SANDIA REPORT

SAND2018-12858
Unlimited Release
Printed November 7, 2018

Initial results from wave tank test of closed-loop WEC control

Ryan G. Coe, Giorgio Bacelli, Steven J. Spencer, and Hancheol Cho

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

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

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by National Technology and Engineering Solutions of Sandia, LLC.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online ordering: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2018-12858
Unlimited Release
Printed November 7, 2018

Initial results from wave tank test of closed-loop WEC control

Ryan G. Coe
Water Power Technologies
Sandia National Laboratories
P.O. Box 5800, MS 1124
Albuquerque, NM 87185
rcoe@sandia.gov

Giorgio Bacelli
Water Power Technologies
Sandia National Laboratories
P.O. Box 5800, MS 1124
Albuquerque, NM 87185
gbacell@sandia.gov

Steven J. Spencer
High Consequence Automation and Robotics Department
Sandia National Laboratories
P.O. Box 5800, MS 1010
Albuquerque, NM 87185
sjspenc@sandia.gov

Hancheol Cho
Water Power Technologies
Sandia National Laboratories
P.O. Box 5800, MS 1124
Albuquerque, NM 87185
hancho@sandia.gov

Abstract

This report describes the set up, execution, and some initial results from a series of wave tank tests of a model-scale wave energy converter (WEC) completed in May 2018 at the Navy's Maneuvering and Sea Keeping (MASK) basin. The purpose of these tests was to investigate the implementation and performance of a series of closed-loop WEC power take-off (PTO) controllers, intended to increase energy absorption/generation.

Acknowledgment

Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525. The views expressed in this report do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

Contents

List of Figures

List of Tables

Nomenclature

PTO power take-off

MASK basin maneuvering and sea keeping basin

WEC wave energy converter

DOF degree of freedom

SID system identification

MPC model predictive control

PI proportional plus integral (a type of controller)

PDF probability density function

CCDF complementary cumulative density function

Executive summary

This reports provides a description of the experiments conducted during the MASK2B testing campaign at the Naval Surface Warfare Center, Carderock Division (NSWCCD) in Bethesda Maryland. The main objectives of the MASK2B wave tank tests are:

- Validation of predictionless control design procedures
- Validation of control design approach for electrical power maximization
- Collection of data to build multi-dimensional maps for multi-objective design
- Implementation of model predictive control (MPC) tuned in the frequency domain (no prediction), and comparison with basic feedback controller.
- Validation of testing approach using period inputs for tuning and performance assessment of control systems
- Collection of data for 3 degree of freedom (3DOF) system identification (SID)
- Collection of wave dataset for model validation with no buoy in place no buoy

This document is intended to be released together with the data collected during the MASK3 experiments. The main purpose of this report is to describe the data collection procedure and to provide some basic data analysis which show how to select and manipulate signals of interest. The Appendix provides sample code used to generate some of the figures in the report.

Chapter 1

Introduction

This report and the testing campaign on which it focuses are part of a multi-year project to investigate the implementation and performance of controllers for wave energy converters (WECs). As part of this project, model-scale wave tank experiments were conducted to validate models and test methods practical implementation. The wave tank tests were conducted in the US Navy's Maneuvering and Sea Keeping (MASK) basin, operated by the Naval Surface Warfare Center, Carderock Division (NSWCCD) in Bethesda Maryland.

The testing campaign includes three testing phases, as shown in Table 1.1. The first test (MASK1) focused on system identification (SID) for a 1 degree of freedom (DOF) WEC. In this test, the advantages of open-loop testing with periodic broad-banded input signals were studied [2–5]. MASK2A and MASK2B represent two phases of testing which were divided to improve the efficiency of experimentation. In MASK2A, basic closed-loop performance of the WEC system was studied [1]. In MASK2B, which is the focus of this report, a more complete study of 1DOF control of the WEC was considered, including maximization of electrical power, multi-objective performance mapping, and implementation of a novel predictionless control strategy [6]. Additionally, experiments were conducted to perform 3DOF SID. In MASK3, this work will be extended to include PTO system modeling, control system self-tuning, fatigue, and 3DOF control with shared power-electronics between each degree of freedom.

This report details the data collected and some initial results from MASK2B. The data collected are publicly available at `https://mhkdr.openei.org`. A test log detailing each experiment is included in Appendix A. In addition, sample code used to generate the results shown in this report is included in Appendix B. While there are a large number of potential analyses which can be conducted using this dataset, some of which are currently under examination, this report focuses only on description of the experiments (Chapter 1) and results comparing the performance of a proportional-integral (PI) controller with a model predictive controller (MPC) (Chapter 2), which has been designed to approximate the PI controller while providing the additional benefit of allow-

Table 1.1: MASK testing phases.

MASK1	**MASK2A**	**MASK2B**	**MASK3**
1DOF SID [2–5]	Study/verification of basic 1DOF closed-loop performance [1]	1DOF control (presented herein) & 3DOF SID	Autonomous (self-tuning) 1DOF & 3DOF control

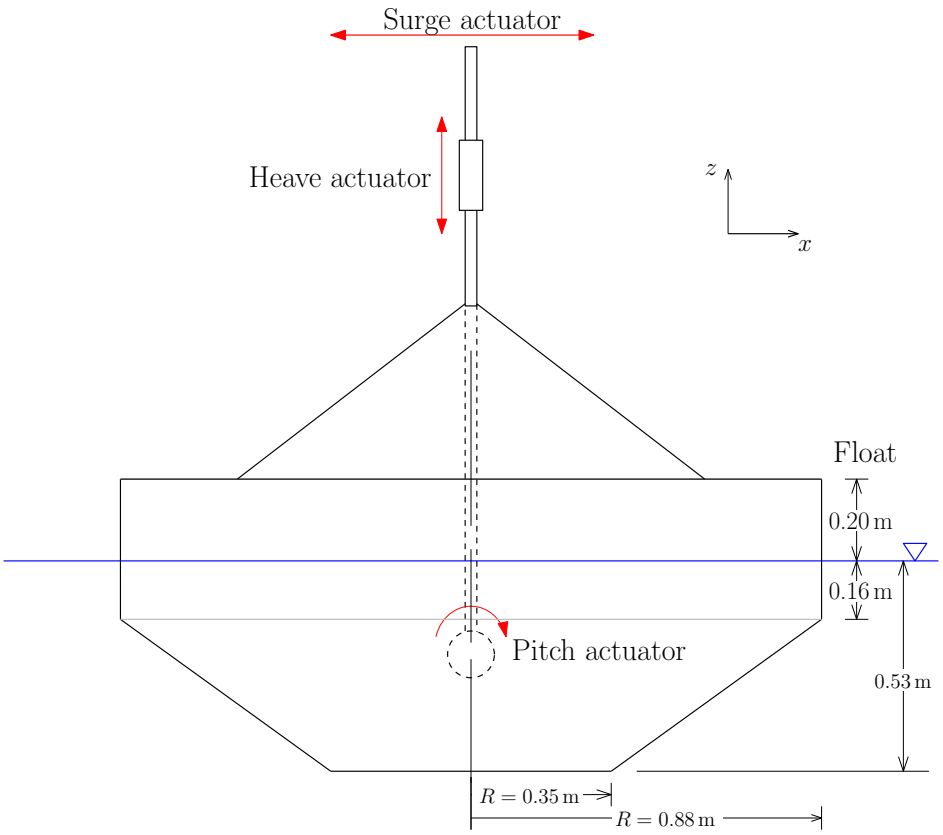


Figure 1.1: Test device diagram.

ing for constraints. An extended discussion of these control approaches and numerical results are presented in [6].

1.1 Test device

For this research project, a single device, which is shown in Figure 1.1, has been developed. The physical properties of this device are shown in Table 1.2. The device is independently actuated in three degrees of freedom: heave, surge, and pitch (all of the motions in a single plane). This device was initially designed in 2015 [7], but has undergone multiple modifications (see, e.g., [1, 8]). Primarily, the drivetrain and measurement/data-acquisition systems have been updated. Additionally, after MASK1, the mounting location was updated (see additional discussion in Section 1.3).

1.2 Drivetrain & instrumentation

Figure 1.3 shows a diagram of the WEC device and a photograph of the physical hardware. In Figure 1.3a, the three actuation motors are called out in blue. Similarly, five locations within the

Table 1.2: Model-scale WEC physical parameters.

Parameter	Value
Surge rigid-body inertia, m_1 [kg]	1420
Heave rigid-body inertia, m_3 [kg]	893
Pitch rigid-body inertia, m_5 [kg m^2]	84
Displaced volume, $\forall$ [m^3]	0.858
Float radius, r [m]	0.88
Float draft, T [m]	0.53
Water density, ρ [kg/m^3]	1000
Water depth, h [m]	6.1
Linear hydrostatic stiffness, G [kN/m]	23.9
Infinite-frequency added mass, A_∞ [kg]	822
Max vertical travel, $\lvert z_{\max} \rvert$ [m]	0.6

system are defined in Figure 1.3a as follows:

1. **Control Station** - The remote location where engineers can sit and conduct/observe the test
2. **Weldment** - The steel structure supporting the WEC
3. **Carriage** - The base of the carriage which moves in surge with the device
4. **PTO Tower** - The mounting location that moves in heave and surge
5. **Float** - The "buoy" that interacts with the water

A diagram of the systems instrumentation is shown in Figure 1.2. The location numbers from Figure 1.2 are shown in Figure 1.3a. More in-depth discussions of the actuators, sensors, and data-acquisition system design processes are provided in [1].

Table 1.3 provides a full listing of the sensors utilized on the model-scale WEC.[1] Additionally, variable names from the `.mat` data files available for download at `https://mhkdr.openei.org` are also provided. For examples on how to use these data, refer to the sample code provided in Appendix B. Some additional discussion on these signals and best-practices for common analyses is provided below.

- **Surge force** - The surge belt tension (`s.F`) measurement is a singled-sided (down-wave) measurement of the belt tension. To get the surge force, use either the surge motor torque or surge motor current with appropriate scaling factors for gearing and current (`S.tau * S.N` or `S.I * S.kt * S.N`). When the surge degree of freedom is locked out, the surge force measurement can be obtained from `s.F_spring`.

[1] See Section 1.3 for information on wave sensors.

- **Heave force** - The heave force can be obtained from `h.F`, `h.tau * h.N`, or `h.I * h.Kt * h.N`. When the heave degree of freedom is locked out, the heave force can be obtained from `h.F`.
- **Pitch moment** - The pitch moment can be obtained from `p.F`, `p.tau`, or `p.I * p.Kt * p.N`. When the heave degree of freedom is locked out, the heave force can be obtained from `p.F_lockout * 0.93`, where 0.93 m is the moment arm for the pitch lockout load cell.

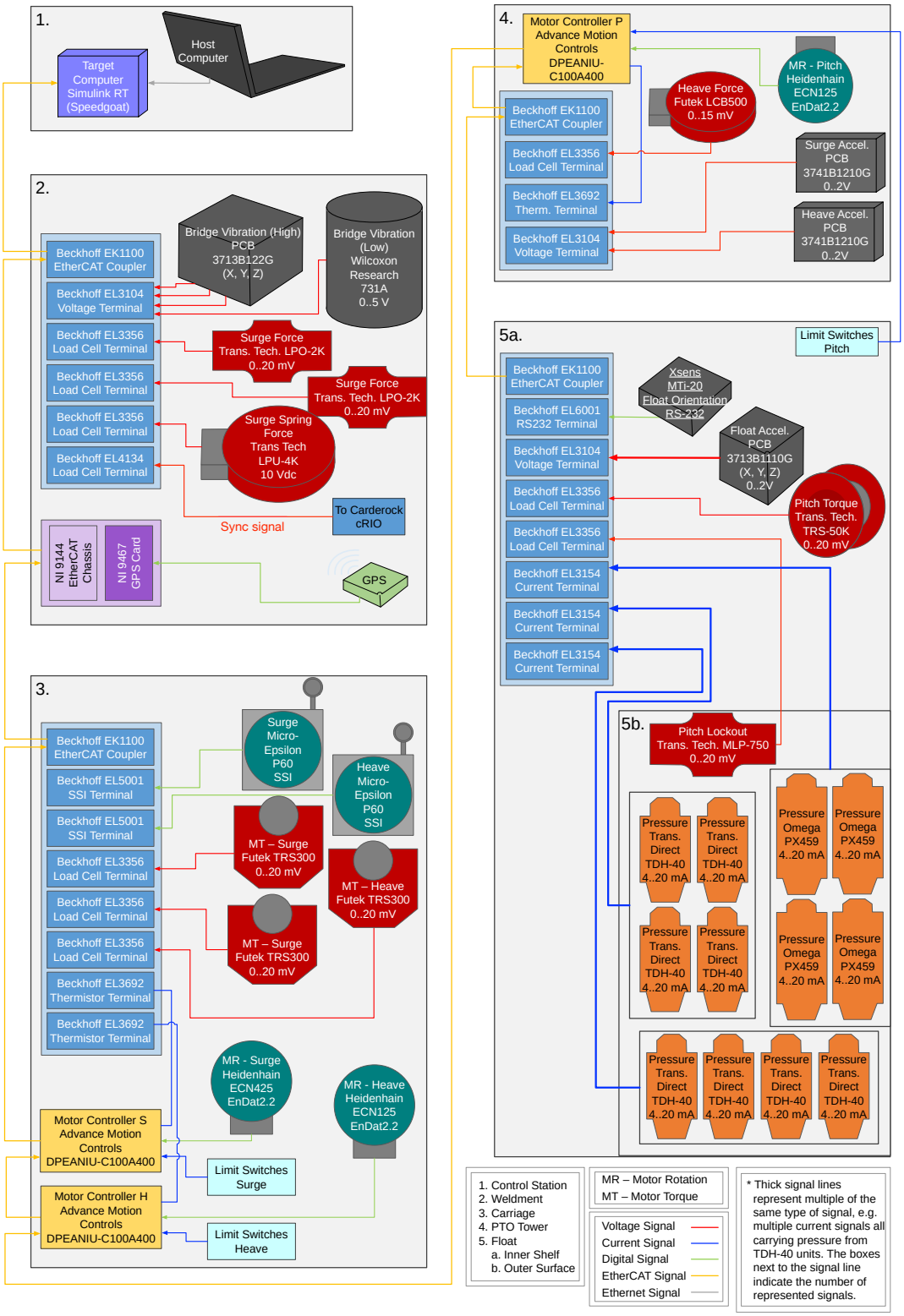
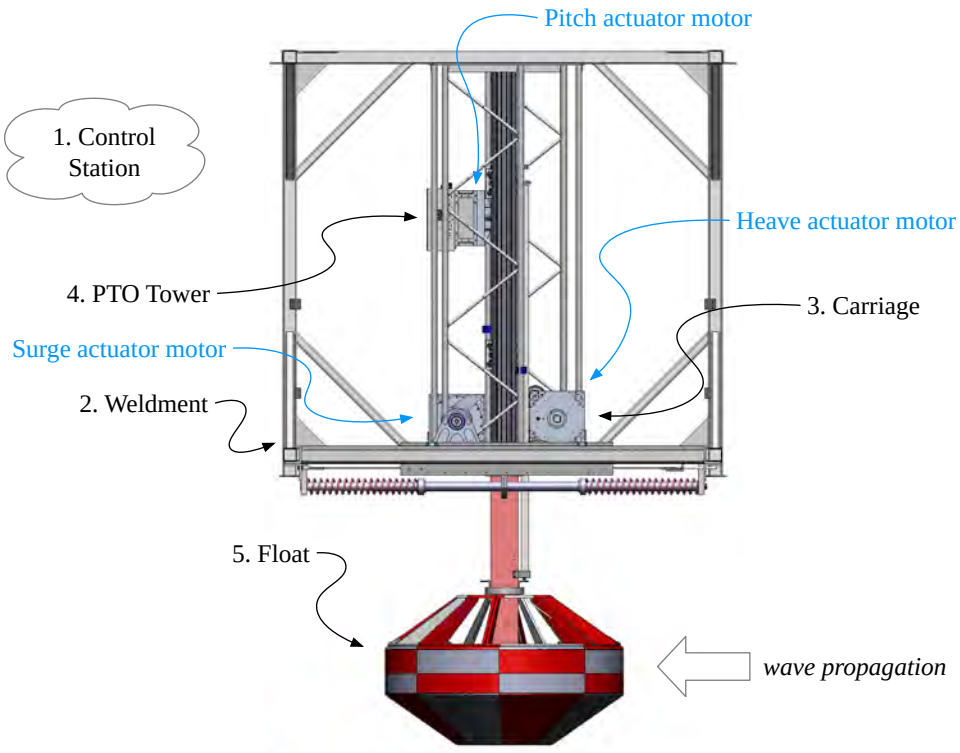
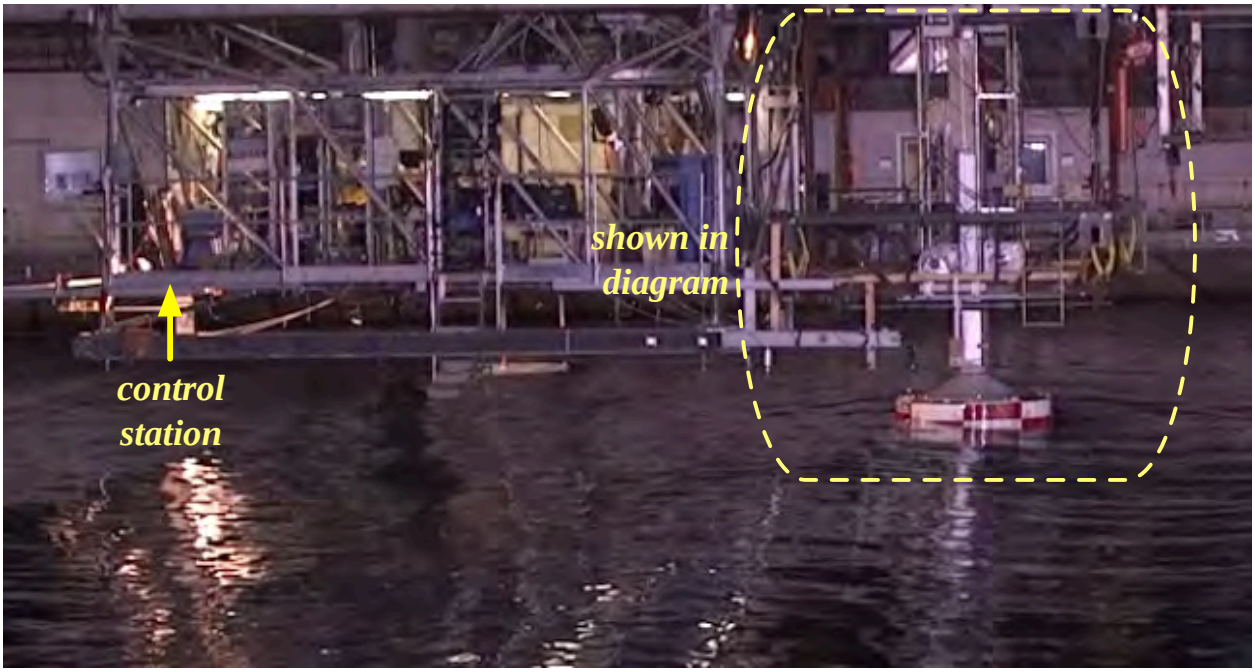


Figure 1.2: Signal diagram for real-time data acquisition and control system [1].



(a) CAD model/diagram; actuator motors shown in blue; DAQ arrangement shown in black with labels and numbering corresponding to Figure 1.2.



(b) Test device installed in MASK basin

Figure 1.3: WEC device used in testing [1].

Table 1.3: WEC experimental instrumentation and sensors.

Measurement/purpose	Instrument		Variable name(s)
	Make	Model	
Surge motor	Allied Motion	MF0310100-C0X	
Surge motor current	AMC	DPEANIU-C100A400	s.I
Surge belt tension	Trans. Tech.	LPO-2K	s.F
Surge lockout force	Trans. Tech.	LPU-4K	s.F_spring
Surge displacement	Micro-Epsilon	P60	s.x_sp
Surge motor torque	Futek	TRS300	s.tau
Surge motor orientation	Heidenhain	ECN125	s.x_enc
Surge acceleration	PCB	3741B1210G	s.acc
Surge gearing			s.N
Surge torque constant			s.Kt
Heave motor	Allied Motion	MF0310100-C0X	
Heave motor current	AMC	DPEANIU-C100A400	h.I
Heave displacement	Micro-Epsilon	P60	h.x_sp
Heave motor orientation	Heidenhain	ECN425	h.x_enc
Heave motor torque	Futek	TRS300	h.tau
Heave force	Futek	LCB500	h.F
Heave acceleration	PCB	3741B1210G	h.acc
Heave gearing			h.N
Heave torque constant			h.Kt
Pitch motor	Allied Motion	MF0310100-C0X	
Pitch motor current	AMC	DPEANIU-C100A400	p.I
Pitch motor orientation	Heidenhain	ECN125	p.x_enc
Pitch torque	Trans. Tech.	TRS-50K	p.tau
Pitch lockout force	Trans. Tech.	MLP-750	p.F_lockout
Pitch accel./orientation	Xsens	MTi-20	p.imu
Pitch gearing			p.N
Pitch torque constant			p.Kt
Float Surface Pressure	Omega	PX459	b.pres
Float Surface Pressure	Trans. Direct	TDH-40	b.pres
GPS time	NI	NI-9467	b.gps
Bridge acceleration	Wilcox	731A	b.acc_pb
Synchronization signal	Beckhoff	EL3104	b.SyncSine
Wavemaker signal	Beckhoff	EL3104	b.WavesOn

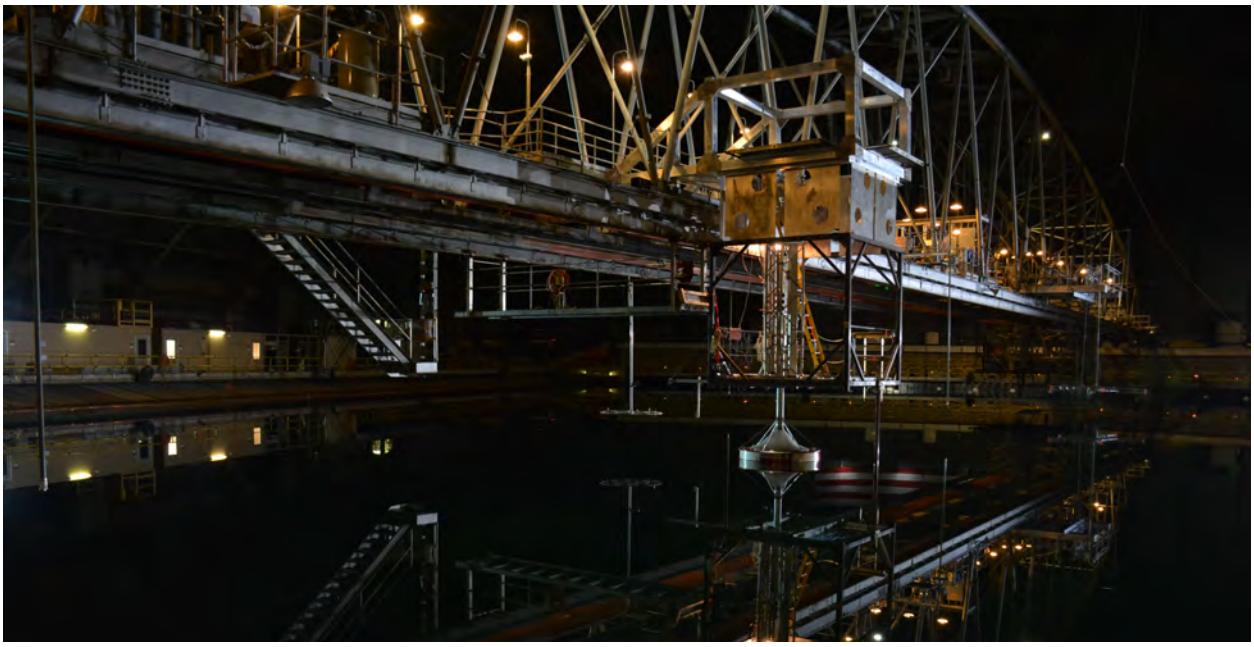
1.3 Wave basin

The MASK is an indoor basin with an overall length of 110 m (360 ft), a width of 73 m (240 ft) and a depth of 6.1 m (20 ft) except for a 10.7 m (35 ft) deep trench that is 15.2 m (50 ft) wide and parallel to the long side of the basin (on the south side). The basin is spanned by a 115 m (376 ft) bridge. The bridge can be rotated within the basin.

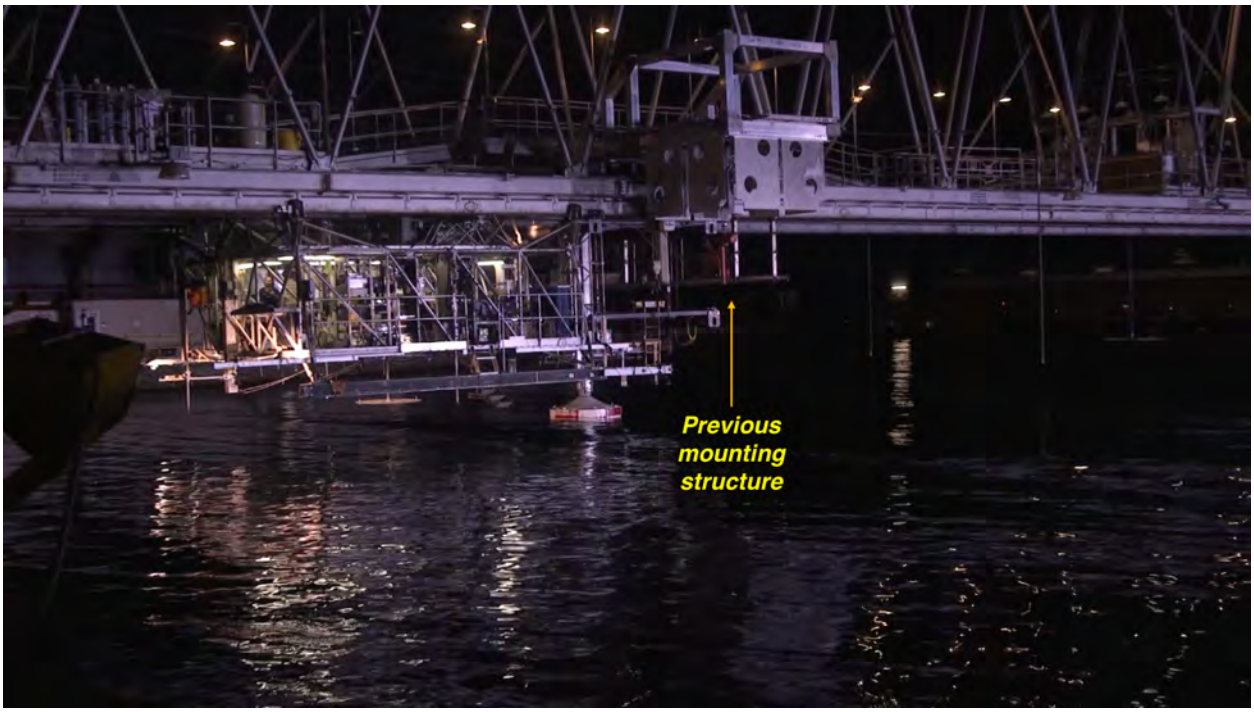
The arrangement of the WEC device and wave measurement sensors within the basin was similar to previous studies [1–3], but with some updates. Referring back to the testing time-line provided in Table 1.1, the location of the WEC device was moved to directly beneath the bridge after MASK1. This was done due to the unplanned compliance of the structure fabricated to cantilever the WEC off the side of the bridge.

