are stored in .h5 files
- Time-series data for power and kinematics of the WEC are stored in .mat files that are deleted during optimizations for organization, but results are stored in the main output text file in the repository.

8 CONCLUSIONS AND RECOMMENDATIONS

This project created an efficient software tool to optimize the design of a WEC, including time-domain results from the automated workflow between Python packages and WEC-Sim in MATLAB. One iteration of each optimization includes the calculation of all dependent design properties, the generation of a new mesh of the WEC in Meshmagick, the computation of hydrodynamic coefficients in Capytaine, and a time-domain simulation of the WEC in WEC-Sim, all of which took on the order of 3 minutes to complete—a relatively small amount of time for that level of modeling. This kind of automation with Python and WEC-Sim had never been done before and can further be improved to include time-domain results in design processes.

Optimizations were performed of different design variables to determine which ones had the most influence on maximizing power output. The diameter of the tube, which houses the internal water coil, has the largest influence on power output, which directly increased the moment of inertia of the water coil, which increased the relative pitch motion of the PIP in WEC-Sim. The radius of the bottom circle was the next most influential design variable on the power output. The tilt of the device and the total system center of gravity in the z-direction had the next highest power output optimizations, all greater than the rest of the geometric design variables. The damping of the PTO had a notable influence on the power output, but only at relatively low damping values. All of these optimizations were completely dependent on the initial conditions of the other variables, otherwise known as the default PIP design. A completely different default PIP design would have likely resulted in much different optimal results and different natural periods. The contour plots of these single variable optimizations captured where the natural period was of the default design and how it changed for iterations of different design variables.

The multi-variable optimizations were the next step towards optimizing the total design and provided interesting insights into how the design wants to change with multiple design variables. These optimizations mostly confirmed the early single variable optimizations, but also showed cases where design variables changed in the opposite direction to improve the overall power output of the device. Including more than three variables in the optimization will likely require a different optimization method to produce an accurate optimal result and better handling of numerical errors in the design tool.

These optimizations were performed at regular wave periods ranging between 6 and 12 s. Changing this range, or the discretization of wave periods within the range, can notably alter the end results. The discretization determines the exact area under the period-power curve, which is used as the objective.

The optimizer chosen was the COBYLA optimizer from the SciPy package, as it was found to converge to the optimal value for a noseX single variable optimization much faster than any other SciPy optimizer. For the initial studies on the sensitivity of each design variable, this optimizer worked well and provided us with valuable insights. However, for larger, multi-variable optimizations, this is likely not the best optimizer. COBYLA may still converge in a relatively short number of iterations for these optimizations, but other SciPy optimizers or different optimization approaches are likely converge to more optimal solutions, which would help achieve the original first objective. For follow-on work, it would be good to integrate different types of optimizers into the design tool to test their efficiency and effectiveness but testing them outside of the functions that require Capytaine and WEC-Sim.

An assessment of the simulation fidelity and assumptions in WEC-Sim could also be performed to verify that the tool is providing accurate time-domain results. WEC-Sim is a robust and useful numerical tool and is able to provide average power results in this workflow in minimal time but has not yet been validated with this exact type of internal mass WEC. As such, there could be assumptions in the model that overlook some aspects of the design, such as the PTO representation or viscous effects, which can be double-checked.

Additionally, the entire design tool software architecture can likely be improved to better modularize the components that parameterize the design, run Capytaine and WEC-Sim, and call an optimizer. Since the tool was primarily in production throughout the length of the project, there is not much documentation or help for other users to run and test. Follow-on work should look at organizing the software better and writing documentation to help with future users.

In the end, two out of the three original project objectives were achieved, where the completion of the other objective would involve a deeper dive into how different optimization methods work and how they can be integrated into the design tool software. This would produce the optimal PIP WEC design. However, the optimal objective is sensitive to the design settings, such as which regular wave periods are used, meaning that the word “optimal” is highly dependent on what kind of simulations are run. Otherwise, all other metrics were achieved and an initial software training was performed to help other users learn how to use the tool.

9 REFERENCES

[1] IEC (2021). *IEC TS 62600-10:2021 – Part 10: Assessment of mooring system for marine energy converters (MECs).* Report by International Electrotechnical Commission (IEC).

[2] Ecole Centrale de Nantes, Meshmagick, https://github.com/LHEEA/meshmagick

[3] Ancellin et al., (2019). Capytaine: a Python-based linear potential flow solver. Journal of Open Source Software, 4(36), 1341, https://doi.org/10.21105/joss.01341

[4] Kelley Ruehl, Adam Keester, Nathan Tom, Dominic Forbush, Jeff Grasberger, Salman Husain, David Ogden, and Jorge Leon, "WEC-Sim v6.0". Zenodo, October 20, 2023. https://doi.org/10.5281/zenodo.10023797

[5] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, CJ Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E.A. Quintero, Charles R Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. (2020) **SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python**. *Nature Methods*, 17(3), 261-272.

[6] Steven G. Johnson, The NLopt nonlinear-optimization package, http://github.com/stevengj/nlopt

[7] J. S. Gray, J. T. Hwang, J. R. R. A. Martins, K. T. Moore, and B. A. Naylor, "OpenMDAO: An Open-Source Framework for Multidisciplinary Design, Analysis, and Optimization," Structural and Multidisciplinary Optimization, 2019.

10 ACKNOWLEDGEMENTS

This project would like to acknowledge the guidance, direction, and ideas from Nick Wynn from iProTech, the initial optimization guidance from John Jasa, and the funding opportunity from the TEAMER program.

11 APPENDIX