Additionally, some minor changes were made to locations of wave probes within the basin (mostly to those mounted in the vicinity of the device). Figure 1.5 shows the locations of wave sensors and the WEC within the basin for this test (MASK2B). A table providing the locations of wave sensors and the device is provided in Table A.2. Table A.2 also provides a description of the variable names for the wave sensors.

Waver makers paddles are located along the $x = 0$ and $y = 0$ walls. The wave makers comprise 216 individually controlled paddles, which rely on a force feedback system. Concrete beaches are located on the remaining walls. Wave propagation, as shown by the arrow in Figure 1.5 occurs at an angle of 70°.



(a) MASK1 test arrangement



(b) MASK2A, 2B, and 3 test arrangement

Figure 1.4: WEC device mounting.

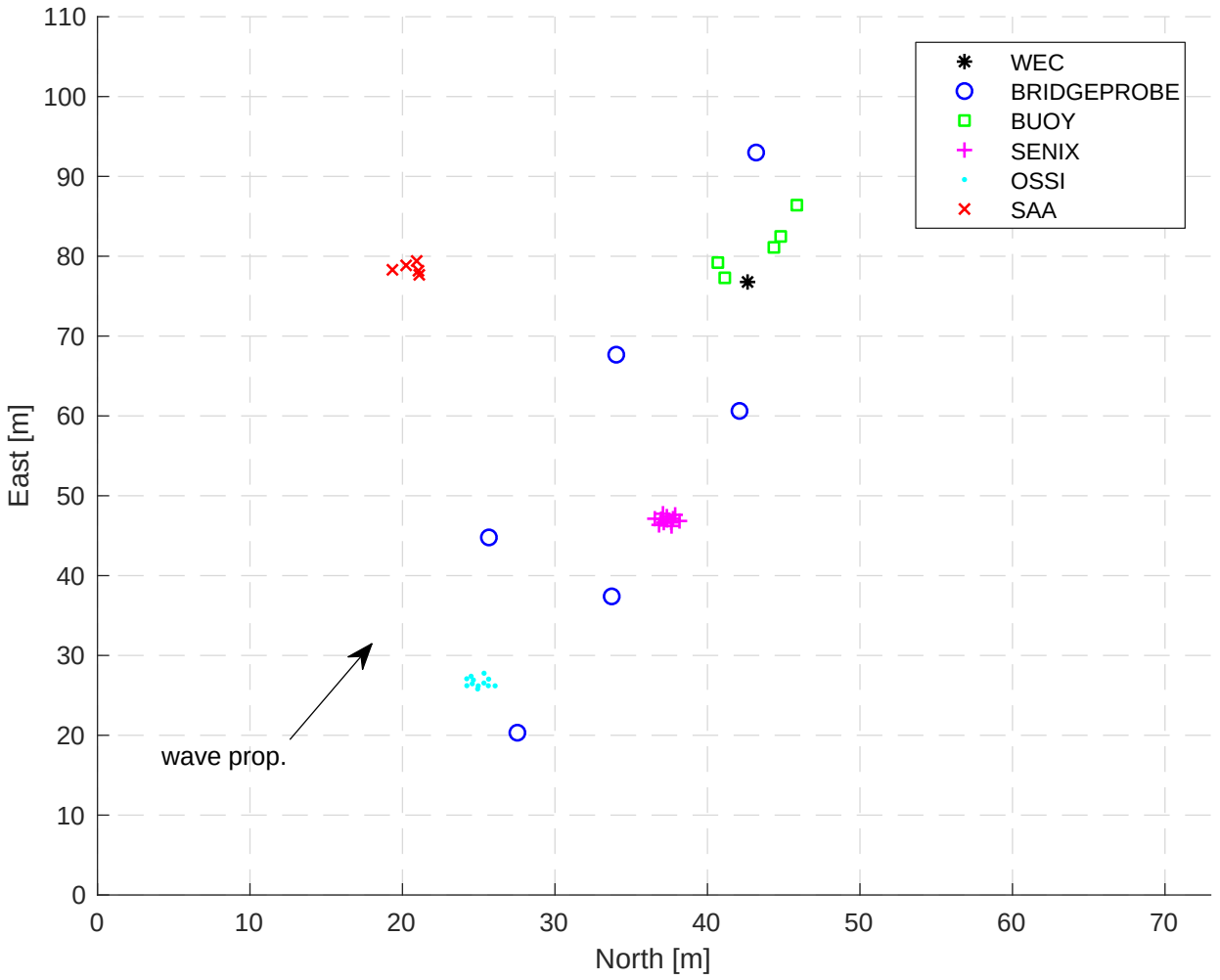


Figure 1.5: Wave basin layout. See Table A.2 for complete listing.

1.4 Test cases

In this report, the external waves applied to the WEC device include eleven different states, all of which are of JONSWAP type and summarized in Table 1.4. A full listing of test cases is provided in Table A.1. These sea states were selected based on the relative location of energy compared to the natural resonance of the tested WEC [1].

Table 1.4: List of sea states acting on the plant (WEC device).

Test Case	**Peak period, T_p [s]**	**Significant wave height, H_s [in]**	**Peak enhancement factor, γ [-]**
1	1.58	5	1
2	1.58	5	3.3
3	2.5	5	1
4	2.5	5	3.3
5	2.5	10	1
6	2.5	10	3.3
7	3.5	5	1
8	3.5	5	3.3
9	3.5	10	1
10	3.5	10	3.3
11	3.5	15	3.3

In general, sea states were realized with a 5 minute repeat period. The repeat period of the sea state is determined by the frequency resolution. This has been shown to provide sufficient frequency resolution and no appreciable difference in energy content when compared with a 2 hour repeat period wave [1, 3]. For each test case, three phase realizations (“A, B, and C”) where considered.

Chapter 2

Initial control performance results

In this chapter, several initial results are provided regarding the performance of the controllers designed for the MASK2B wave tank tests. As discussed in the Executive Summary, the results presented herein do not, by any means, represent an exhaustive analysis of this experimental dataset. Instead, the results presented in this report serve to provide an example of an analysis which can be performed with this dataset.

Thus, here we consider the performance of two WEC controllers. These are a proportional-integral (PI) controller and a model predictive controller (MPC). The MPC is an approximation of the PI controller based on the method proposed by Cairano and Bemporad [9], and requires only a nominal wave prediction (e.g., 0.001 s). The MPC has the added benefit of enabling the implementation of constraints. The theoretical basis for these controllers and a numerical comparison are provided and detailed in [6].

For this comparison of the PI and MPC, there are two major objectives to be achieved. First, we check if the MPC behaves as a predesigned PI controller when constraints are inactive and if the MPC simultaneously satisfies given constraints when they are active. Second, the effect of varying the gains of the PI controllers on the power absorption is investigated and the optimal gains that maximize the power absorption are estimated. In this chapter, it is assumed that only the heave motion is controlled. The controllers that incorporate the motions of heave, surge, and pitch together in three degrees of freedom will be handled in future work.

2.1 Problem Setup

A block diagram describing the whole closed-loop system is shown in Figure 2.1. There are two inputs into the WEC plant: wave excitation force and a control force. In the block diagram, the controller comprises either the PI controller or the unconstrained/constrained MPC. The outputs of the plant are the (vertical) displacement of the buoy (z) and the velocity (v); both z and v are used by the controller to calculate its control signal that is applied to the plant. The control objective is to maximize the average (electrical) power capture, which is calculated using F_{cntrl} and v, where F_{cntrl} is the control force. To obtain electrical power, use a simple, but fairly accurate model a DC electric motor: $P_{elec} = F_{cntrl}v + (\frac{F_{cntrl}}{K_m K_N})^2$, where K_m is the motor (loss) constant ($\frac{Nm}{\sqrt{W}}$) and K_N is the transmission ratio.

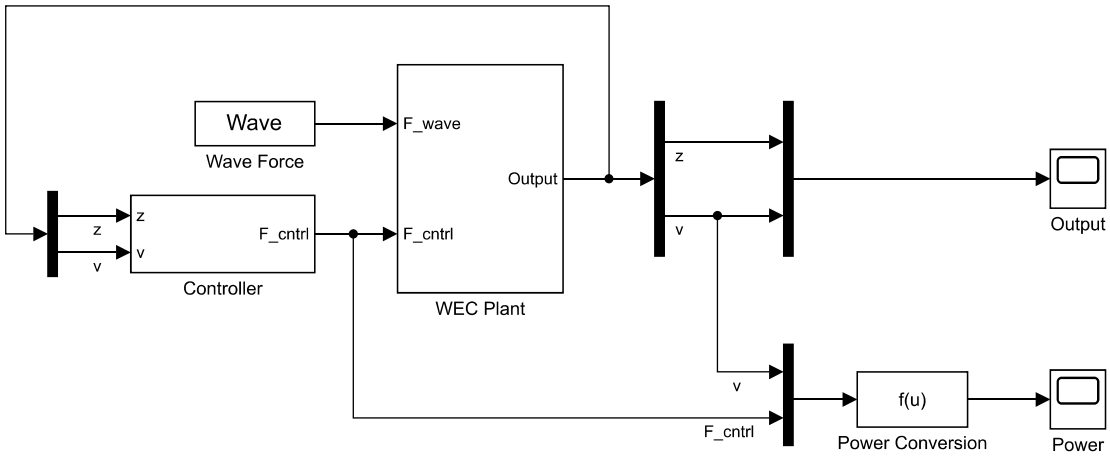


Figure 2.1: Block diagram of the closed-loop system.

First, let us consider the design of a PI controller in discrete time. With the state vector $\boldsymbol{x}(k) = [z(k)\ v(k)]'$ where $\cdot(k)$ is a quantity at the time step k, the PI controller has the following form:

$$\boldsymbol{u}_{\mathrm{PI}}(k) = \mathbf{K}\boldsymbol{x}(k), \tag{2.1}$$

where $\mathbf{K} = [K_I\ K_P]$ is a gain matrix and K_I and K_P are the I and P gains, respectively. Next, an MPC is designed by solving at every control cycle $k = 0, 1, ...$, the finite-horizon optimal control problem

$$\mathscr{V}(\boldsymbol{x}(k), \boldsymbol{u}(k)) = \min_{\boldsymbol{U}(k)} \boldsymbol{x}'(N|k)\mathbf{P}\boldsymbol{x}(N|k) + \sum_{i=0}^{N-1} \boldsymbol{x}'(i|k)\mathbf{Q}\boldsymbol{x}(i|k) + \boldsymbol{u}'(i|k)\mathbf{R}\boldsymbol{u}(i|k), \tag{2.2}$$

s.t.

$$\boldsymbol{x}(i+1|k) = \mathbf{A}\boldsymbol{x}(i|k) + \mathbf{B}\boldsymbol{u}(i|k), \quad i = 0, ..., N-1, \tag{2.3}$$

$$x_{min} \leq \|\boldsymbol{x}(i|k)\| \leq x_{max}, \quad i = 0, ..., N, \tag{2.4}$$

$$u_{min} \leq \|\boldsymbol{u}(i|k)\| \leq u_{max}, \quad i = 0, ..., N-1, \tag{2.5}$$

$$\boldsymbol{x}(0|k) = \boldsymbol{x}(k), \tag{2.6}$$

where (2.3) is the discrete-time state space equation of the plant model, $\boldsymbol{x} \in \mathbb{R}^n$ is the state vector, $\boldsymbol{u} \in \mathbb{R}^m$ is the control input vector, N is the prediction horizon, $\boldsymbol{U}(k) = [\boldsymbol{u}'(0|k) \ \ldots \ \boldsymbol{u}'(N-1|k)] \in \mathbb{R}^{Nm}$ is the vector to be optimized, and $\mathscr{V} : \mathbb{R}^n \to \mathbb{R}_{0+}$ is the value function. The matrices $\mathbf{P} \in \mathbb{R}^{n\times n}$, $\mathbf{Q} \in \mathbb{R}^{n\times n}$, and $\mathbf{R} \in \mathbb{R}^{m\times m}$ are the weight matrices that should be tuned such that the resulting MPC yields exactly the same controller as the PI controller in (2.1).

Di Cairano and Bemporad [9] proposed a detailed tuning procedure to get such weight matrices $\mathbf{P}$, $\mathbf{Q}$, and $\mathbf{R}$ by recasting the problem into a convex optimization problem but it necessitates a numerical optimization solver and the performance can be potentially degraded as a result of convexification. Hence, in this report, new analytical methodology proposed in [6] is used that immediately calculates the weight matrices without any numerical optimization. The MPC signals are computed by the Model Predictive Control Toolbox of MATLAB. The prediction horizon is set as $N = 2$ and the control horizon is set as the same as the prediction horizon because shorter horizon requires less computational loads and the control performance is independent of N in an unconstrained application. Even for constrained cases, it will be shown that good performance is still observed with the short horizon where the control output saturation is considered as the constraint in this test. The obtained control results, such as the control forces, velocities of the buoy, and the mean power capture, will be shown in the next section.

As mentioned earlier, the state vector is defined as $\boldsymbol{x}(k) = [z(k) \ v(k)]'$ and so the following state space equation of second order is employed in discrete time for the WEC plant:

$$\begin{aligned} \boldsymbol{x}(k+1) &= \mathbf{A}\boldsymbol{x}(k) + \mathbf{B}\boldsymbol{u}(k), \\ \boldsymbol{y}(k) &= \mathbf{C}\boldsymbol{x}(k) + \mathbf{D}\boldsymbol{u}(k). \end{aligned} \tag{2.7}$$

Assuming the sampling interval $T_s = 0.005$ s, the matrices in (2.7) are given by

$$\mathbf{A} = \begin{bmatrix} 0.9998 & 0.004983 \\ -0.07804 & 0.9930 \end{bmatrix}, \ \mathbf{B} = \begin{bmatrix} 0.005097 \\ -0.008431 \end{bmatrix}, \ \mathbf{C} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \ \mathbf{D} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{2.8}$$

2.2 Experimental results #1: MPC performance

Using the MPC designed in Section 2.1, MPC's performance is analyzed and compared with the PI controller. First, the control signal created by the MPC is checked to assess whether it matches the one provided by the PI controller. Next, the resulting velocity of the buoy obtained by the two controllers (MPC and PI) are compared. Finally, the power captured using the MPC and PI controller is also investigated. Throughout all the experiments the prediction horizon is selected as $N = 2$. The MPC is created for both unconstrained and constrained cases and the constraint includes only the control output saturation.

Among the 11 cases listed in Table 1.4, the sea states #3, 5, 8, and 9 were selected to obtain the MPC performance. Also, for each sea state, three different phases were introduced, designated A, B, and C, respectively. Hence, the total number of test cases considered in this section is 12 (i.e., #3A, 3B, 3C, 5A, 5B, 5C, $\cdots$). The PI controller and the MPC were tested and compared for

Table 2.1: Test case numbers and the corresponding test ID numbers in Table A.1.

Test Case #		Test ID	Test Case #		Test ID	Test Case #		Test ID
3A	**PI**	062	**3B**	**PI**	065	**3C**	**PI**	066
	MPC	067		**MPC**	068		**MPC**	069
5A	**PI**	033	**5B**	**PI**	031	**5C**	**PI**	032
	MPC	034		**MPC**	035		**MPC**	036
8A	**PI**	009	**8B**	**PI**	041	**8C**	**PI**	042
	MPC	038		**MPC**	043		**MPC**	044
9A	**PI**	070	**9B**	**PI**	071	**9C**	**PI**	072
	MPC	073		**MPC**	074		**MPC**	078

each case and Table 2.1 provides the test case numbers and the corresponding experimental test ID numbers in Table A.1 used to evaluate the PI and MPC.

The relative error between the two controllers is measured by 'FIT' (%) - the mean square error between the MPC and PI signals.

$$FIT = (1 - NRMSE) \times 100 \tag{2.9}$$

Here, the term 'NRMSE' is defined as

$$NRMSE = \frac{||s_{\text{PI}} - s_{\text{MPC}}||_2}{||s_{\text{PI}} - \bar{s}_{\text{PI}}||_2}, \tag{2.10}$$

where s_{PI} is the signal created by the PI controller, $\bar{s}_{\text{PI}}$ is the mean value of s_{PI}, and s_{MPC} is the signal calculated by the (unconstrained) MPC. The FIT values were obtained in terms of control force, velocity, and power capture. Also, one more metric K_{FIT} was calculated by estimating the K_I and K_P gains back from the data and comparing them with the actual gains of the PI controller used for the test. Table 2.2 shows the commanded gains for PI controller for each test case. This metric is defined by

$$K_{FIT} = \left(1 - \sqrt{\left(\frac{\tilde{K}_I - K_I}{K_I}\right)^2 + \left(\frac{\tilde{K}_P - K_P}{K_P}\right)^2}\right) \times 100, \tag{2.11}$$

where $\tilde{K}_I$ and $\tilde{K}_P$ are the integral and proportional gains, respectively, estimated from the data and K_I and K_P are the commanded gains listed in Table 2.2. The gains were estimated by dividing the measured control force by the measured position and velocity and then picking up the constant optimal values that minimize the fitting error. The estimated gains are given in Table 2.3.

Note that in order to compare two experiments (e.g., MPC and PI for case #3A, which requires comparing tests 062 and 067, per Table 2.1), we utilize a synchronization method based on the

Table 2.2: Commanded K_I and K_P gains used for PI controller.

Test Case	K_I	K_P
3A, 3B, 3C	3000	-2400
5A, 5B, 5C	3000	-2400
8A, 8B, 8C	2500	-3300
9A, 9B, 9C	2400	-3200

Table 2.3: Recomputed integral, $\tilde{K_I}$, and proportional, $\tilde{K_P}$, gains from MPC tests.

3A	$\tilde{K_I}$	2983	**3B**	$\tilde{K_I}$	2981	**3C**	$\tilde{K_I}$	2986
	$\tilde{K_P}$	-2417		$\tilde{K_P}$	-2417		$\tilde{K_P}$	-2417
5A	$\tilde{K_I}$	2985	**5B**	$\tilde{K_I}$	2986	**5C**	$\tilde{K_I}$	2984
	$\tilde{K_P}$	-2417		$\tilde{K_P}$	-2417		$\tilde{K_P}$	-2417
8A	$\tilde{K_I}$	2450	**8B**	$\tilde{K_I}$	2450	**8C**	$\tilde{K_I}$	2451
	$\tilde{K_P}$	-3303		$\tilde{K_P}$	-3303		$\tilde{K_P}$	-3303
9A	$\tilde{K_I}$	2389	**9B**	$\tilde{K_I}$	2388	**9C**	$\tilde{K_I}$	2389
	$\tilde{K_P}$	-3213		$\tilde{K_P}$	-3213		$\tilde{K_P}$	-3213

rising edge of the wavemaker activation signal. Additionally, as discussed in [1], we used periodic wave maker signals which repeat every 5 minutes. This allows for efficient testing and data processing.

Table 2.4 shows the values of the metrics FIT and K_{FIT} for the cases #3, 5, 8, and 9. The values were evaluated in terms of the control force, velocity, power capture, and the recomputed controller gains. It is found that the phase difference (i.e., comparing the A, B, and C phase realizations of each sea state) has a generally, but not always, small influence on the agreement and that the MPC produces signals that are very similar to the PI signals on the whole. The test case #3B has exceptionally low FIT values and the reason will be rigorously investigated in future work. However, even in this worst case, K_{FIT} for the recomputed controller gains is obtained as 99.07%. This likely points to some large difference in the wave input. The total averages of the FIT and K_{FIT} of all test cases for the control force, velocity, power capture, and recomputed controller gains are calculated as 89.62%, 89.91%, 85.47%, and 98.91%, respectively.

Table 2.5 lists the average power captured by the PI controller and the MPC for each test case. Note that the convention of this report is to show absorbed power as negative power. It is obvious that the differences between the PI controller and the MPC are small. Note particularly, that even in case #3B, where we observed the largest mismatch between the controllers in Table 2.4, the difference in average power captured is roughly 6%.

The top-most plot in Figure 2.2 displays the time history of the control signals created by the PI controller and the MPC for test case #3B. Even in this worst case, the FIT is calculated as 72.54%, which verifies that the obtained MPC behaves as the PI controller on the whole except that there is a little time delay between the two curves. Qualitatively, one can see from Figure 2.2 that the two

Table 2.4: Recomputed integral, $\tilde{K_I}$, and proportional, $\tilde{K_P}$, gains from MPC tests.

3A	F_{cntrl}	90.4	**3B**	F_{cntrl}	72.54	**3C**	F_{cntrl}	89.77	**3, avg.**	F_{cntrl}	84.24
	v	88.88		v	70.22		v	90.93		v	83.34
	Power	83.74		*Power*	56.95		*Power*	86.65		*Power*	75.78
	K_{FIT}	99.09		K_{FIT}	99.07		K_{FIT}	99.15		K_{FIT}	99.10
5A	F_{cntrl}	96.82	**5B**	F_{cntrl}	94.75	**5C**	F_{cntrl}	94.83	**5, avg.**	F_{cntrl}	95.47
	v	97.21		v	94.73		v	95.79		v	95.91
	Power	96.04		*Power*	91.94		*Power*	94.32		*Power*	94.10
	K_{FIT}	99.13		K_{FIT}	99.16		K_{FIT}	99.12		K_{FIT}	99.14
8A	F_{cntrl}	80.25	**8B**	F_{cntrl}	95.85	**8C**	F_{cntrl}	95.94	**8, avg.**	F_{cntrl}	90.68
	v	85.46		v	95.75		v	96.06		v	92.42
	Power	79.32		*Power*	93.98		*Power*	94.66		*Power*	89.32
	K_{FIT}	98		K_{FIT}	98.01		K_{FIT}	98.02		K_{FIT}	98.01
9A	F_{cntrl}	95.79	**9B**	F_{cntrl}	88.85	**9C**	F_{cntrl}	79.63	**9, avg.**	F_{cntrl}	88.09
	v	95.52		v	89.01		v	79.4		v	87.98
	Power	93.81		*Power*	83.7		*Power*	70.57		*Power*	82.69
	K_{FIT}	99.38		K_{FIT}	99.37		K_{FIT}	99.39		K_{FIT}	99.38

Table 2.5: Average power (W) captured by PI controller and MPC.

3A	**PI**	-5.28	**3B**	**PI**	-4.90	**3C**	**PI**	-5.05	**3, avg.**	**PI**	-5.09
	MPC	-5.07		**MPC**	-5.18		**MPC**	-5.17		**MPC**	-5.13
5A	**PI**	-26.83	**5B**	**PI**	-27.25	**5C**	**PI**	-26.95	**5, avg.**	**PI**	-27.04
	MPC	-26.51		**MPC**	-26.55		**MPC**	-26.86		**MPC**	-26.53
8A	**PI**	-8.08	**8B**	**PI**	-8.58	**8C**	**PI**	-8.67	**8, avg.**	**PI**	-8.33
	MPC	-8.90		**MPC**	-8.78		**MPC**	-8.69		**MPC**	-8.84
9A	**PI**	-33.61	**9B**	**PI**	-35.21	**9C**	**PI**	-33.37	**9, avg.**	**PI**	-34.41
	MPC	-33.76		**MPC**	-34.05		**MPC**	-32.85		**MPC**	-33.91

controllers perform quite similarly.

In order to ensure that the two controllers (PI and MPC) were influenced by the same external wave, the wave elevation data collected from the PI controller and MPC experiments are plotted in frequency domain and compared in Figure 2.3. In the inset, the wave elevation data in time domain are plotted for reference. The wave elevation was measured by a sensor that was located far enough from the buoy so that any radiation effect from the buoy could be ignored.[1] Figure 2.3 shows that the two controllers were indeed under the quite similar wave forces.

In Table 2.6, the FIT values comparing the two wave elevations for each test case are given. It is noted that a low pass filter with cutoff frequency 5 Hz is applied to each case to remove high-frequency noise. It is found that the two wave elevations match quite well for each test case. In fact, the FIT of the two wave elevations in the #3B case is calculated as 83.28%.

Since the FIT values are calculated in time domain, we employ another metric to check spectral equivalence. Table 2.7 provides the spectral moments of interest, which are defined for a spectral density $S(\omega)$ as

[1]The OSSI01 sensor (see location in Figure 1.5 and Table A.2) was used for this analysis.

Table 2.6: FIT values (%) comparing two (PI and MPC) wave elevations.

3A	75.89	**3B**	83.28	**3C**	77.66	**3, avg.**	78.94
5A	86.33	**5B**	77.7	**5C**	85.81	**5, avg.**	83.28
8A	75.19	**8B**	78.68	**8C**	83.05	**8, avg.**	78.97
9A	93.16	**9B**	92.26	**9C**	82.79	**9, avg.**	89.40

Table 2.7: Spectral moments from wave spectra for PI and MPC tests.

Test Case	**Controller**	$m_{-1} \times 10^{-3}$ [m^{2}s]	$m_0 \times 10^{-3}$ [m^2]	$m_1 \times 10^{-3}$ [m^2/s]
3B	**PI**	5.860	2.287	2.255
	MPC	5.888	2.307	2.554
5B	**PI**	11.72	4.650	4.096
	MPC	11.89	4.896	4.291
8B	**PI**	8.992	2.580	3.071
	MPC	8.143	2.159	2.063
9B	**PI**	12.34	4.096	2.852
	MPC	12.27	4.070	2.690

$$m_n = \int \omega^n S(\omega) d\omega, \tag{2.12}$$

where n is an integer. From Table 2.7, it is again clear that the two #3B wave cases match each other quite closely.

In the middle plot in Figure 2.2, the time history of the velocities of the buoy obtained by the PI controller and the MPC in case #3B is depicted. Since the control signal of the MPC fairly mimics the PI control signal and the two controllers were influenced by the same wave force, the FIT values of the resulting velocities are also similar (70.22%). The lower plot in Figure 2.2 displays the time history of the power captured by using the PI controller and the MPC. The FIT is calculated as 56.95% which is lower than the FIT values of the control force or the velocity. However, this result is close to the product of $FIT_F * FIT_v = 50.9\%$. It seems that the discrepancy between the PI and MPC signals is amplified by the time delay (˜0.09 sec) in the control and velocity signals. The average power captures by the PI controller and the MPC are -4.90 W and -5.18 W, respectively.

Figure 2.4 depicts the power spectral density obtained by the PI controller and the MPC for test case #3B. As seen from the figure, both controllers share a similar shape but a little more power is captured by the MPC over the frequencies as indicated by the higher average power capture (-5.18 W) than the PI controller (-4.90 W).

The probability density functions (PDFs) for the control commands, velocities, and power captures calculated by the PI controller and the MPC are presented in Figure 2.5. The area under each curve is set to 1, and the number of bins is selected as 100 for each plot, where the total number of samples is 300,000. As can be seen from the figure, the two curves obtained from the PI

controller and the MPC overlap with each other on the whole. Also, the PDFs for the control force and the velocity are relatively symmetric about zero while the PDFs for the power distribution is left-tailed. This fact is more clearly seen in Figure 2.6 which plots the 2D-PDFs. In the upper figures, the x-axis is the control force, the y-axis is the velocity, and the z-axis is the number of samples that fall into each bin (the bin number is 100 along each of the 2 dimensions). As can be seen, the PDFs are centered about zero. The lower figures are the projection of the upper ones onto the xy-plane. The electrical-power contour levels are also displayed, where the positive power level is specified as black numbers and the negative power level as red numbers. One can clearly see that the distributions are skewed toward negative power producing regions.

The complementary cumulative density functions (CCDFs), sometimes also referred to as survival functions, are also provided in Figure 2.7 for reference. For example, referring to the lower plot in Figure 2.7, at CCDF(x) = 0.1 on the y-axis, the corresponding power is about 20 W on the x-axis (for the negative part), which means that the (negative) power exceeds 20 W for 10% of the time.

In the same way as discussed previously for the #3B case, Figure 2.8-2.13, Figure 2.14-2.19, and Figure 2.20-2.25, present the results for test cases #5B, #8B, and #9B, respectively. The agreement in these cases is better than that seen in #3B.

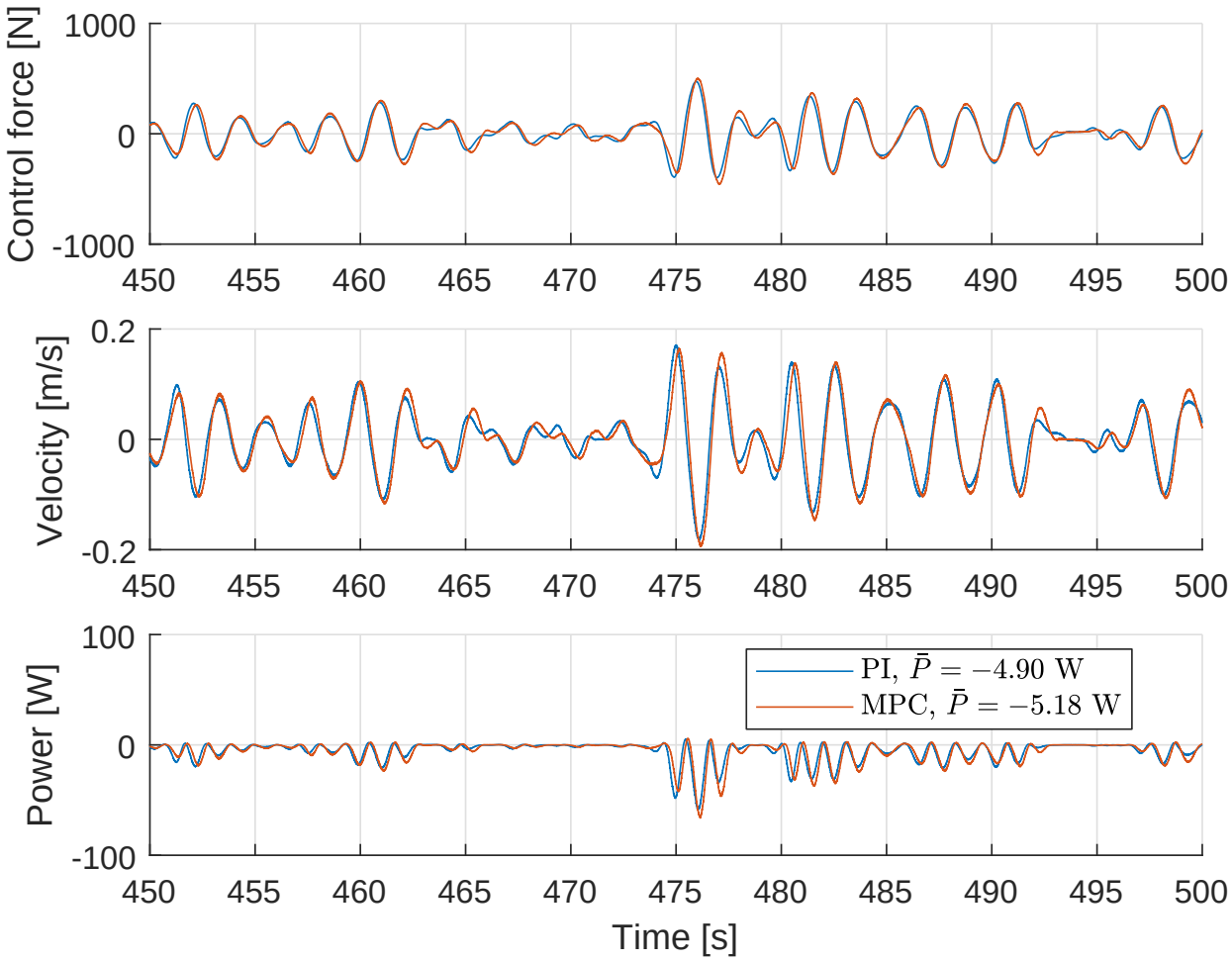


Figure 2.2: Control forces, velocities, and power obtained by PI controller and MPC for test case #3B.

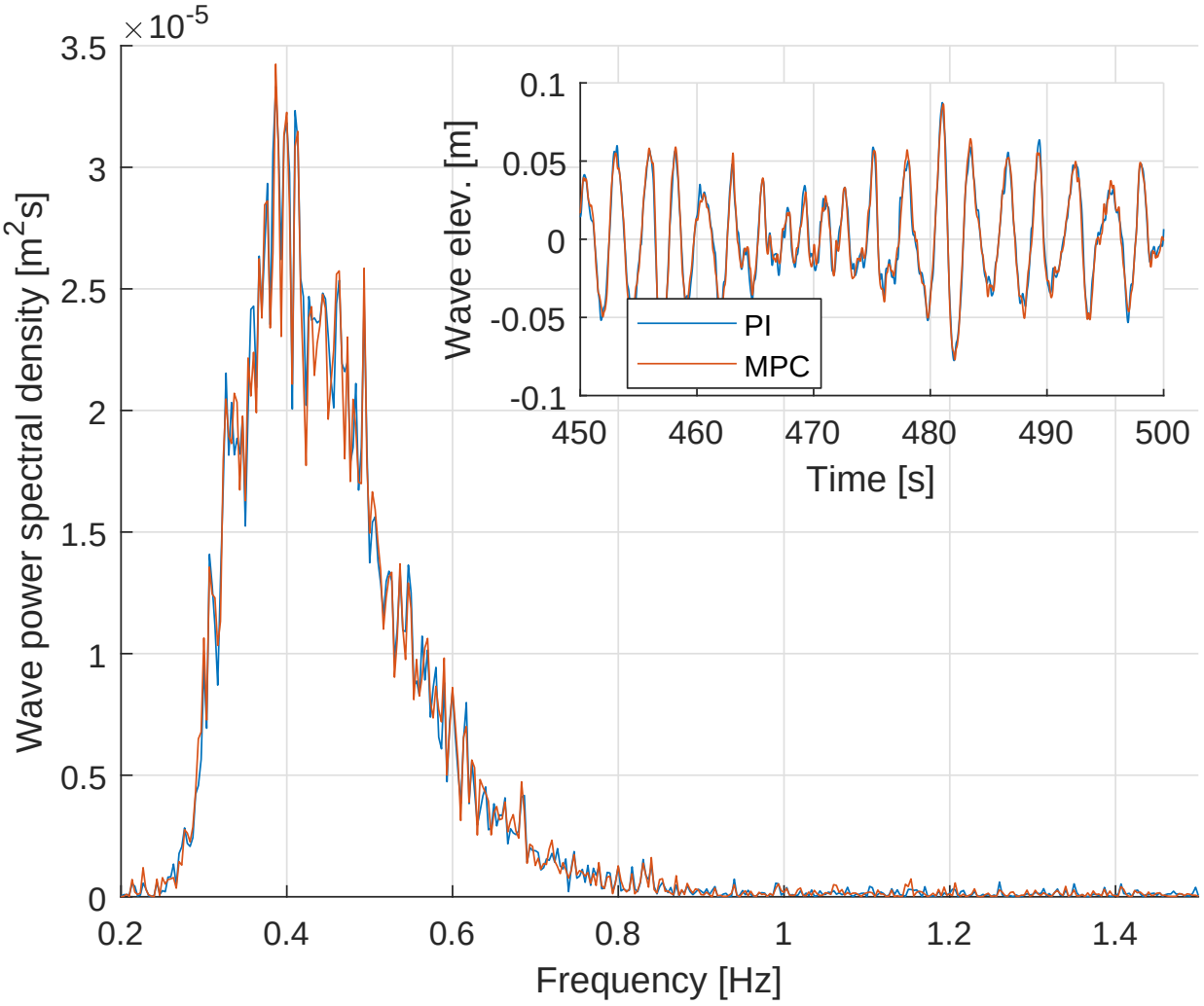


Figure 2.3: Wave elevation in frequency domain and in time domain (inner box) when PI controller and MPC were tested for test case #3B.

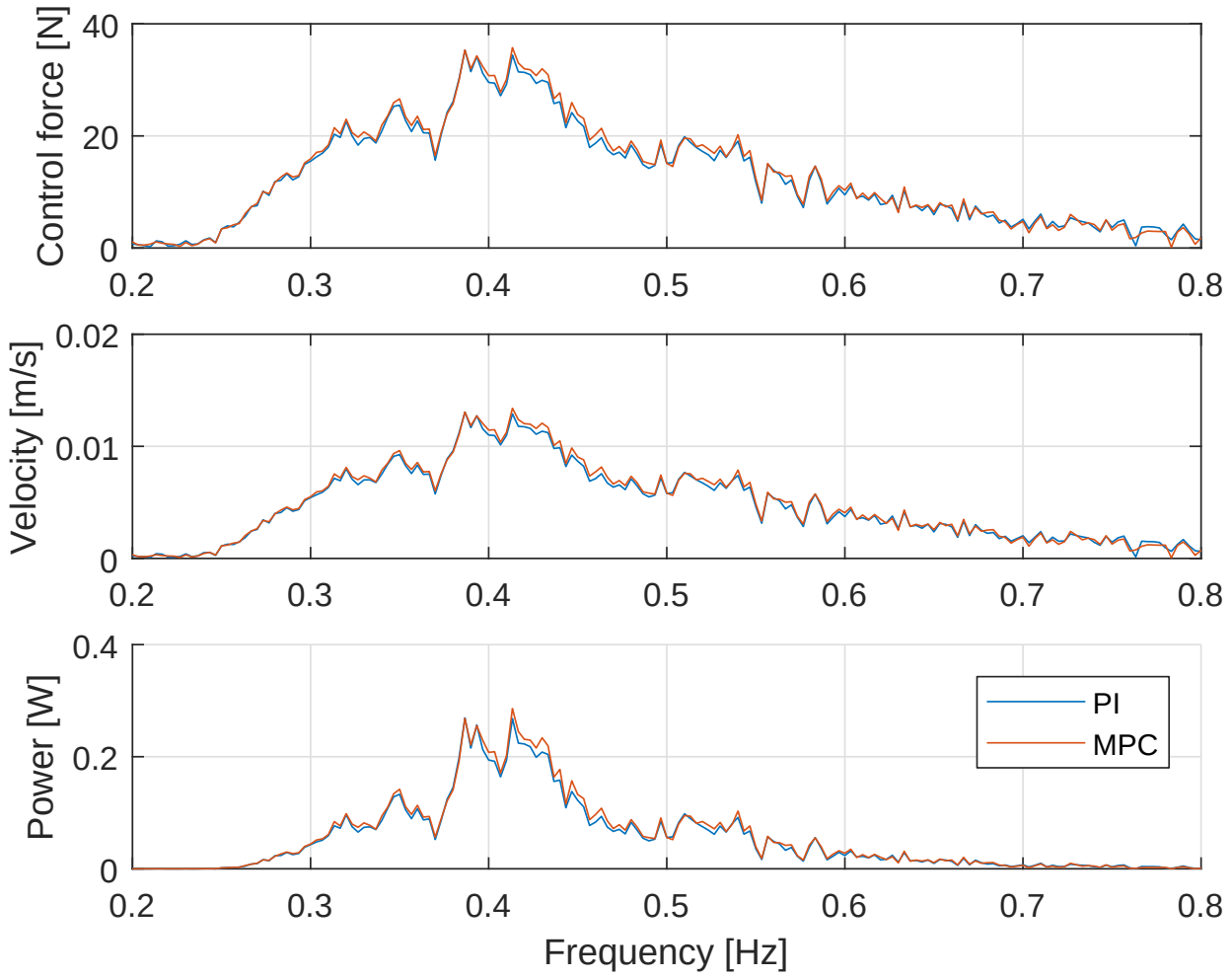


Figure 2.4: Power spectral density obtained by PI controller and MPC for test case #3B.

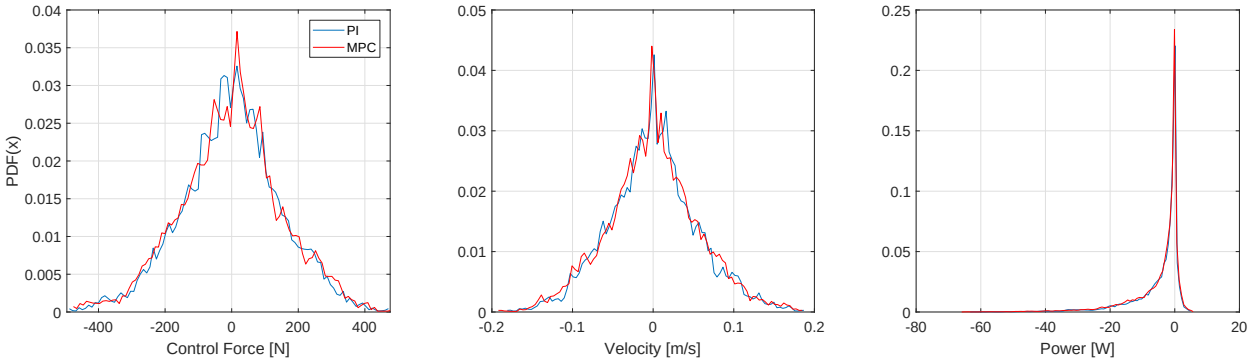


Figure 2.5: PDFs for control commands, velocities, and power calculated by PI controller and MPC for test case #3B.

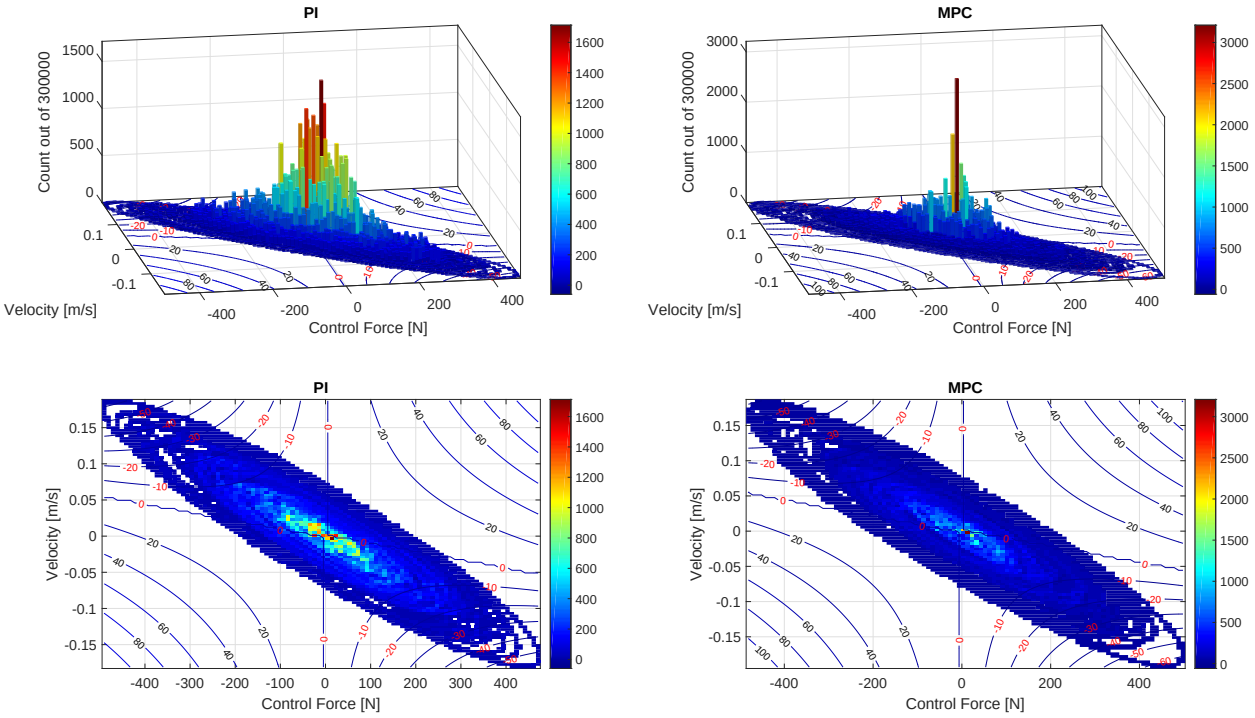


Figure 2.6: 2D-PDFs for test case #3B. (upper) 3D-view, (lower) 2D-projections.

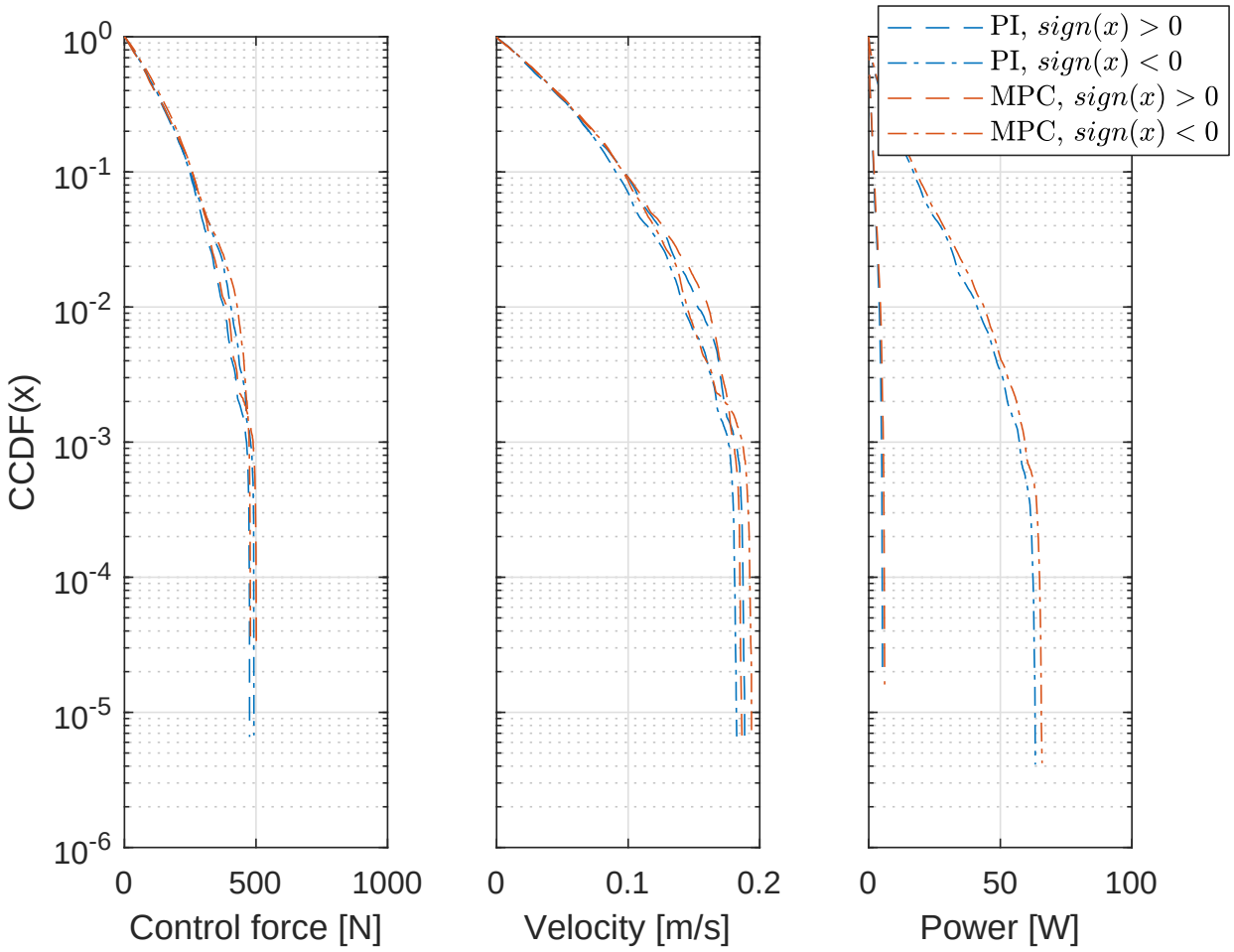


Figure 2.7: CCDFs for control commands, velocities, and power calculated by PI controller and MPC for test case #3B.

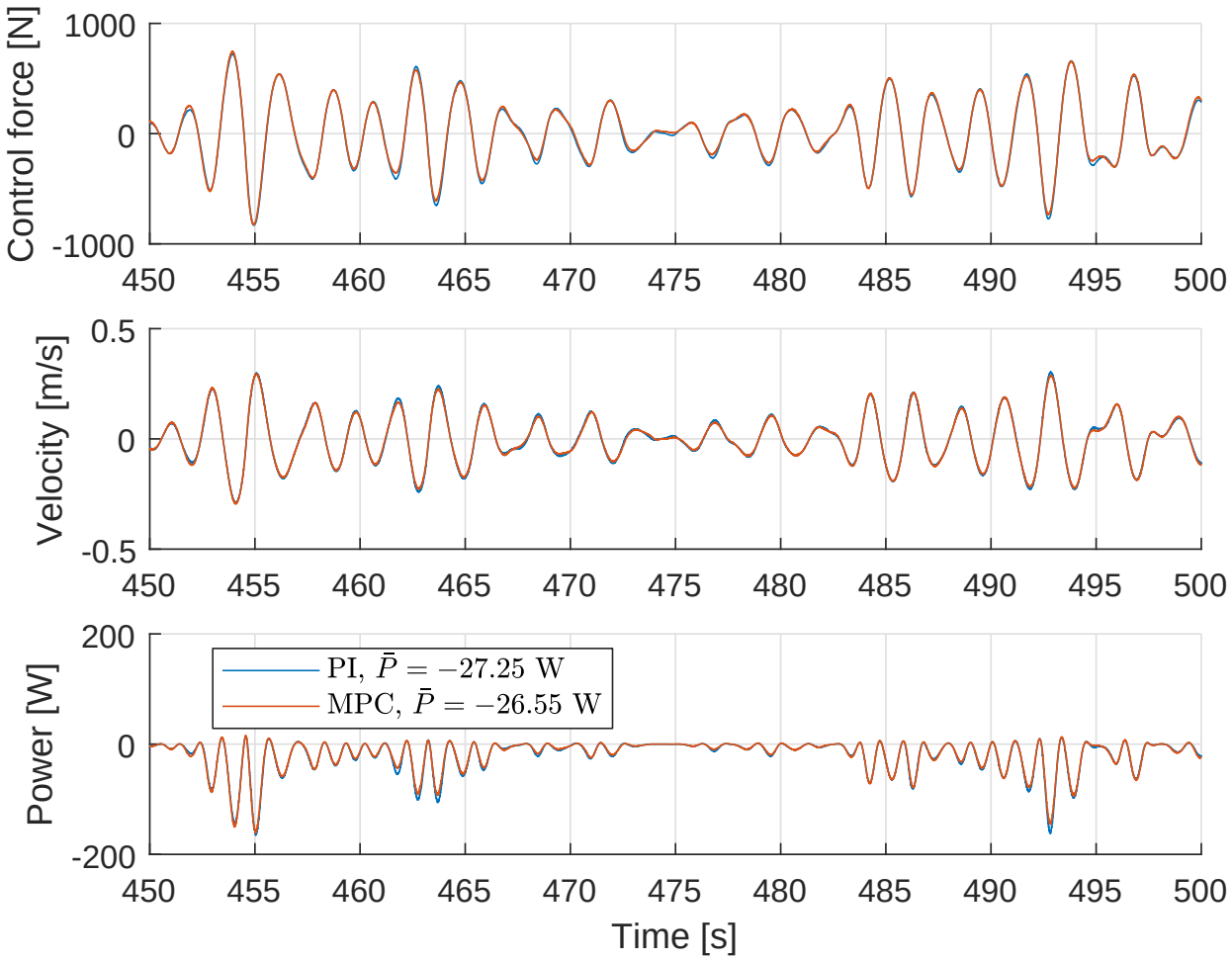


Figure 2.8: Control forces, velocities, and power captures obtained by PI controller and MPC for test case #5B.

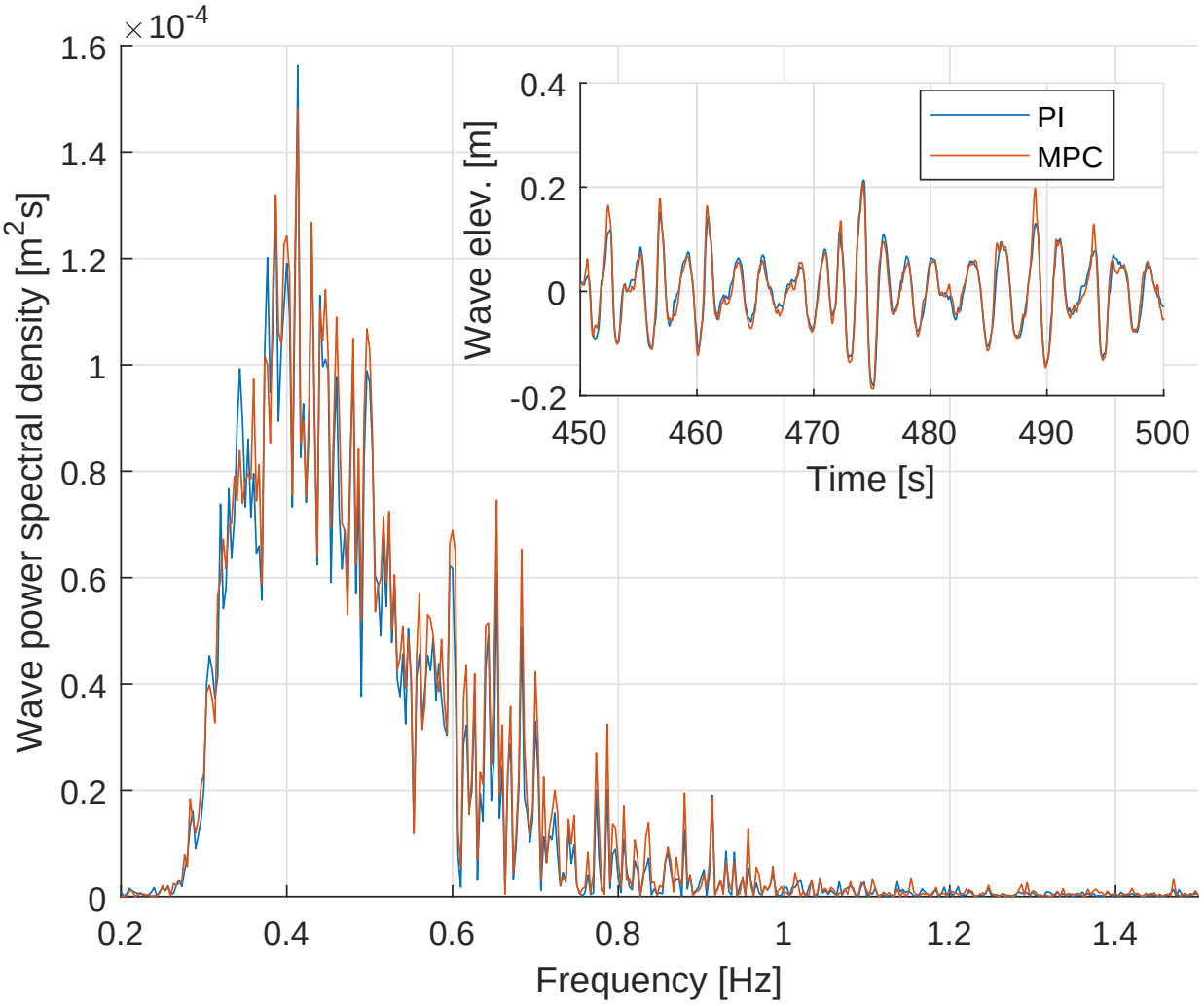


Figure 2.9: Wave elevation in frequency domain and in time domain (inner box) when PI controller and MPC were tested for test case #5B.

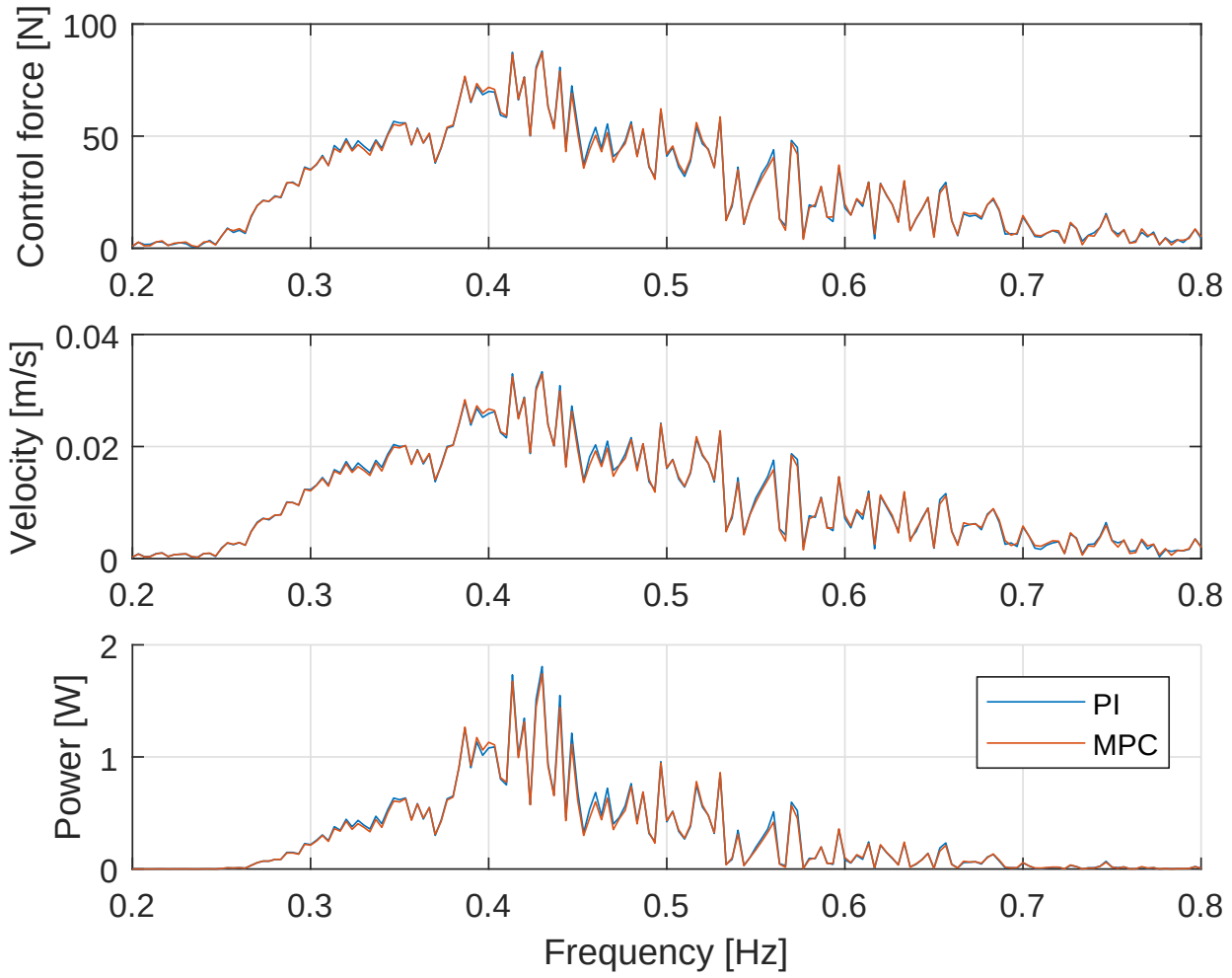


Figure 2.10: Power spectral density obtained by PI controller and MPC for test case #5B.

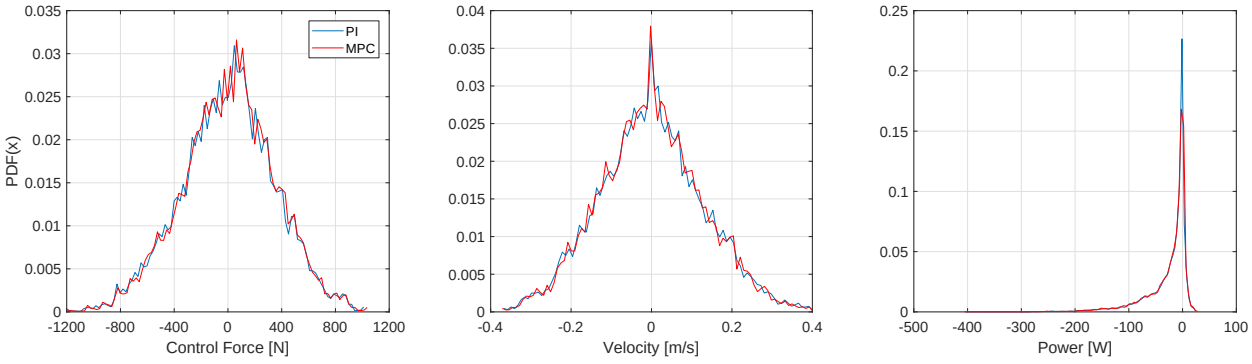


Figure 2.11: PDFs for control commands, velocities, and power calculated by PI controller and MPC for test case #5B.

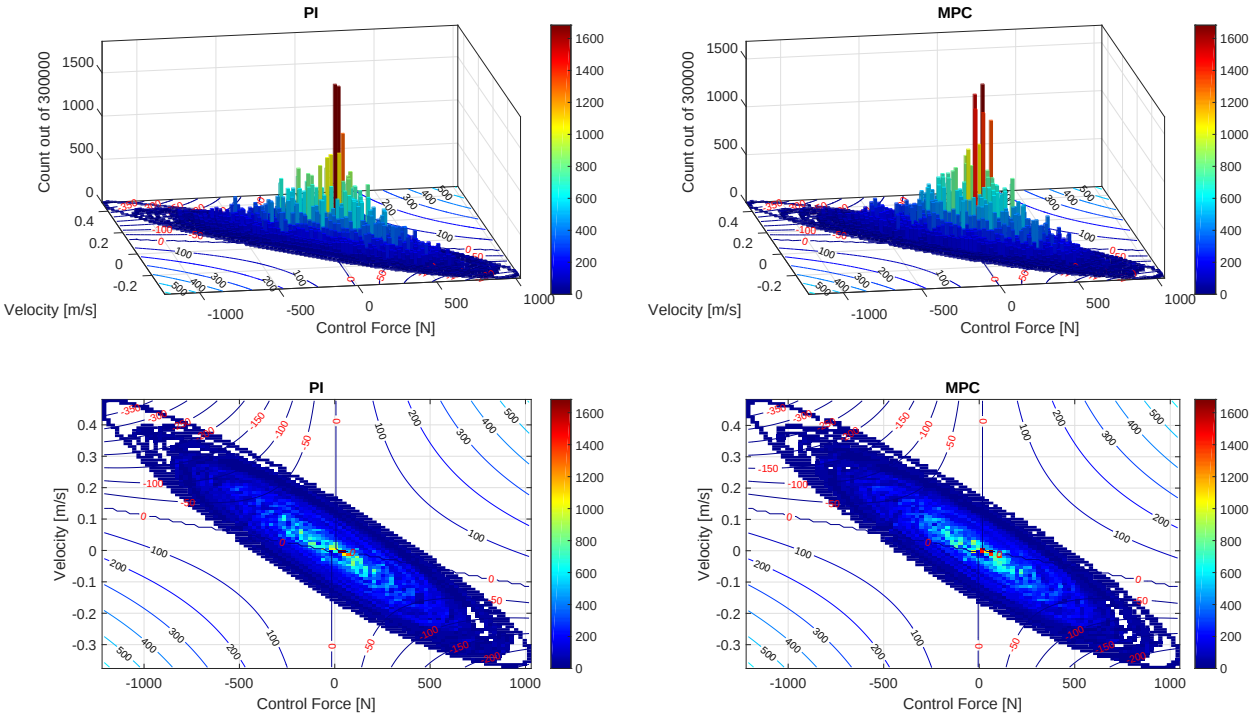


Figure 2.12: 2D-PDFs for test case #5B. (upper) 3D-view, (lower) 2D-projections.

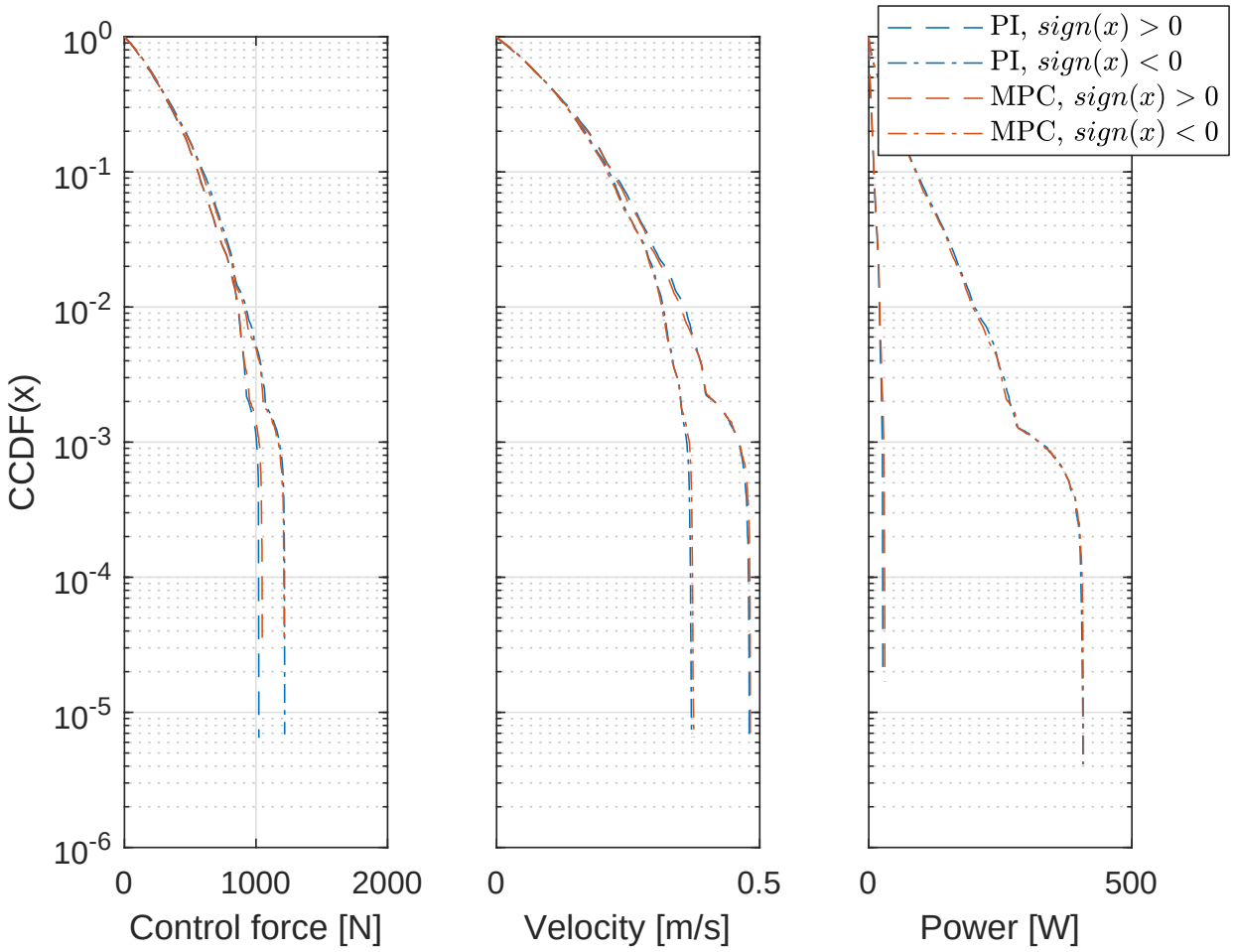


Figure 2.13: CCDFs for control commands, velocities, and power captures calculated by PI controller and MPC for test case #5B.

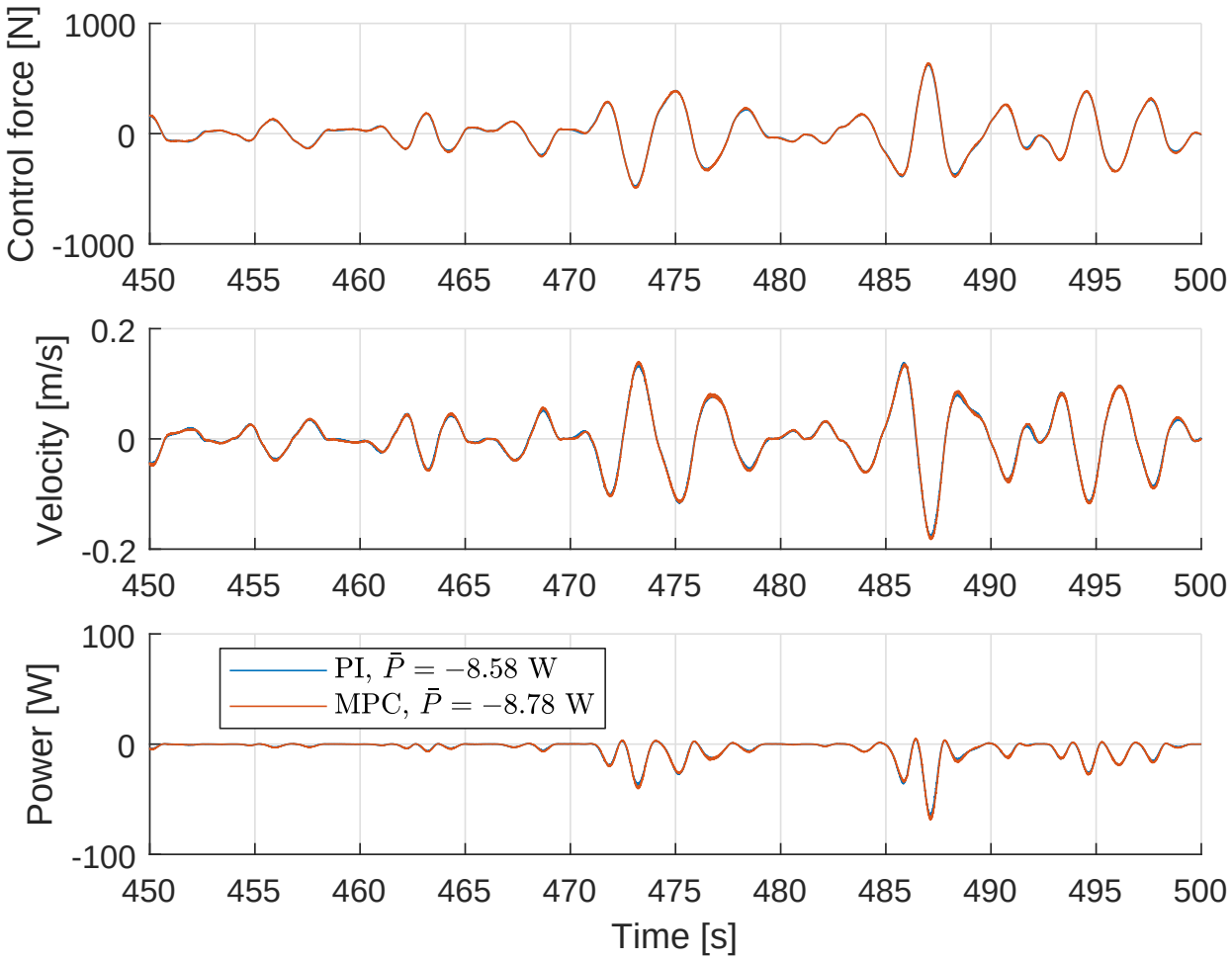


Figure 2.14: Control forces, velocities, and power captures obtained by PI controller and MPC for test case #8B.

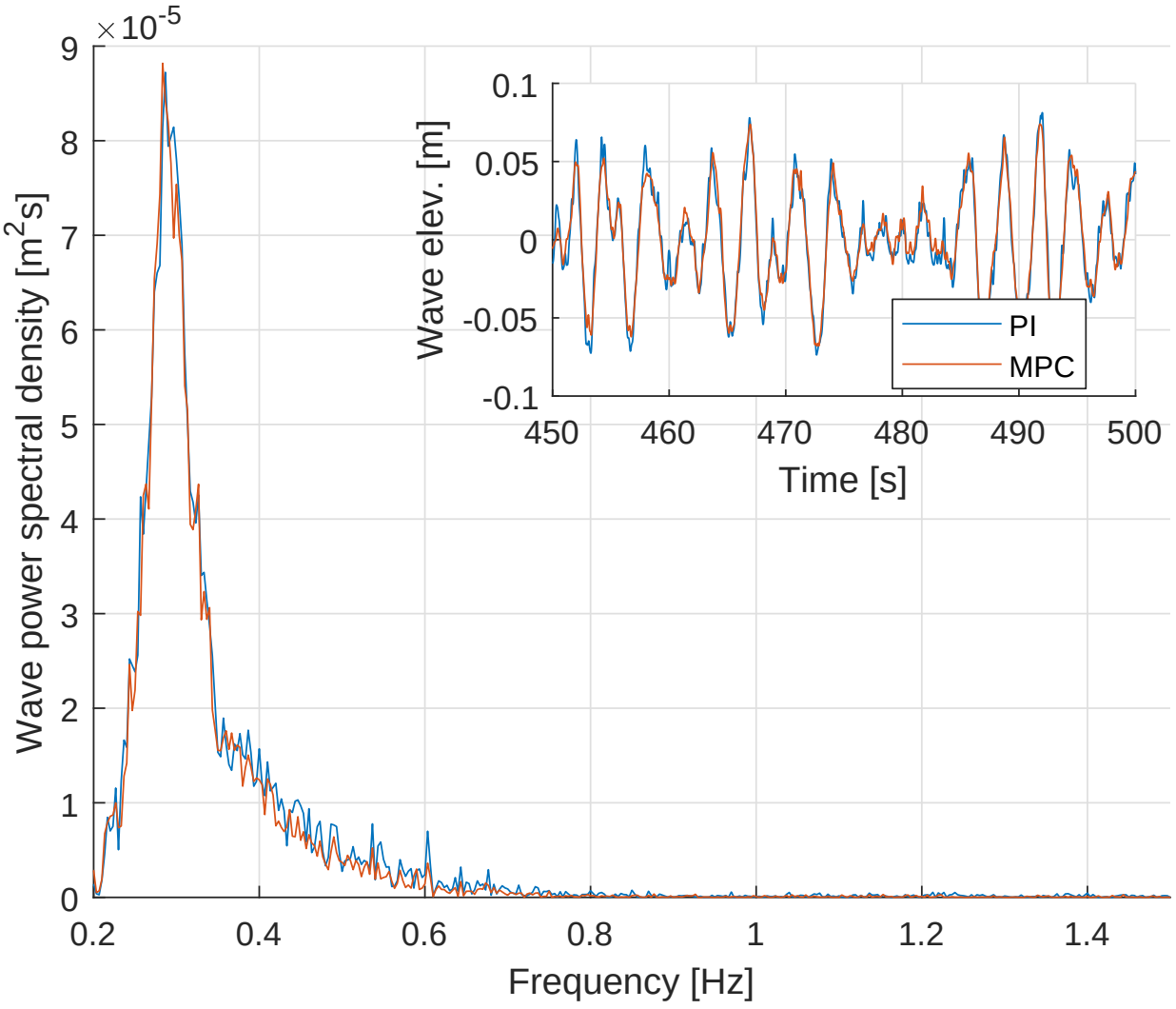


Figure 2.15: Wave elevation in frequency domain and in time domain (inner box) when PI controller and MPC were tested for test case #8B.

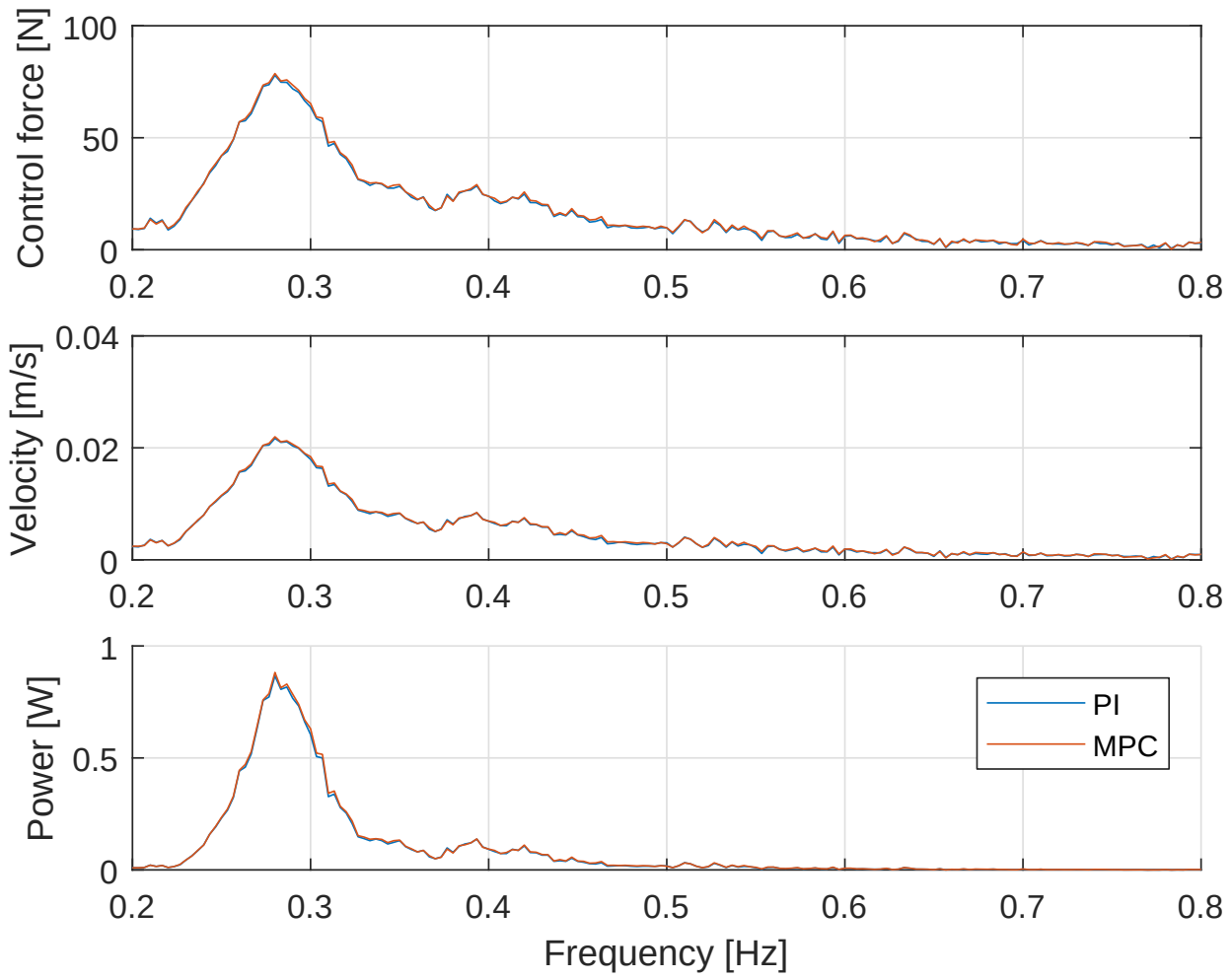


Figure 2.16: Power spectral density obtained by PI controller and MPC for test case #8B.

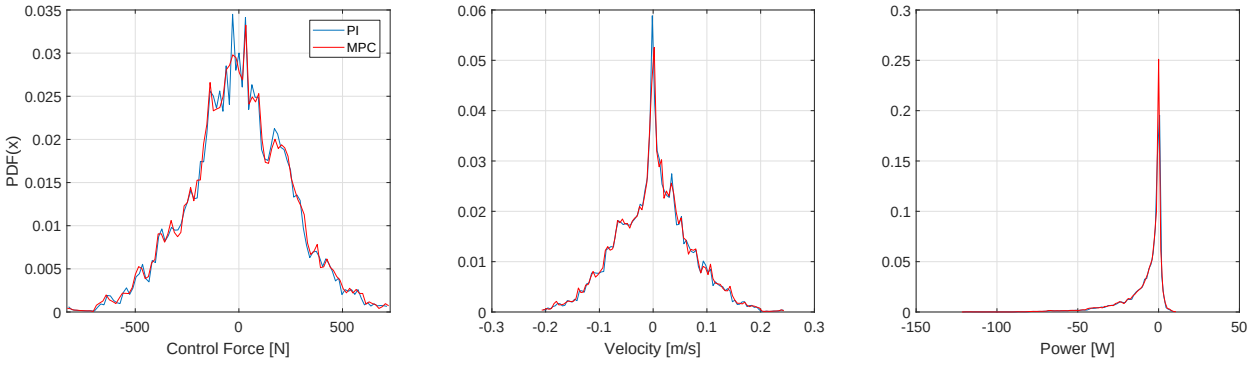


Figure 2.17: PDFs for control commands, velocities, and power calculated by PI controller and MPC for test case #8B.

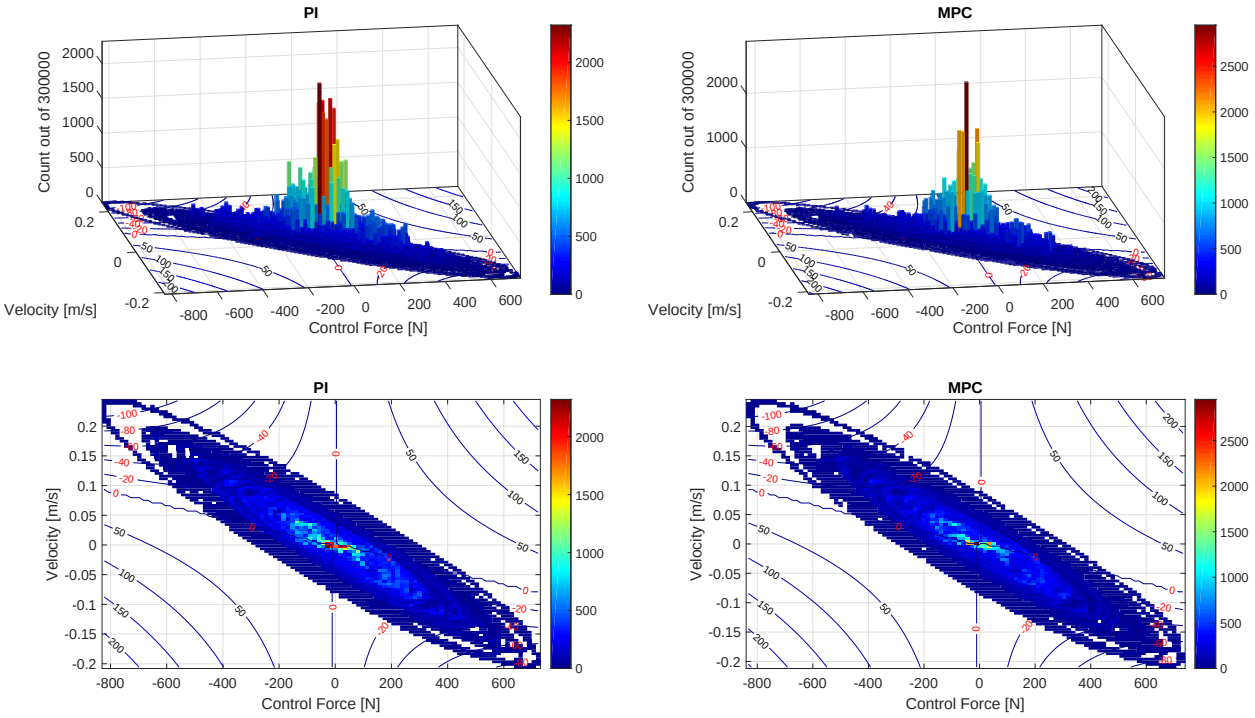


Figure 2.18: 2D-PDFs for test case #8B. (upper) 3D-view, (lower) 2D-projections.

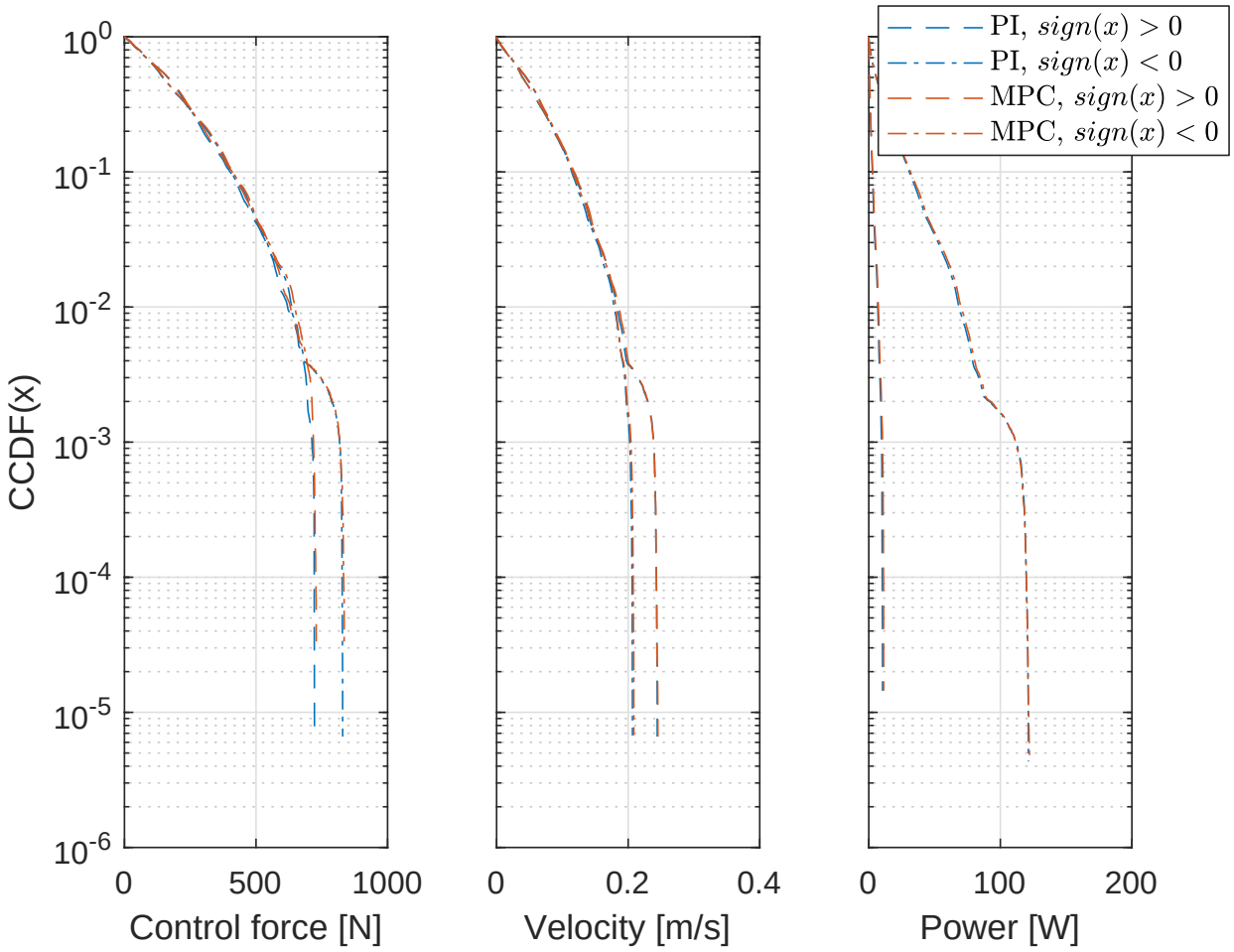


Figure 2.19: CCDFs for control commands, velocities, and power captures calculated by PI controller and MPC for test case #8B.

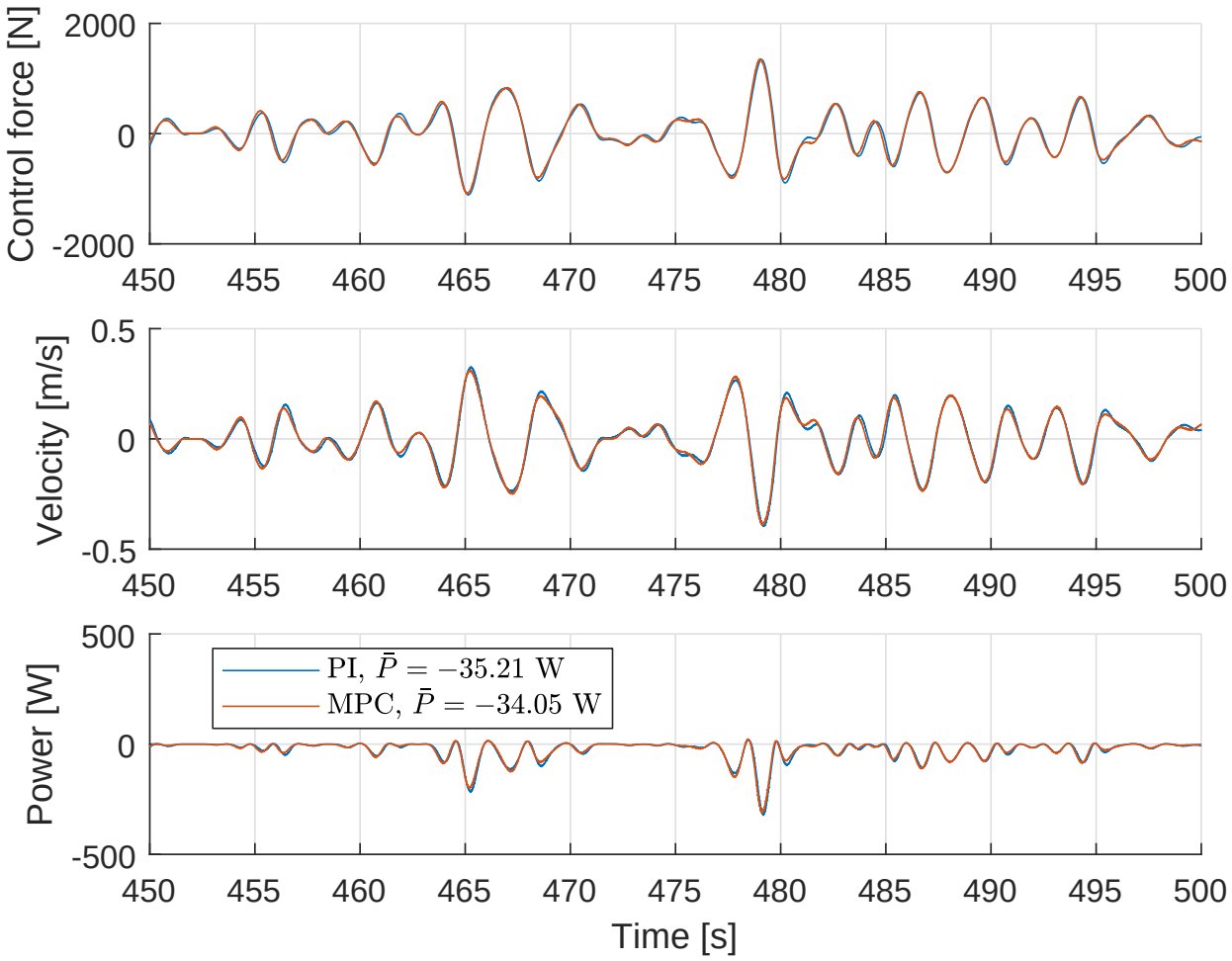


Figure 2.20: Control forces, velocities, and power captures obtained by PI controller and MPC for test case #9B.

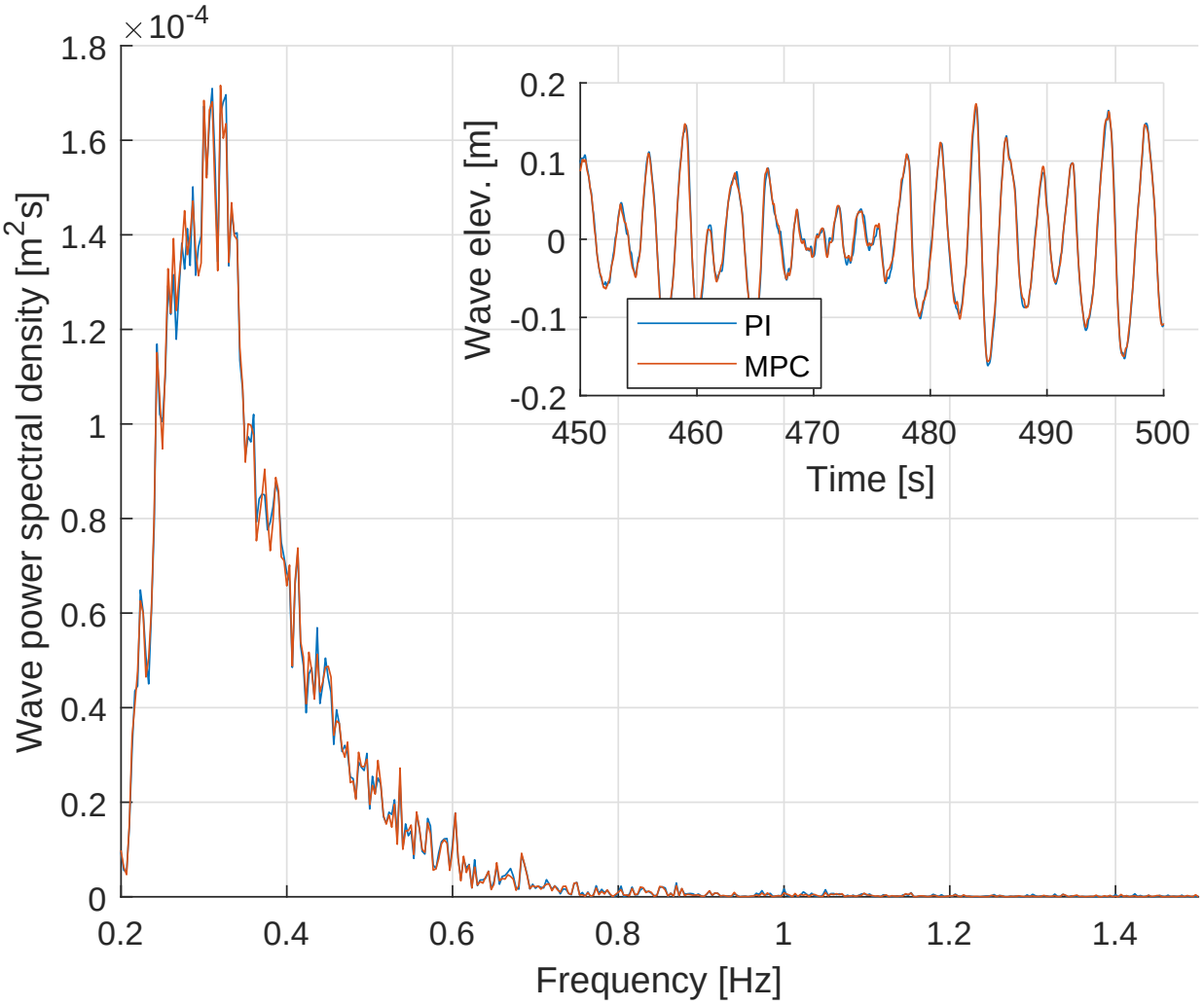


Figure 2.21: Wave elevation in frequency domain and in time domain (inner box) when PI controller and MPC were tested for test case #9B.

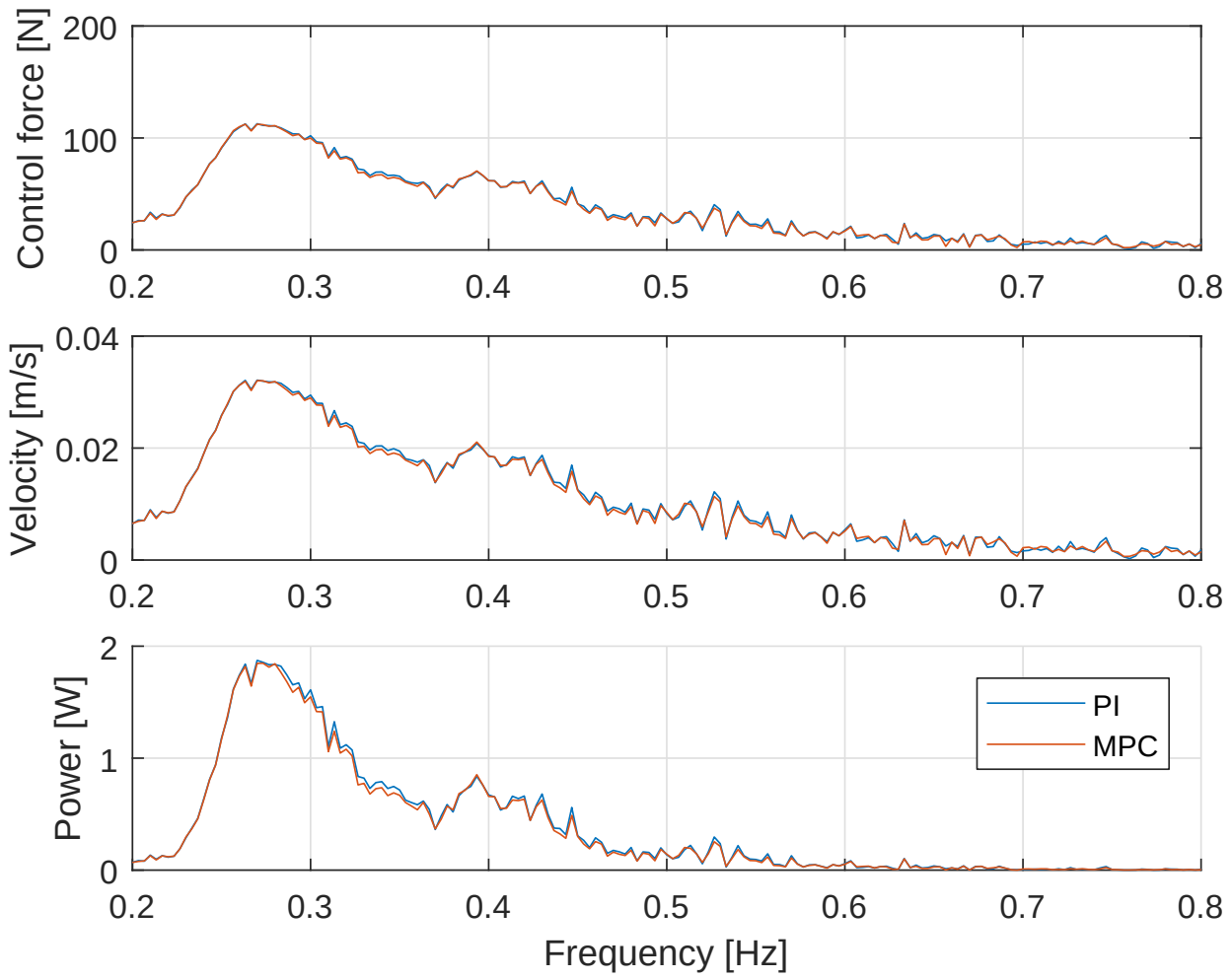


Figure 2.22: Power spectral density obtained by PI controller and MPC for test case #9B.

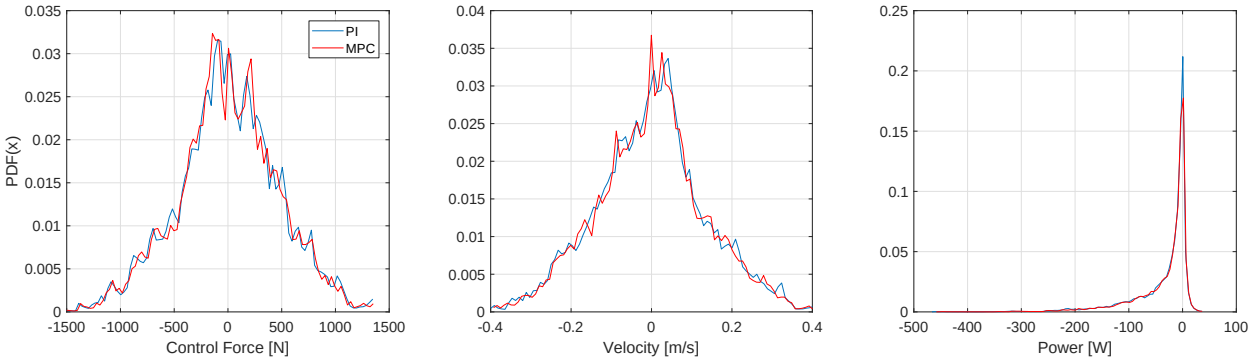


Figure 2.23: PDFs for control commands, velocities, and power calculated by PI controller and MPC for test case #9B.

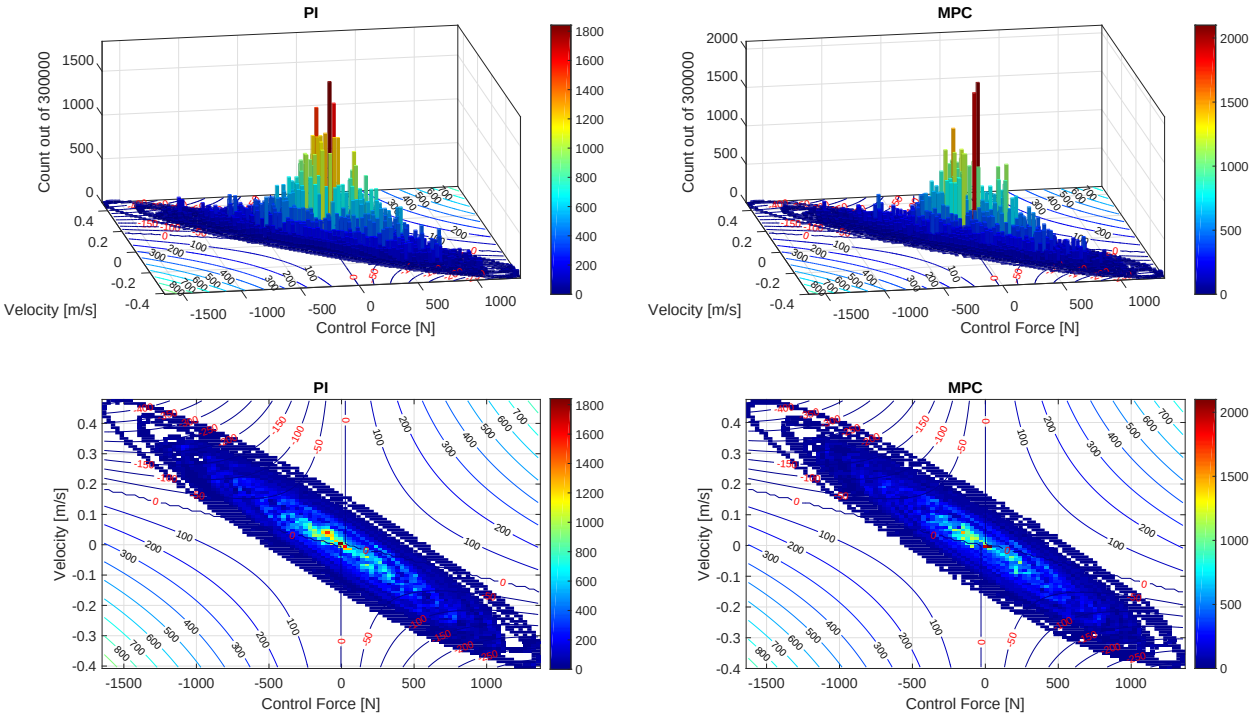


Figure 2.24: 2D-PDFs for test case #9B. (upper) 3D-view, (lower) 2D-projections.

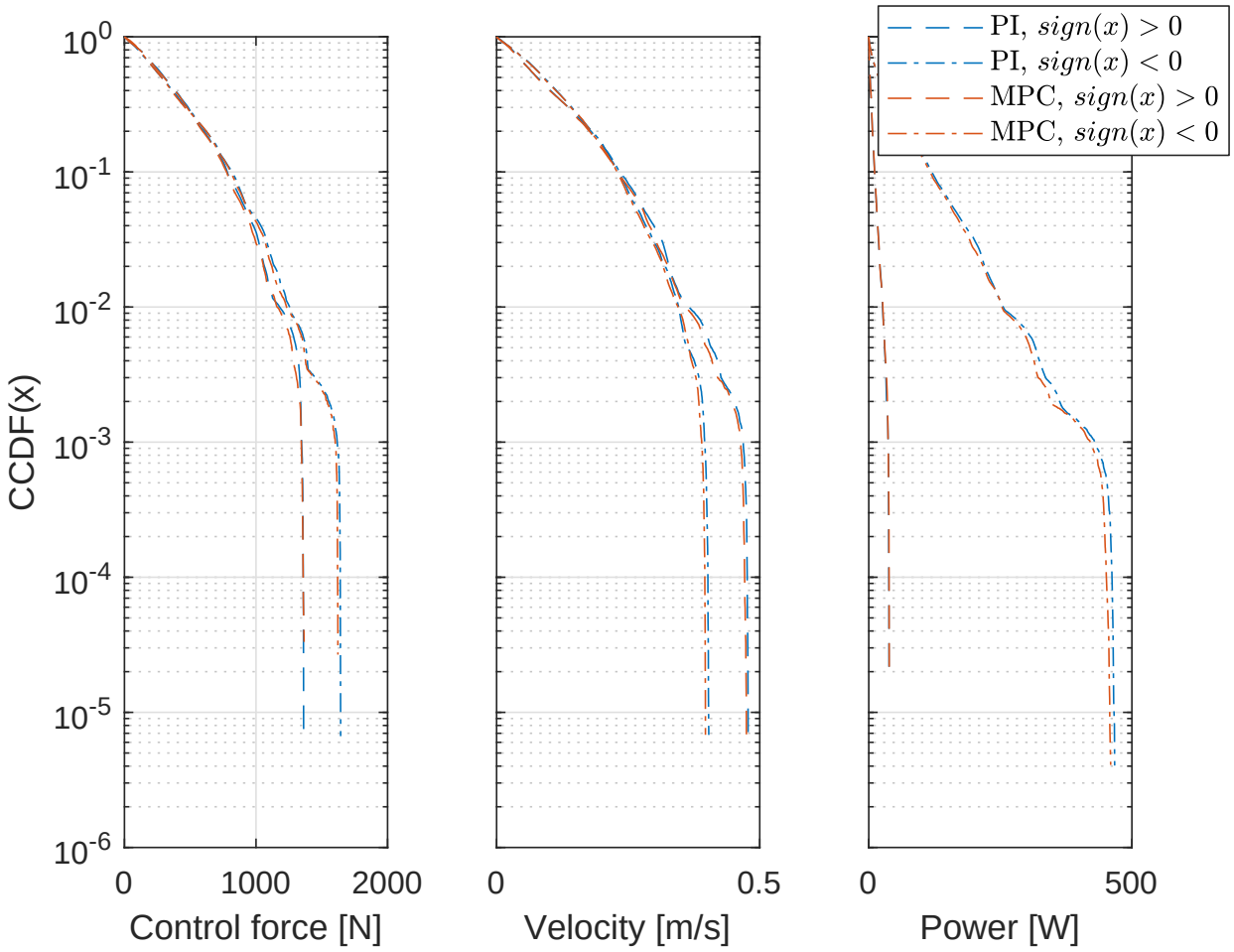


Figure 2.25: CCDFs for control commands, velocities, and power captures calculated by PI controller and MPC for test case #9B.

Up to this point, no constraints have been considered. One strength of using MPC strategies is their capability to handle constraints. Hence, the constrained MPCs were also tested such that the control signal is saturated with given upper/lower limits and their performance is provided. It will be shown that the power captured by the constrained MPC is reduced only by a small amount when compared with the unconstrained MPC even though a short horizon $N = 2$ is employed.

First, the test case #3B is considered. Let U_{max} be the maximum control magnitude that can be reached by the actuator. Since the maximum magnitude of the control force when the unconstrained MPC is applied is found to be about 500 N for case #3B from the top-most plot in Figure 2.2, let us constrain the system by forcing U_{max} to be 450, 300, and 150. The top-most plot in Figure 2.26 shows the control forces obtained by the unconstrained and constrained ($U_{max} = 450, 300, 150$) MPCs. It is obvious that the control forces successfully stay within the desired range. In the middle plot in Figure 2.26, the resulting velocities are plotted for the unconstrained and constrained MPCs. With the constrained MPCs, the magnitude of the resulting velocity is greater than the unconstrained case because the saturated control force cannot fully regulate the velocity. The lower plot in Figure 2.26 displays the time history of the power captured by using the unconstrained and constrained MPCs. As expected, when the constraint is active, less power is captured.

However, Table 2.8 indicates that the power captured by the constrained MPCs is worse only by a small percentage when compared with the unconstrained MPC. In the Table 2.8, the average power capture obtained using the unconstrained and constrained MPCs is listed for test cases #3, 5, 8, and 9 and the power loss in percentage is also presented in the parenthesis compared with the unconstrained case. U_{max} values for the constrained cases 1, 2, and 3 are shown in Table 2.9 for each test case. For example, consider the test case #3B and when the control force is constrained with $U_{max} = 300$ (Constrained Case 2) which corresponds to 60% of the maximum control magnitude obtained when the unconstrained MPC is applied, the power capture is -5.15 W, which is worse only by 0.60% compared with the unconstrained case (-5.18 W).

In the same way, in Figure 2.27, Figure 2.28, and Figure 2.29, the control forces, velocities, and power captures obtained by the unconstrained and constrained MPCs are provided for the test cases #5B, #8B, and #9B, respectively. In every case, the control output saturation constraint is well satisfied.

Table 2.8: Power captured (W and % change from unconstrained) by using unconstrained and constrained MPCs.

Test Case	**Unconstrained**	**Constrained Case 1**	**Constrained Case 2**	**Constrained Case 3**
3A	-5.07	-5.05 (0.40%)	-5.00 (1.38%)	-4.33 (14.60%)
3B	-5.18	-5.18 (0.00%)	-5.15 (0.60%)	-4.40 (15.06%)
3C	-5.17	-5.15 (0.39%)	-5.07 (1.93%)	-4.25 (17.80%)
5A	-26.51	-27.12 (-2.30%)	-26.44 (0.26%)	-20.80 (21.54%)
5B	-26.51	-26.50 (0.038%)	-25.85 (2.49%)	-20.61 (22.26%)
5C	-26.85	-27.32 (-1.75%)	-26.55 (1.12%)	-20.07 (25.25%)
8A	-8.90	-8.89 (0.11%)	-8.50 (4.49%)	-6.62 (25.62%)
8B	-8.78	-8.76 (0.23%)	-8.49 (3.30%)	-6.69 (23.80%)
8C	-8.69	-8.70 (-0.12%)	-8.45 (2.76%)	-6.65 (23.48%)
9A	-33.76	-33.51 (0.74%)	-31.47 (6.78%)	-22.94 (32.05%)
9B	-34.05	-33.81 (0.70%)	-31.91 (6.28%)	-23.49 (31.01%)
9C	-32.85	-32.68 (0.52%)	-31.02 (5.57%)	-23.02 (29.92%)

Table 2.9: U_{max} values (N) for three constained cases.

Test Case	**Constrained Case 1**	**Constrained Case 2**	**Constrained Case 3**
3A, 3B, 3C	450	300	150
5A, 5B, 5C	750	500	250
8A, 8B, 8C	600	400	200
9A, 9B, 9C	900	600	300

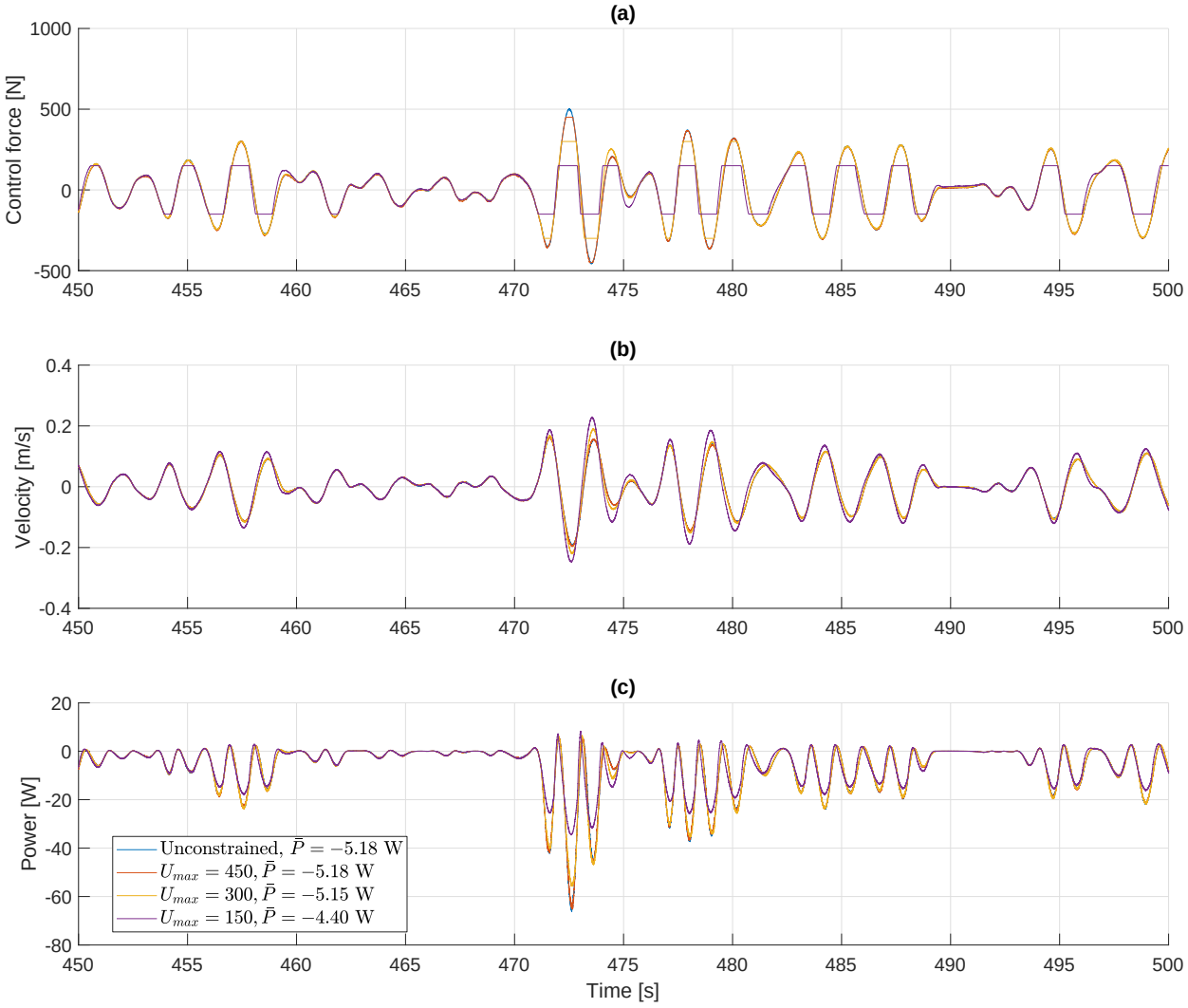


Figure 2.26: Control forces, velocities, and power obtained by unconstrained and constrained MPC for test case #3B.

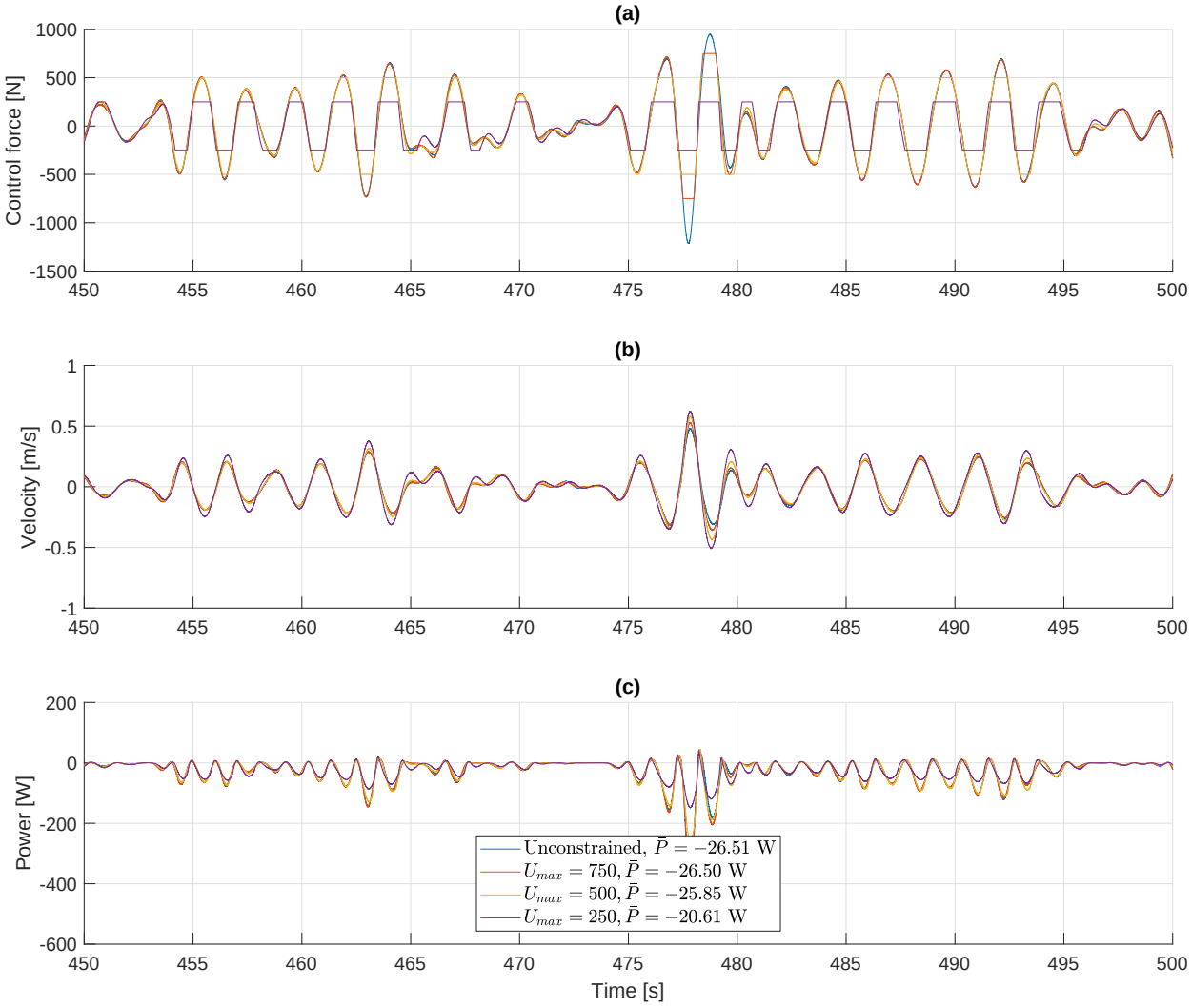


Figure 2.27: Control forces, velocities, and power obtained by unconstrained and constrained MPC for test case #5B.

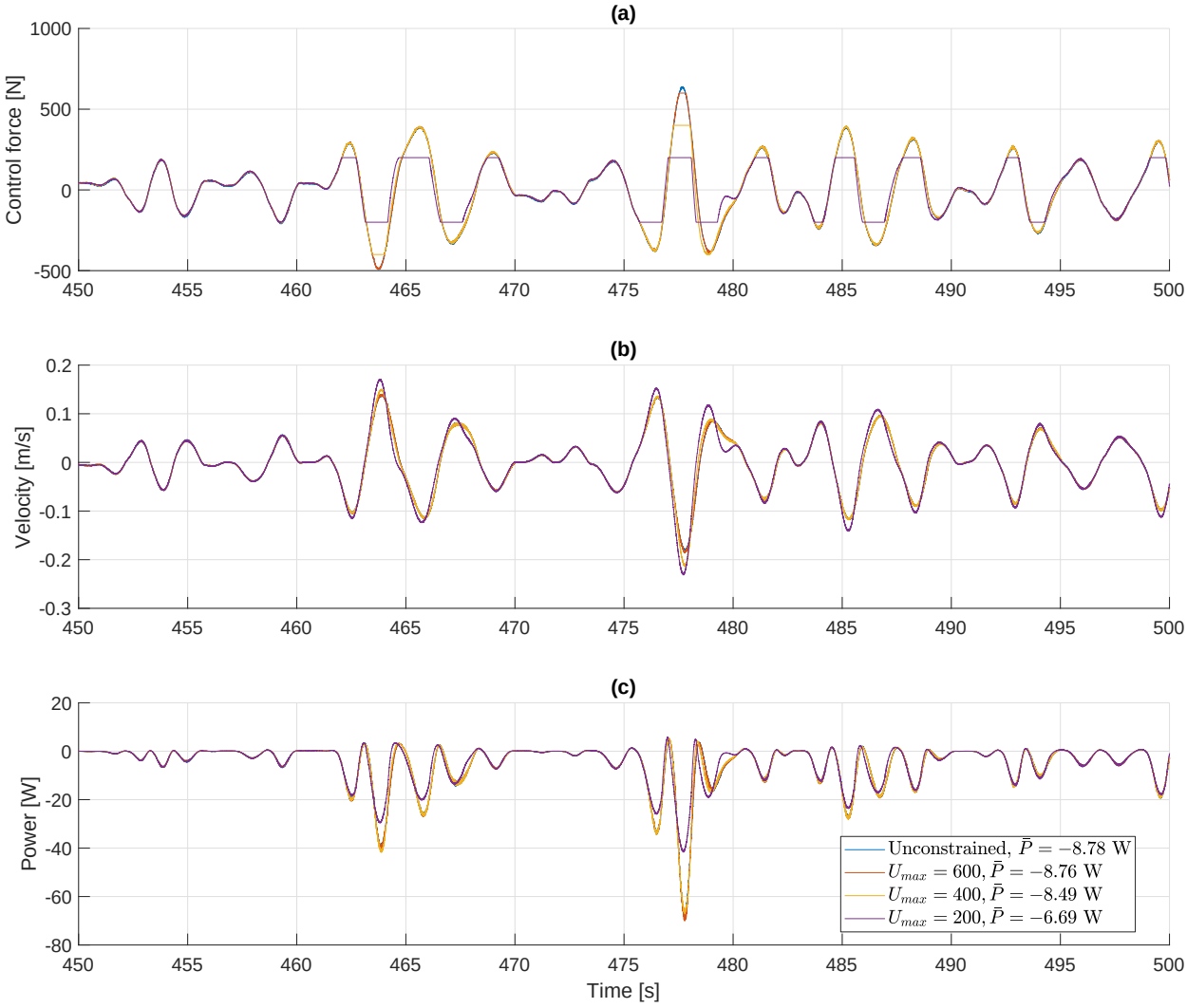


Figure 2.28: Control forces, velocities, and power obtained by unconstrained and constrained MPC for test case #8B.

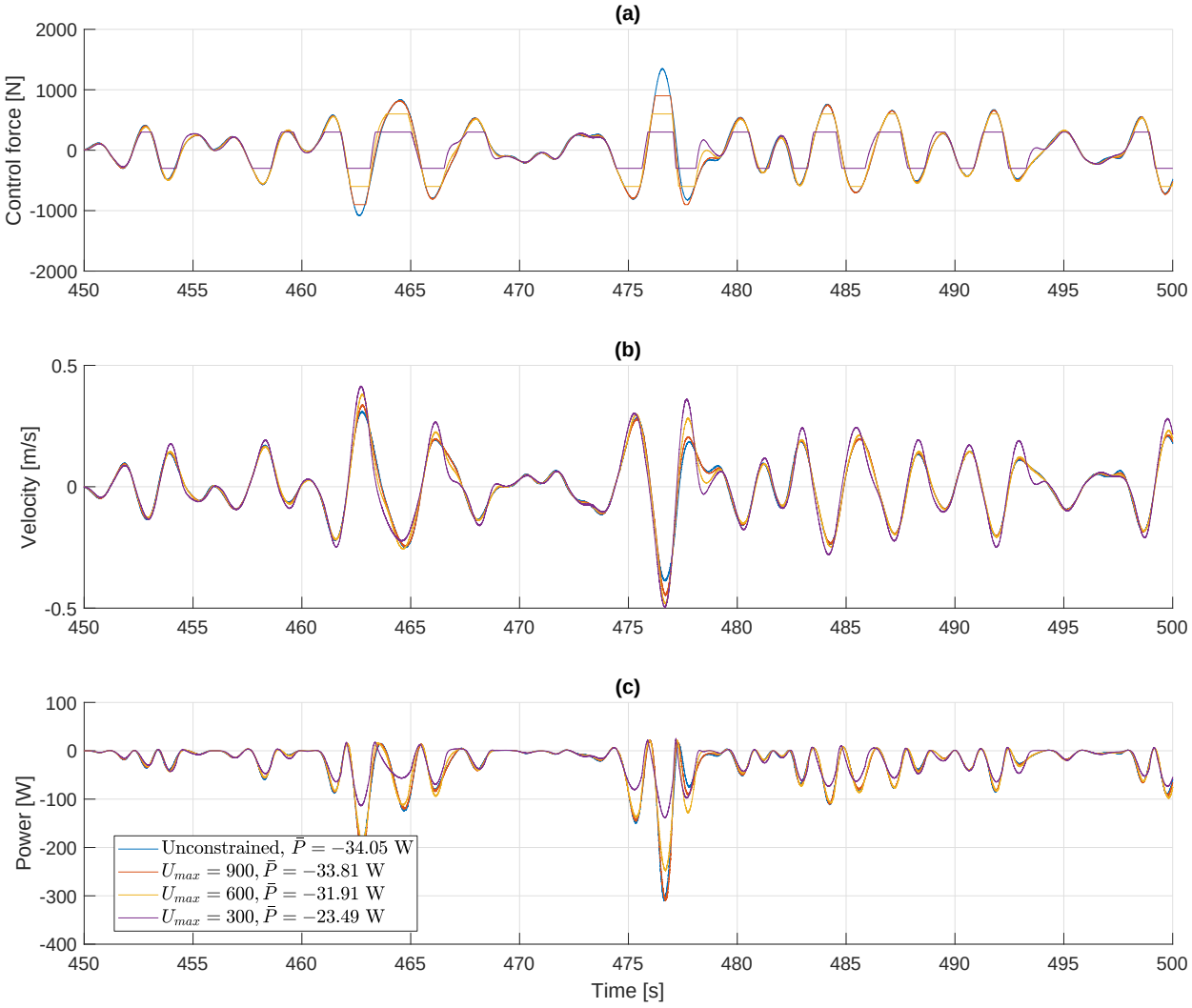


Figure 2.29: Control forces, velocities, and power obtained by unconstrained and constrained MPC for test case #9B.

2.3 Experimental results #2: Gain variation of PI controller

In this section, we consider the effect of the gain variation of the PI controller on the power absorption. Accordingly, the optimal K_I and K_P gains with which the power capture is maximized will be also estimated for future use via surface fitting. The sea states #3, 5, 7, 8, 9, 10, and 11 in Table 1.4 were tested, and for each sea state three different phase realizations (A, B, C) were introduced as in the previous section.

Tests were performed in ~ 90 minute[2] segments. Each gain setting was tested for 5 minutes during a run. The gain settings were scheduled to change in a cyclical manner. The data from the first 5 minute segment was thrown away to allow the wave basin to settle and to allow the device operators time to start the device and verify that the motion of the device resulting from the incoming waves would be reasonable for the duration of the test. For this latter reason, the gain settings predicted to excite the most motion were chosen for this first five minute segment; these gains were the largest magnitude I gain and the smallest magnitude P gain. Then next gain setting to be used was the value near the predicted optimum. The gains then spiral outward until finishing the sequence, at which point they return to the initial setting (which had previously been thrown out). Finally, the point at the beginning of the spiral is repeated to assess repeatability of these tests.

For example, let us consider the test cases #3A, 3B, and 3C. The 16 circle marker points in the left-hand plot of Figure 2.30 represent the tested K_I and K_P gains, and the corresponding power is plotted in the right-hand plot of Figure 2.30. It is seen that there is a little discrepancy between the three curves and one can expect that there would exist a convex surface that minimizes the distance error (in the sense of least squares) between the convex surface and the 48 (=16 $\times$ 3) points. This convex surface was estimated by quadratic surface fitting. More specifically, the surface was assumed to have the following form:

$$C_1X^2 + C_2XY + C_3Y^2 + C_4X + C_5Y + C_6 = 0. \tag{2.13}$$

Then, we find the coefficients C_i ($i = 1, \cdots, 6$) that minimize the least squares error and the vertex of the resulting quadratic surface (2.13) with the determined coefficients would indicate optimal K_I and K_P gains. Figure 2.31 displays the obtained quadratic surface and the vertex is located at the point (2865,-2309), that is, the optimal gains are estimated as $K_I = 2865$ and $K_P = -2309$ and the resulting power capture is -5.12 W.

Analysis of the remaining test cases was conducted in a similar fashion. Figure 2.32 shows the tested K_I and K_P gains for test cases #5, 7, 8, 9, 10, and 11, respectively, and in Figure 2.33 the corresponding power captures and the estimated quadratic surfaces are plotted. Table 2.10 lists the estimated K_I and K_P gains and the resulting power capture for each test case. Comparing with Figure 2.5, one can find that the estimated K_I and K_P gains offer marginally better power absorption, as the commanded K_I and K_P gains are close to optima.

[2]With the exception of wave case #8A, which was performed with 25 points during a 135 minute experiment.

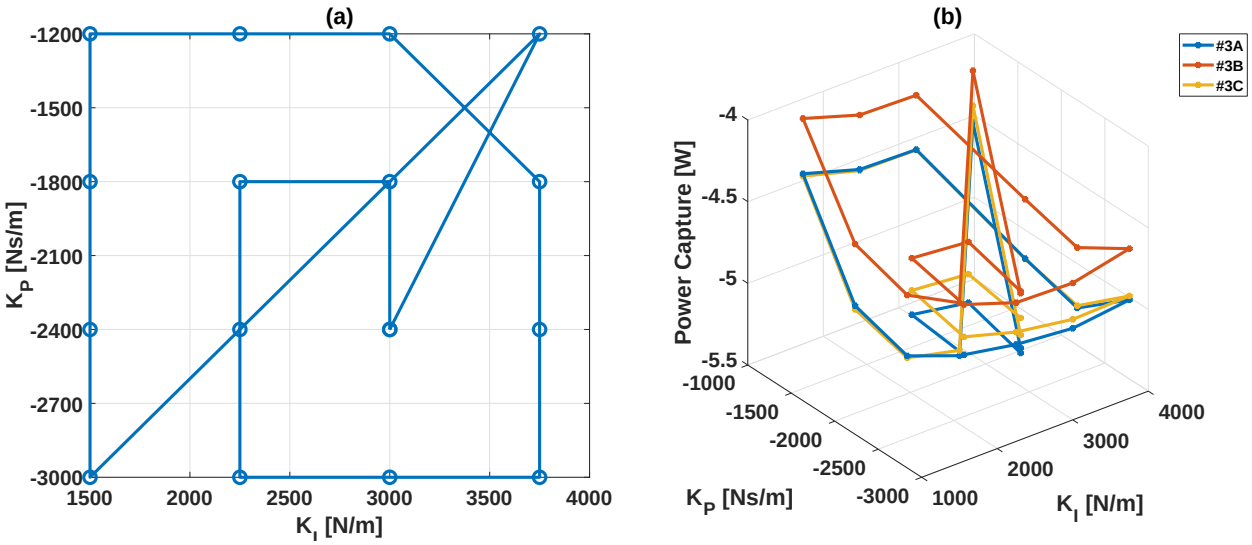


Figure 2.30: Tested K_I and K_P gains (left) and the corresponding power capture (right) for test cases #3A, 3B, and 3C.

Table 2.10: Estimated K_I and K_P gains and the resulting power capture.

Test Case	K_I $[\frac{N}{m}]$	K_P $[\frac{Ns}{m}]$	Power Capture [W]
3A, 3B, 3C	2865	-2309	-5.12
5A, 5B, 5C	2865	-2260	-27.32
7A, 7B, 7C	2522	-3073	-7.58
8A, 8B, 8C	2425	-3269	-8.66
9A, 9B, 9C	2507	-3016	-34.25
10A, 10B, 10C	2626	-3192	-39.75
11A, 11B, 11C	2374	-3126	-92.42

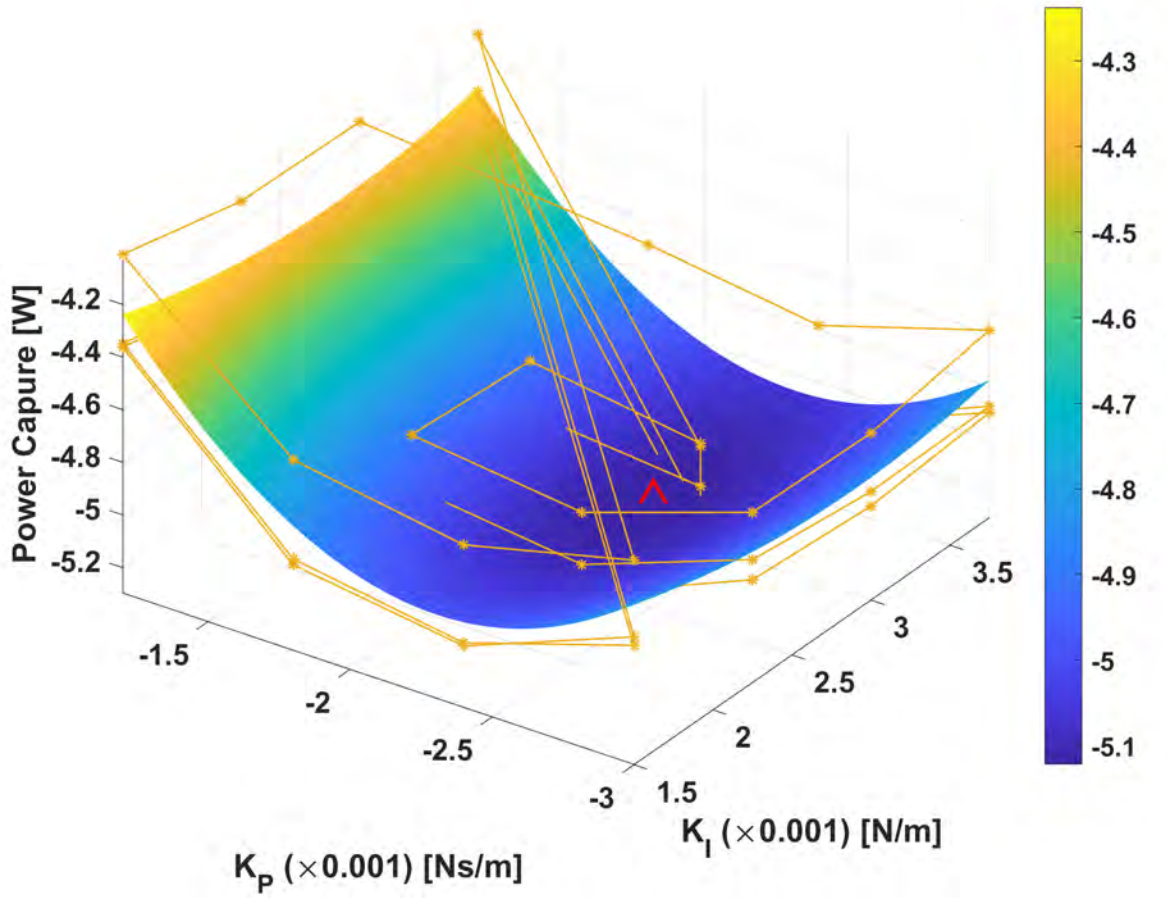


Figure 2.31: Estimated quadratic surface and optimal K_I and K_P gains located at the red $\triangle$ marker for test cases #3A, 3B, and 3C.

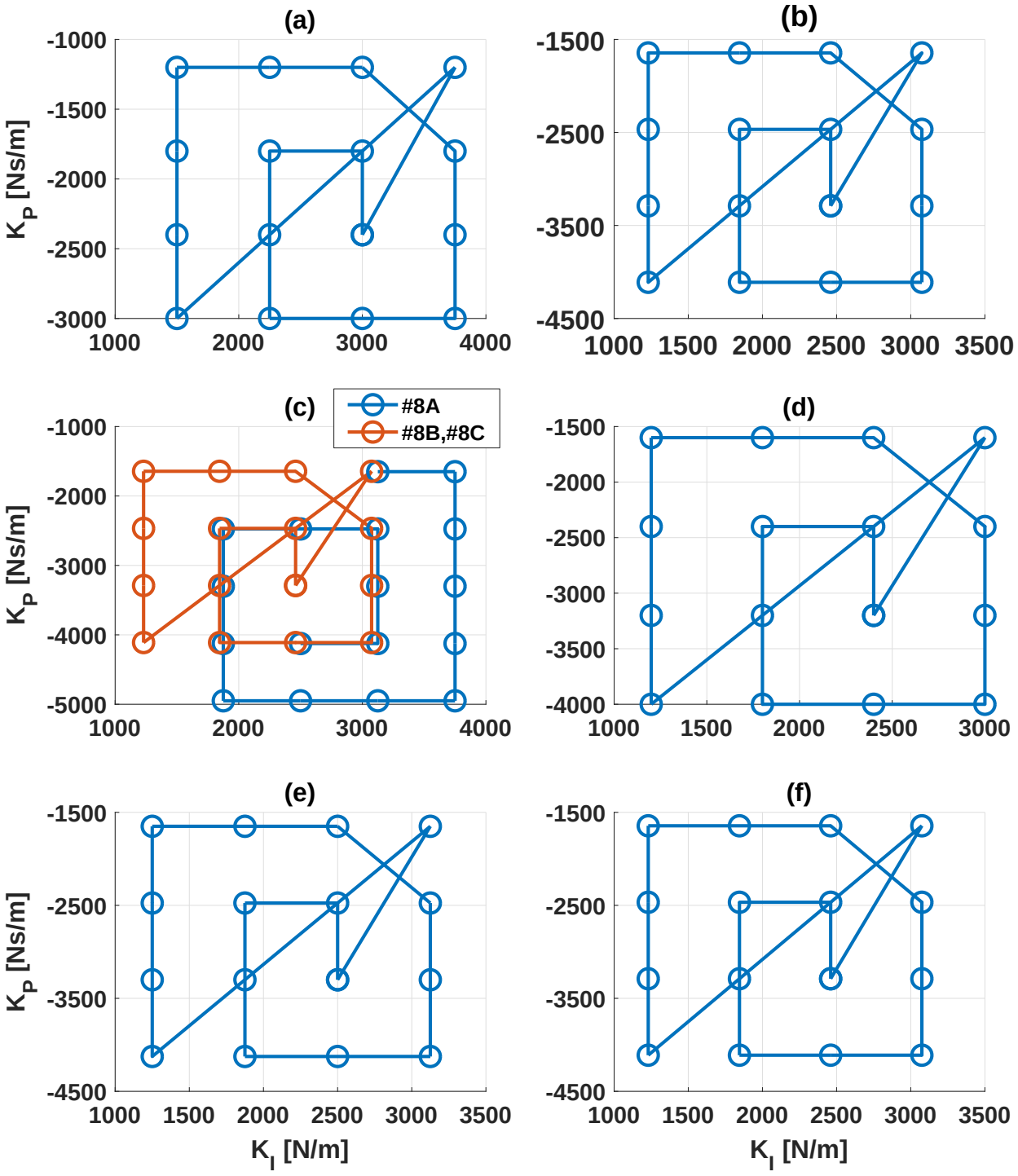


Figure 2.32: Tested K_I and K_P gains for test cases (a) #5A, 5B, and 5C; (b) #7A, 7B, and 7C; (c) #8A, 8B, and 8C; (d) #9A, 9B, and 9C; (e) #10A, 10B, and 10C; (f) #11A, 11B, and 11C.

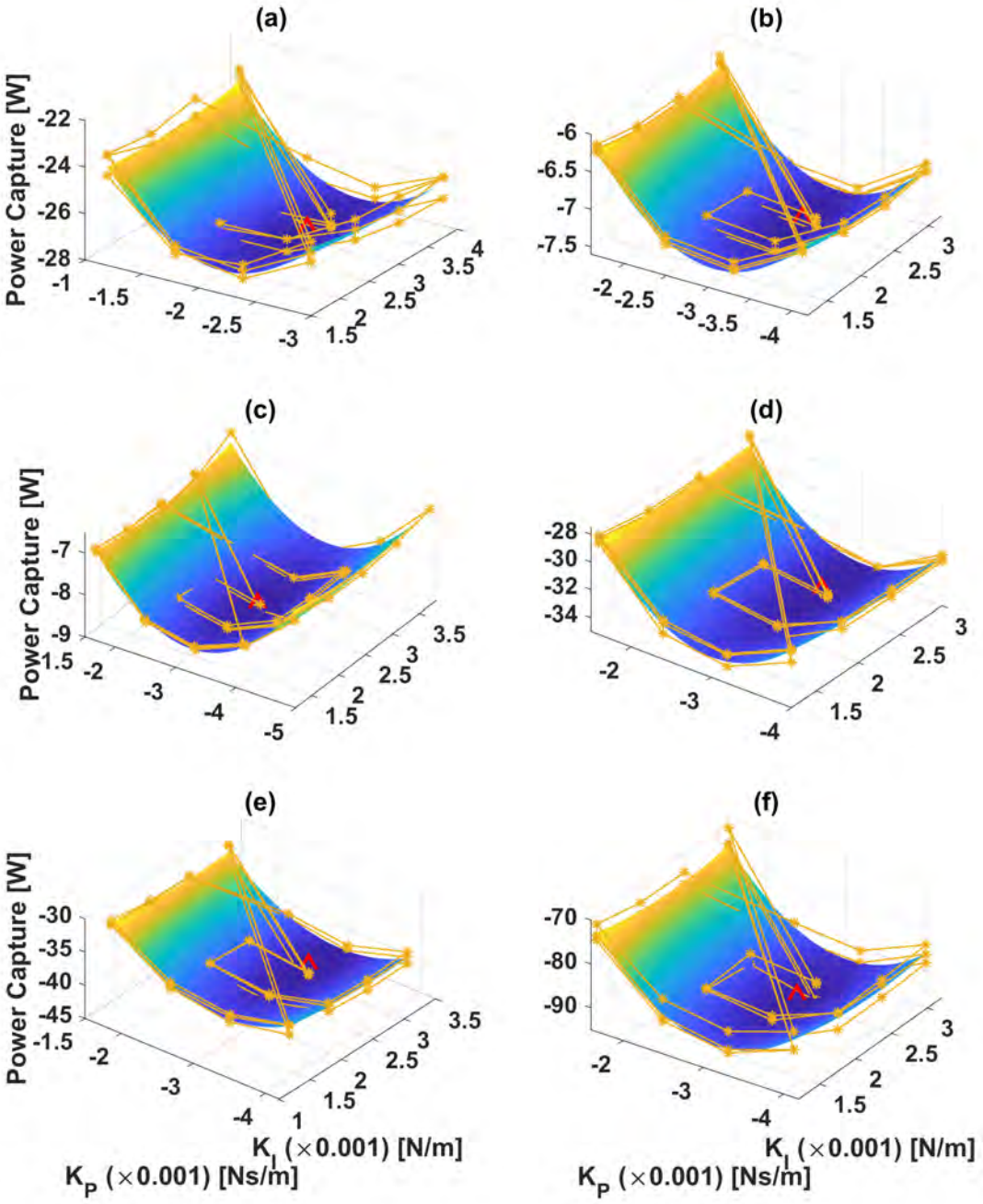


Figure 2.33: Estimated quadratic surfaces and optimal K_I and K_P gains located at the red $\triangle$ markers for test cases (a) #5A, 5B, and 5C; (b) #7A, 7B, and 7C; (c) #8A, 8B, and 8C; (d) #9A, 9B, and 9C; (e) #10A, 10B, and 10C; (f) #11A, 11B, and 11C.

References

[1] G. Bacelli, S. J. Spencer, D. C. Patterson, and R. G. Coe, "Wave tank and bench-top control testing of a wave energy converter," *accepted by Applied Ocean Research*, 2018.

[2] R. G. Coe, G. Bacelli, D. Patterson, and D. G. Wilson, "Advanced WEC Dynamics & Controls FY16 testing report," Sandia National Labs, Albuquerque, NM, Tech. Rep. SAND2016-10094, October 2016. [Online]. Available: https://mhkdr.openei.org/submissions/151

[3] G. Bacelli, R. G. Coe, D. Patterson, and D. Wilson, "System identification of a heaving point absorber: Design of experiment and device modeling," *Energies*, vol. 10, no. 10, p. 472, 2017. [Online]. Available: http://www.mdpi.com/1996-1073/10/4/472

[4] G. Bacelli and R. G. Coe, "WEC system identification and model validation," in *Marine Energy Technology Symposium (METS2017)*, Washington, D.C., 2017.

[5] H. Cho, G. Bacelli, and R. G. Coe, "Linear and nonlinear system identification of a wave energy converter," in *Proceedings of the 6th Marine Energy Technology Symposium (METS)*, Washington, D.C., 2018.

[6] R. G. Coe, G. Bacelli, V. Nevarez, H. Cho, and F. Wilches-Bernal, "A comparative study on wave prediction for WECs," Sandia National Laboratories, Tech. Rep. SAND2018-10945, 2018.

[7] D. Patterson, D. Bull, G. Bacelli, and R. Coe, "Instrumentation of a WEC device for controls testing," in *Proceedings of the 3rd Marine Energy Technology Symposium (METS2015)*, Washington DC, Apr. 2015.

[8] G. Bacelli, S. J. Spencer, R. G. Coe, A. Mazumdar, D. Patterson, and K. Dullea, "Design and bench testing of a model-scale WEC for advanced PTO control research," in *European Wave and Tidal Energy Conference (EWTEC)*, Cork, Ireland, 2017.

[9] S. D. Cairano and A. Bemporad, "Model predictive control tuning by controller matching," *IEEE Transactions on Automatic Control*, vol. 55, no. 1, pp. 185–190, Jan 2010.

Appendix A

Dataset description

The dataset collected by this experiment is available online at `https://mhkdr.openei.org` as MATLAB `.mat` files Additionally, some example plotting and analysis are provided as `.m` scripts. Table A.1 provides a listing of each experiment. The input signals for the wave maker, heave, surge, and pitch actuators are listed for each test. Note that tests 35-41, 43-52, and 56-61 are wave calibration test, in which the buoy is not present in the basin.

A listing of the device location and wave probe locations within the basin is provided in Table A.2 (these locations are also plotted in Figure 1.5).

Table A.1: Test log.

Test ID	End Time	Test length (approx.) [min]	Wave maker	Heave actuator	Surge actuator	Pitch actuator
001	5/11/18 8:37:00	11	None	Manual Saw Tooth	Manual Saw Tooth	Manual Saw Tooth
002	5/11/18 9:05:00	6	None	Manual Desired, No Damping vs Damping	Manual Desired, No Damping vs Damping	Manual Desired, No Damping vs Damping
003	5/11/18 9:42:00	13	None	WaveformA60: gain=1500	WaveformB60: gain=1500	WaveformC60: gain=400
004	5/11/18 10:02:00	14	None	WaveformC60: gain=1500	WaveformA60: gain=1500	WaveformB60: gain=400
005	5/11/18 10:18:00	12	None	WaveformB60: gain=1500	WaveformC60: gain=1500	WaveformA60: gain=400
006	5/11/18 11:34:00	4	None	None	Test lock out (1000N sine)	Test lock out (400 Nm sine)
009	5/11/18 14:48:00	130	#8A (3.5s, 5in, 3.3)	PI Matrix (25 point)	Locked out	Locked out
010	5/11/18 15:46:00	50	#10A (3.5s, 10in, 3.3)	PI Matrix (9 point)	Locked out	Locked out
011	5/11/18 16:20:00	20	#8A (3.5s, 5in, 3.3)	MPC (umax = 3900, 400, 200)	Locked out	Locked out
012	5/11/18 16:32:00	11	None	WaveformB60: gain=1500	Locked out	Locked out
013	5/14/18 4:49:00	11	None	WaveformB60: gain=1500	Locked out	Locked out
016	5/14/18 9:26:00	90	#10B (3.5s, 10in, 3.3)	PI Matrix (16 point)	Locked out	Locked out
017	5/14/18 11:08:00	90	#10C (3.5s, 10in, 3.3)	PI Matrix (16 point)	Locked out	Locked out
018	5/14/18 12:58:00	90	#10A (3.5s, 10in, 3.3)	PI Matrix (16 point)	Locked out	Locked out
021	5/14/18 15:09:00	30	#10A (3.5s, 10in, 3.3)	MPC (umax = 3900, 1200, 800, 400, 3900)	Locked out	Locked out
022	5/14/18 15:43:00	30	#10B (3.5s, 10in, 3.3)	MPC (umax = 3900, 1200, 800, 400, 3900)	Locked out	Locked out
023	5/14/18 16:18:00	30	#10C (3.5s, 10in, 3.3)	MPC (umax = 3900, 1200, 800, 400, 3900)	Locked out	Locked out
024	5/14/18 16:31:00	9	None	WaveformB60: gain=1500	Locked out	Locked out
025	5/15/18 4:28:00	6	None	WaveformB60: gain=1500	Locked out	Locked out
031	5/15/18 10:00:00	90	#5B (2.5, 10, 1)	PI Matrix (16 point)	Locked out	Locked out
032	5/15/18 11:43:00	90	#5C (2.5, 10, 1)	PI Matrix (16 point)	Locked out	Locked out
033	5/15/18 13:27:00	90	#5A (2.5, 10, 1)	PI Matrix (16 point)	Locked out	Locked out
034	5/15/18 14:10:00	30	#5A (2.5, 10, 1)	MPC (umax = 3900, 750, 500, 250, 3900)	Locked out	Locked out
035	5/15/18 15:04:00	30	#5B (2.5, 10, 1)	MPC (umax = 3900, 750, 500, 250, 3900)	Locked out	Locked out
036	5/15/18 15:41:00	30	#5C (2.5, 10, 1)	MPC (umax = 3900, 750, 500, 250, 3900)	Locked out	Locked out
038	5/15/18 16:27:00	30	#8A (3.5s, 5in, 3.3)	MPC (umax = 3900, 600, 400, 200)	Locked out	Locked out
039	5/15/18 16:41:00	6	None	WaveformB60: gain=1500	Locked out	Locked out
040	5/16/18 4:23:00	6	None	WaveformB60: gain=1500	Locked out	Locked out
041	5/16/18 6:04:00	90	#8B (3.5s, 5in, 3.3)	PI Matrix (16 point)	Locked out	Locked out
042	5/16/18 7:45:00	90	#8C (3.5s, 5in, 3.3)	PI Matrix (16 point)	Locked out	Locked out
043	5/16/18 8:25:00	30	#8B (3.5s, 5in, 3.3)	MPC (umax = 3900, 600, 400, 200)	Locked out	Locked out
044	5/16/18 9:00:00	30	#8C (3.5s, 5in, 3.3)	MPC (umax = 3900, 600, 400, 200)	Locked out	Locked out
045	5/16/18 10:40:00	90	#11A (3.5s, 15in, 3.3)	PI Matrix (16 point)	Locked out	Locked out
046	5/16/18 12:25:00	90	#11B (3.5s, 15in, 3.3)	PI Matrix (16 point)	Locked out	Locked out
047	5/16/18 14:10:00	90	#11C (3.5s, 15in, 3.3)	PI Matrix (16 point)	Locked out	Locked out
048	5/16/18 14:51:00	30	#11A (3.5s, 15in, 3.3)	MPC (umax = 3900, 1875, 1250, 625)	Locked out	Locked out
049	5/16/18 15:27:00	30	#11B (3.5s, 15in, 3.3)	MPC (umax = 3900, 1875, 1250, 625)	Locked out	Locked out
050	5/16/18 16:02:00	30	#11C (3.5s, 15in, 3.3)	MPC (umax = 3900, 1875, 1250, 625)	Locked out	Locked out
051	5/16/18 16:14:00	6	None	WaveformB60: gain=1500	Locked out	Locked out
052	5/17/18 4:36:00	6	None	WaveformB60: gain=1500	Locked out	Locked out
055	5/17/18 7:30:00	90	#7A (3.5s, 5in, 1)	PI Matrix (16 point)	Locked out	Locked out
056	5/17/18 9:20:00	90	#7B (3.5s, 5in, 1)	PI Matrix (16 point)	Locked out	Locked out
057	5/17/18 11:02:00	90	#7C (3.5s, 5in, 1)	PI Matrix (16 point)	Locked out	Locked out
058	5/17/18 11:41:00	30	#7A (3.5s, 5in, 1)	MPC (umax = 3900, 600, 400, 200)	Locked out	Locked out
059	5/17/18 12:17:00	30	#7B (3.5s, 5in, 1)	MPC (umax = 3900, 600, 400, 200)	Locked out	Locked out
060	5/17/18 12:52:00	30	#7C (3.5s, 5in, 1)	MPC (umax = 3900, 600, 400, 200)	Locked out	Locked out
061	5/17/18 14:42:00	90	#7A (3.5s, 5in, 1)	Mechanical PI Matrix (16 point)	Locked out	Locked out
062	5/17/18 16:23:00	90	#3A (3.5s, 5in, 1)	PI Matrix (16 point)	Locked out	Locked out
063	5/17/18 16:35:00	7	None	WaveformB60: gain=1500	Locked out	Locked out
064	5/18/18 4:29:00	7	None	WaveformB60: gain=1500	Locked out	Locked out
065	5/18/18 6:25:00	90	#3B (2.5s, 5in, 1)	PI Matrix (16 point)	Locked out	Locked out
066	5/18/18 8:05:00	90	#3C (2.5s, 5in, 1)	PI Matrix (16 point)	Locked out	Locked out
067	5/18/18 8:43:00	30	#3A (2.5s, 5in, 1)	MPC (umax = 3900, 450, 300, 150)	Locked out	Locked out
068	5/18/18 9:18:00	30	#3B (2.5s, 5in, 1)	MPC (umax = 3900, 450, 300, 150)	Locked out	Locked out
069	5/18/18 9:53:00	30	#3C (2.5s, 5in, 1)	MPC (umax = 3900, 450, 300, 150)	Locked out	Locked out
070	5/18/18 11:31:00	90	#9A (3.5s, 10in, 1)	PI Matrix (16 point)	Locked out	Locked out
071	5/18/18 13:11:00	90	#9B (3.5s, 10in, 1)	PI Matrix (16 point)	Locked out	Locked out
072	5/18/18 14:51:00	90	#9C (3.5s, 10in, 1)	PI Matrix (16 point)	Locked out	Locked out
073	5/18/18 15:32:00	30	#9A (3.5s, 10in, 1)	MPC (umax = 3900, 900, 600, 300)	Locked out	Locked out
074	5/18/18 16:07:00	30	#9B (3.5s, 10in, 1)	MPC (umax = 3900, 900, 600, 300)	Locked out	Locked out

Table A.1: Test log (cont.)

Test ID	End Time	Test length (approx.) [min]	Wave maker	Heave actuator	Surge actuator	Pitch actuator
075	5/18/18 16:19:00	7	None	WaveformB60: gain=1500	Locked out	Locked out
076	5/18/18 16:29:00	6	None	WaveformB60: gain=1500	Locked out	Locked out
077	5/21/18 4:26:00	8	None	WaveformB60: gain=1500	Locked out	Locked out
078	5/21/18 5:02:00	30	#9C (3.5s, 10in, 1)	MPC (umax = 3900, 900, 600, 300)	Locked out	Locked out
080	5/21/18 6:20:00	6	None	WaveformB60: gain=1500	Locked out	Locked out
083	5/21/18 8:58:00	90	#3A (2.5s, 5in, 1)	Mechanical PI Matrix (16 point)	Locked out	Locked out
085	5/21/18 10:22:00	30	#3A (2.5s, 5in, 1)	Mechanical MPC (umax = 3900, 1200, 800, 400)	Locked out	Locked out
086	5/21/18 11:02:00	25	#3A (2.5s, 5in, 1)	Multisine over PI	Locked out	Locked out
087	5/21/18 11:39:00	25	#7A (3.5s, 5in, 1)	Multisine over PI	Locked out	Locked out
088	5/21/18 12:10:00	25	#7A (3.5s, 5in, 1)	Multisine B	Locked out	Locked out
089	5/21/18 12:42:00	25	#7A (3.5s, 5in, 1)	Multisine A	Locked out	Locked out
090	5/21/18 13:11:00	25	#7A (3.5s, 5in, 1)	Multisine B	Locked out	Locked out
095	5/21/18 15:57:00	6	None	WaveformB60: gain=1500	WaveformA60: gain=1500	Locked out
096	5/21/18 16:07:00	6	None	WaveformC60: gain=1500	WaveformA60: gain=1500	Locked out
097	5/21/18 16:22:00	12	None	WaveformC60: gain=4000	WaveformA60: gain=2250	Locked out
098	5/21/18 16:35:00	6	None	WaveformA60: gain=-1500	WaveformA60: gain=1500	Locked out
099	5/21/18 16:40:00	2	None	WaveformA60: gain=5500	Virtual Spring	Locked out
100	5/21/18 16:50:00	6	None	WaveformA60: gain=4000	Virtual Spring	Locked out
101	5/22/18 4:45:00	6	None	WaveformB60: gain=1500	WaveformA60: gain=1500	Locked out
106	5/22/18 5:59:00	2	None	WaveformB60: gain=1500	WaveformA60: gain=1500	Locked out
109	5/22/18 7:27:00	12	None	WaveformB60: gain=4000	WaveformA60: gain=2250	Locked out
111	5/22/18 9:45:00	90	#3A (2.5s, 5in, 1)	PI	PI Matrix (16 point)	Locked out
112	5/22/18 11:37:00	90	#3A (2.5s, 5in, 1)	PI Matrix (16 point)	PI	Locked out
113	5/22/18 13:19:00	90	#10A (3.5s, 10in, 3.3)	PI	PI Matrix (16 point)	Locked out
114	5/22/18 14:18:00	50	#7A (3.5s, 5in, 1)	PI	PI Matrix (9 point)	Locked out
115	5/22/18 15:30:00	55	#11A (3.5s, 15in, 3.3)	PI	PI Matrix (9 point)	Locked out
116	5/22/18 16:24:00	35	#11A (3.5s, 15in, 3.3)	PI	PI Matrix (6 point)	Locked out
117	5/22/18 16:45:00	6	None	WaveformB60: gain=1500	WaveformA60: gain=1500	Locked out
118	5/23/18 5:02:00	7	None	WaveformA60: gain=1500	WaveformB60: gain=1500	WaveformC60: gain=400
119	5/23/18 5:17:00	7	None	None	Virtual Spring and Manual PI	None
120	5/23/18 5:27:00	7	None	None	Virtual Spring and Manual PI	WaveformC60: gain=400
121	5/23/18 5:45:00	12	None	None	Virtual Spring and Manual PI	WaveformC60: gain=600
122	5/23/18 6:00:00	12	None	None	Virtual Spring and Manual PI	Random: gain=200
123	5/23/18 6:41:00	11	None	None	Virtual Spring; WaveformA60: gains 1500, 2000	Manual PI
124	5/23/18 7:01:00	11	None	None	Virtual Spring; WaveformA60: gains 2000	Manual PI
125	5/23/18 7:18:00	7	None	WaveformC60: gain=1500	WaveformB60: gain=1500	WaveformA60: gain=400
126	5/23/18 7:27:00	7	None	WaveformB60: gain=1500	WaveformC60: gain=1500	WaveformA60: gain=400
127	5/23/18 7:36:00	7	None	WaveformC60: gain=1500	WaveformA60: gain=1500	WaveformB60: gain=400
128	5/23/18 9:35:00	7	#3A (2.5s, 5in, 1)	Manual PI	Manual PI	PI Matrix
131	5/23/18 12:14:00	90	#2A (1.58s, 5in, 1)	Manual PI	Manual PI	PI Matrix
132	5/23/18 14:04:00	90	#5A (2.5s, 10in, 1)	Manual PI	Manual PI	PI Matrix
133	5/23/18 15:06:00	50	#3A (2.5s, 5in, 1)	Waveform (ACBC) = 700	Waveform (BBCA) = 700	Waveform (CAAB) = 200
134	5/23/18 16:06:00	50	#7A (3.5s, 5in, 1)	Waveform (ACBC) = 700	Waveform (BBCA) = 700	Waveform (CAAB) = 200
135	5/23/18 16:21:00	9	None	WaveformC60: gain = 1500	WaveformA60: gain = 1500	WaveformB60: gain = 400
136	5/24/18 4:32:00	6	None	WaveformC60: gain = 1500	WaveformA60: gain = 1500	WaveformB60: gain = 400
137	5/24/18 6:22:00	90	#7A (3.5s, 5in, 1)	Manual PI	Manual PI	PI Matrix
138	5/24/18 8:10:00	90	#3A (2.5s, 5in, 1)	Manual PI	Manual PI	PI Matrix
139	5/24/18 9:13:00	50	#2A (1.58s, 5in, 1)	Manual PI	Manual PI	PI Matrix (200,400,600) (2000,2500,3000)
140	5/24/18 10:05:00	15	#10A (3.5s, 10in, 1)	Disturbed VA; VA Manual PI	Disturbed VA; VA Manual PI	WaveformB60: gain = 400; Manual PI
141	5/24/18 10:14:00	5	#11A (3.5s, 15in, 1)	VA Manual PI	VA Manual PI	Manual PI
142	5/24/18 10:32:00	10	#11A (3.5s, 15in, 1)	VA Manual PI	VA Manual PI	Manual PI
143	5/24/18 10:51:00	20	None	Manual PI	Disturbed CC Manual PI	CC Manual PI
144	5/24/18 11:07:00	12	#3A (2.5s, 5in, 1)	Manual PI	CC Manual PI	CC Manual PI
145	5/24/18 11:18:00	7	#5A (2.5s, 10in, 1)	Manual PI	CC Manual PI	CC Manual PI
146	5/24/18 11:40:00	12	None	Random: gain=700*.8	Random: gain=700*.8	Random: gain=200
147	5/24/18 12:10:00	15	#7A (3.5s, 5in, 1)	Locked out	Locked out	Locked out
148	5/24/18 12:30:00	15	#3A (2.5s, 5in, 1)	Locked out	Locked out	Locked out
149	5/24/18 12:50:00	15	#2A (1.58s, 5in, 1)	Locked out	Locked out	Locked out
935			#5A	(WEC absent)		
936			#6A	(WEC absent)		

Table A.1: Test log (cont.)

Test ID	End Time	Test length (approx.) [min]	Wave maker	Heave actuator	Surge actuator	Pitch actuator
937			#7A	(WEC absent)		
938			#8A	(WEC absent)		
939			#9A	(WEC absent)		
940			#10A	(WEC absent)		
941			#11A	(WEC absent)		
943			#1A	(WEC absent)		
944			#2A	(WEC absent)		
945			#3A	(WEC absent)		
946			#4B	(WEC absent)		
947			#4C	(WEC absent)		
948			#5B	(WEC absent)		
949			#5C	(WEC absent)		
950			#6B	(WEC absent)		
951			#6B	(WEC absent)		
952			#6C	(WEC absent)		
956			#7B	(WEC absent)		
957			#7C	(WEC absent)		
958			#8B	(WEC absent)		
959			#8C	(WEC absent)		
960			#10B	(WEC absent)		
961			#10C	(WEC absent)		

Table A.2: Wave sensor locations (see also Figure 1.5).

Item	x location [m]	y location [m]	Sensor type	Variable name
WEC	42.630	76.774	N/A	N/A
BRIDGEPROBE1	43.191	92.988	Senix ToughSonic TSPC-30S1-232	BPDAQ.BRIDGEPROBE1
BRIDGEPROBE3	34.019	67.668	Senix ToughSonic TSPC-30S1-232	BPDAQ.BRIDGEPROBE3
BRIDGEPROBE4	42.101	60.626	Senix ToughSonic TSPC-30S1-232	BPDAQ.BRIDGEPROBE4
BRIDGEPROBE5	25.669	44.781	Senix ToughSonic TSPC-30S1-232	BPDAQ.BRIDGEPROBE5
BRIDGEPROBE6	33.722	37.397	Senix ToughSonic TSPC-30S1-232	BPDAQ.BRIDGEPROBE6
BRIDGEPROBE8	27.536	20.323	Senix ToughSonic TSPC-30S1-232	BPDAQ.BRIDGEPROBE8
BUOY01	44.356	81.111	Senix ToughSonic TSPC-30S1-232	BPDAQ.BUOY01
BUOY03	44.804	82.479	Senix ToughSonic TSPC-30S1-232	BPDAQ.BUOY03
BUOY04	45.860	86.399	Senix ToughSonic TSPC-30S1-232	BPDAQ.BUOY04
BUOY02	40.681	79.205	Senix ToughSonic TSPC-30S1-232	BPDAQ.BUOY02
BUOY05	41.135	77.296	Senix ToughSonic TSPC-30S1-232	BPDAQ.BUOY05
SENIX7	37.357	46.976	Senix ToughSonic TSPC-30S1-232	BPDAQ.SENIX7
SENIX8	36.960	47.076	Senix ToughSonic TSPC-30S1-232	BPDAQ.SENIX8
SENIX9	37.136	46.637	Senix ToughSonic TSPC-30S1-232	BPDAQ.SENIX9
SENIX10	37.619	46.665	Senix ToughSonic TSPC-30S1-232	BPDAQ.SENIX10
SENIX11	37.740	47.127	Senix ToughSonic TSPC-30S1-232	BPDAQ.SENIX11
SENIX12	37.334	47.391	Senix ToughSonic TSPC-30S1-232	BPDAQ.SENIX12
SENIX13	36.549	47.124	Senix ToughSonic TSPC-30S1-232	BPDAQ.SENIX13
SENIX14	36.825	46.351	Senix ToughSonic TSPC-30S1-232	BPDAQ.SENIX14
SENIX15	37.081	47.747	Senix ToughSonic TSPC-30S1-232	BPDAQ.SENIX15
SENIX17	37.881	47.609	Senix ToughSonic TSPC-30S1-232	BPDAQ.SENIX17
SENIX18	37.645	46.215	Senix ToughSonic TSPC-30S1-232	BPDAQ.SENIX18
SENIX16	38.167	46.842	Senix ToughSonic TSPC-30S1-232	BPDAQ.SENIX16
OSSI01	24.576	26.455	Ocean Sensor Systems, Inc. OSSI-010-002F	BADAQ.OSSI01
OSSI02	24.967	26.189	Ocean Sensor Systems, Inc. OSSI-010-002F	BADAQ.OSSI02
OSSI03	25.340	26.544	Ocean Sensor Systems, Inc. OSSI-010-002F	BADAQ.OSSI03
OSSI04	24.647	26.926	Ocean Sensor Systems, Inc. OSSI-010-002F	BADAQ.OSSI04
OSSI05	24.931	25.802	Ocean Sensor Systems, Inc. OSSI-010-002F	BADAQ.OSSI05

Table A.2: Wave sensor locations (cont.)

Item	x location [m]	y location [m]	Sensor type	Variable name
OSSI06	25.631	26.208	Ocean Sensor Systems, Inc. OSSI-010-002F	BADAQ.OSSI06
OSSI07	25.651	27.035	Ocean Sensor Systems, Inc. OSSI-010-002F	BADAQ.OSSI07
OSSI08	24.216	27.077	Ocean Sensor Systems, Inc. OSSI-010-002F	BADAQ.OSSI08
OSSI09	26.076	26.193	Ocean Sensor Systems, Inc. OSSI-010-002F	BADAQ.OSSI09
OSSI10	25.349	27.773	Ocean Sensor Systems, Inc. OSSI-010-002F	BADAQ.OSSI10
OSSI11	24.499	27.394	Ocean Sensor Systems, Inc. OSSI-010-002F	BADAQ.OSSI11
OSSI12	24.223	26.213	Ocean Sensor Systems, Inc. OSSI-010-002F	BADAQ.OSSI12
SAA01	20.239	78.852	Senix ToughSonic TSPC-30S1-232	SAADAQ.SAA01
SAA02	19.343	78.300	Senix ToughSonic TSPC-30S1-232	SAADAQ.SAA02
SAA04	21.101	77.670	Senix ToughSonic TSPC-30S1-232	SAADAQ.SAA04
SAA05	21.048	78.179	Senix ToughSonic TSPC-30S1-232	SAADAQ.SAA05
SAA03	20.931	79.404	Senix ToughSonic TSPC-30S1-232	SAADAQ.SAA03

Appendix B

Sample code

This section contains sample code used to process the results.

B.1 MPC vs. PI

This code was used to process the results shown in Section 2.2 which compare the performance of MPC and PI controllers.

```
function [] = MASK2B_plotMpcPi_elec()
close all
clear
clc

ID = {'3B', '5B', '8B', '9B'};    % test case #
Pi = [65, 31, 41, 71];    % test ID for PI
Mpc = [68, 35, 43, 74];   % test ID for MPC

xtz = [450, 500];         % time range to be plotted

for ii = 1:length(ID)
    tmp(ID{ii},Mpc(ii),Pi(ii),xtz)
end
end

function [] = tmp(TestID,MpcNum,PiNum,xtz)
%This function plots figures

PiFile = sprintf('MASK2B_%03d',PiNum);     % loading PI data
MpcFile = sprintf('MASK2B_%03d',MpcNum);   % loading MPC data

hin(1) = struct2array(load(PiFile,'h'));   % loading heave data
hin(2) = struct2array(load(MpcFile,'h'));

bin(1) = struct2array(load(PiFile,'b'));   % loading synchronization data
bin(2) = struct2array(load(MpcFile,'b'));

win(1) = struct2array(load(PiFile,'BADAQ'));  % loading wave data
win(2) = struct2array(load(MpcFile,'BADAQ'));

```

```
bname = sprintf('MpcPi_%s_%03d_%03d',TestID,MpcNum,PiNum);
                                                % filename for figures

%%

hin(1).name = 'PI';
hin(2).name = 'MPC';

hin(1).delay = 0;   % detect time delay between PI and MPC
hin(2).delay = (find(bin(1).WavesOn > 0.5,1) - ...
    find(bin(2).WavesOn > 0.5,1)) / 1e3;

hin(1).valid = 300001:600001;   % valid data range
hin(2).valid = hin(1).valid - hin(2).delay * 1e3;

%%
figure('name','time domain')
ii = 0;
for h = hin
    ii = ii + 1;
    h.K2 = h.Kt.^2*2/3/(.484);     % motor constant
    ax(1) = subplot(3,1,1);        % plot 'Time vs. Control Force'
    grid on
    hold on
    ylabel('Control force [N]')
    plot(h.t(h.valid) + h.delay,h.F_des(h.valid),'DisplayName',h.name)

    ax(2) = subplot(3,1,2);        % plot 'Time vs. Velocity'
    grid on
    hold on
    ylabel('Velocity [m/s]')
    plot(h.t(h.valid) + h.delay,h.v_enc(h.valid),'DisplayName',h.name)

    h.pow = h.v_enc .* h.F_des + h.F_des.^2./h.K2./h.N.^2;
                                   % calculation of electrical power
    h.pow_mean = mean(h.pow(h.valid));
    lstr{ii} = sprintf('%s, $\\bar{P} = %.2f$ W',h.name,h.pow_mean);
    hin(ii).pow = h.pow;

    ax(3) = subplot(3,1,3);        % plot 'Time vs. Power'
    grid on
    hold on
    plot(h.t(h.valid) + h.delay,h.pow(h.valid),'DisplayName',h.name)
    ylabel('Power [W]')
    xlabel('Time [s]')
end

subplot(3,1,3);
legend(lstr,'interpreter','latex','location','best')
linkaxes(ax,'x')
xlim(xtz)
export_fig([bname,'_td.pdf'],'-transparent')  % save figure file in pdf

%%
```

```
% plot wave power spectral density in frequency domain
figure('name','waves')
grid on
hold on
xlabel('Frequency [Hz]')
ylabel('Wave power spectral density [m^2s]')
ax(1) = gca;
ax(2) = axes('Parent',gcf,'Position',[0.46 0.59 0.42 0.30]);
hold(ax(2),'on')
grid on
xlabel('Time [s]')
ylabel('Wave elev. [m]')
ii = 0;
for ii = 1:length(win)
    w = win(ii);
    h = hin(ii);

    [~,inds] = min(abs([h.t(h.valid(1)), h.t(h.valid(end))] - w.Time));
    w.valid = inds(1):1:inds(2);

    f0 = 1/mean(diff(w.Time));   % sampling frequency

    N = length(w.OSSI01(w.valid));
    freq = f0/2*linspace(0,1,N/2+1);
                    % frequency domain for single-sided Fourier transform

    fc = 5;    % cutoff frequency
    d = designfilt('lowpassfir','FilterOrder',8,...
        'CutoffFrequency',fc,'DesignMethod','window',...
        'Window',{@kaiser,3},'SampleRate',f0);   % filter specification

    OSSI01org = w.OSSI01(w.valid)-mean(w.OSSI01(w.valid));% subtract offset
    OSSI01filter = filtfilt(d,OSSI01org);   % apply (causal) filter
    WW = fft(OSSI01org)/N;   % Fourier transform (two-sided)
    sWW = 2*abs(WW(1:floor(N/2+1)));   % single-sided spectrum
    axes(ax(1));   % wave power spectral density
    hold on
    plot(freq,sWW.^2,'DisplayName',h.name)
    xlim([0.2,1.5])

    axes(ax(2));   % plot wave elevation in time domain
    hold on
    plot(w.Time(w.valid)  + h.delay,OSSI01filter,'DisplayName',h.name)
    xlim(xtz)

    mw{ii} = spectMom(freq,sWW,3);   % calculate spectral moments of waves
end

legend('location','best')
export_fig([bname,'_waves.pdf'],'-transparent')

%%
% plot power spectral density of control force, velocity, and power
figure('name','freq domain')
```

```
ii = 0;
for h = hin
    ii = ii + 1;
    h.K2 = h.Kt.^2*2/3/(.484);
    f0 = 1/mean(diff(h.t));

    N = length(h.v_enc(h.valid));
    freq = f0/2*linspace(0,1,N/2+1);
    VV = fft(h.v_enc(h.valid))/N;     % two-sided spectrum of velocity
    sVV = 2*abs(VV(1:floor(N/2+1))); % single-sided spectrum of velocity
    FF = fft(h.F_des(h.valid))/N;     % two-sided spectrum of control force
    sFF = 2*abs(FF(1:floor(N/2+1)));
                                      % single-sided spectrum of control force
    PP = conj(VV).*FF+VV.*conj(FF) + 2*conj(FF).*FF./h.K2./h.N.^2;
                                      % two-sided spectrum of elec power
    sPP = 2*abs(PP(1:floor(N/2+1))); % single-sided spectrum of elec power

    ax(1) = subplot(3,1,1);        % power spectral density of control force
    plot(freq,sFF,'DisplayName',h.name)
    grid on
    hold on
    ylabel('Control force [N]')

    ax(2) = subplot(3,1,2);        % power spectral density of velocity
    plot(freq,sVV,'DisplayName',h.name)
    grid on
    hold on
    ylabel('Velocity [m/s]')

    ax(3) = subplot(3,1,3);        % power spectral density of elec power
    grid on
    hold on
    plot(freq,sPP,'DisplayName',h.name)
    ylabel('Power [W]')
    xlabel('Frequency [Hz]')

    mp{ii} = spectMom(freq,sPP,3);  % calculate spectral moments of power

end

subplot(3,1,3);
legend('position','best')
linkaxes(ax,'x')
xlim([0.2,0.8])
export_fig([bname,'_fd.pdf'],'-transparent')

%%
% plot PDFs of control force, velocity, and power
figure('name','PDF','Position',[360   278   2*560   420])
ii = 0;

for h = hin
    ii = ii + 1;

```

```
    ax(1) = subplot(1,3,1);   % plot PDF of control force
    plotHist(h,'F_des',5,0)
    xlabel('Control force [N]')
    grid on
    hold on
    ylabel('PDF(x)')

    ax(2) = subplot(1,3,2);   % plot PDF of velocity
    plotHist(h,'v_enc',0.001,0)
    xlabel('Velocity [m/s]')
    grid on
    hold on
    set(gca,'YTickLabel',[])

    ax(3) = subplot(1,3,3);   % plot PDF of electrical power
    plotHist(h,'pow',0.5,0)
    xlabel('Power [W]')
    grid on
    hold on
    set(gca,'YTickLabel',[])

end

subplot(1,3,1);
legend('PI','MPC','location','best')

linkaxes(ax,'y')
export_fig([bname,'_pdf.pdf'],'-transparent')

%%
% plot CCDFs of control force, velocity, and power
figure('name','CCDF')
ii = 0;
for h = hin
    ii = ii + 1;

    ax(1) = subplot(1,3,1);   % plot CCDF of control force
    plotCCDF(h,'F_des',0)
    xlabel('Control force [N]')
    ylabel('CCDF(x)')
    grid on
    hold on

    ax(2) = subplot(1,3,2);   % plot CCDF of velocity
    plotCCDF(h,'v_enc',0)
    xlabel('Velocity [m/s]')
    grid on
    hold on
    set(gca,'YTickLabel',[])

    ax(3) = subplot(1,3,3);   % plot CCDF of electrical power
    plotCCDF(h,'pow',0)
    xlabel('Power [W]')
    grid on
```

```
    hold on
    set(gca,'YTickLabel',[])

end

subplot(1,3,3);
l1 = legend();
set(l1,'position',[0.73 0.82 0.20 0.12],'interpreter','latex');
linkaxes(ax,'y')
export_fig([bname,'_ccdf.pdf'],'-transparent')

%%
fprintf('%s\n',TestID)   % display current Test ID
fprintf('delay: %.2f\n',hin(2).delay) % display time delay between PI & MPC
for ii = 1:length(hin)   % display spectral moments of waves
    fprintf('%3s wave mom:\t',hin(ii).name)
    fprintf('%.3e\t',mw{ii})
    fprintf('\n')
end
for ii = 1:length(hin)   % display spectral moments of power
    fprintf('%3s pow mom:\t',hin(ii).name)
    fprintf('%.3e\t',mp{ii})
    fprintf('\n')
end
getFit(hin,'F_des');
getFit(hin,'v_enc');
getFit(hin,'pow');
fprintf('\n')

end
%%
% calculate FIT values
function [fit,nrmse] = getFit(s,fieldname)
for jj = 1:2
    tmp = s(jj).(fieldname);
    sig{jj} = tmp(s(jj).valid);
end
sig = fliplr(sig);
nrmse = norm(sig{1}-sig{2})/norm(sig{1}-mean(sig{1}));  % NMRSE
fit = (1-nrmse)*100;   % calculate FIT
fprintf('%5s,\tFIT = %.2f, NRMSE = %.2f\n',fieldname,fit,nrmse)
end

function plotHist(s,fieldname, binwidth, absflag)
%This function plots PDF
tmp = s.(fieldname);
if absflag
    tmp = abs(tmp);
end
sig = tmp(s.valid);
[n,x] = hist(sig,min(sig):binwidth:max(sig));
n = n / sum(n);
plot(x,n)
end
```

```
function m = spectMom(freq,S,n)
%This function computes spectral moments
for ii = 1:n
    m(ii) = trapz(freq(2:end),S(2:end)' .* freq(2:end) .^ (ii - 2));
end
end

function plotCCDF(s,fieldname,absflag)
%This function compute CCDF according to the sign of x value
sn = {'$sign(x)>0$','$sign(x)<0$'};
ls = {'--','-.'};
q = s.(fieldname);
q = q(s.valid);
if absflag   % if absflag == 1, do not consider the sign of x
    [f,x] = ecdf(abs(q));   % compute ECDF
    ccdf = 1-f;             % compute CCDF
    semilogy(x,ccdf, ls{ii},'DisplayName',s.name)
    hold on
else
    tmp{1} = q(q > 0);      % when x value is positive
    tmp{2} = q(q < 0);      % when x value is negative
    for ii = 1:2
        [f,x] = ecdf(abs(tmp{ii}));  % compute ECDF
        ccdf = 1-f;                  % compute CCDF
        semilogy(x,ccdf, ls{ii},'DisplayName', ...
            sprintf('%s, %s',s.name, sn{ii}))
        hold on
        set(gca,'ColorOrderIndex',get(gca,'ColorOrderIndex') - 1)
    end
    set(gca,'ColorOrderIndex',get(gca,'ColorOrderIndex') + 1)
end
end
```

B.2 MPC: unconstrained vs. constrained

This code was used to process the results shown in Section 2.2 which analyze the performance of constrained MPC.

```
function [] = MASK2B_plotMpc_Constrained_elec()
close all
clear
clc

ID = {'3B', '5B', '8B', '9B'};    % test case #
Mpc = [68, 35, 43, 74];           % test ID for MPC

xtz = [450, 500];                 % time range to be plotted

for ii = 1:length(ID)
```

```
    tmp(ID{ii},Mpc(ii),xtz)
end

end

function [] = tmp(TestID,MpcNum,xtz)
% loading MPC data
MpcFile = sprintf('MASK2B_%03d',MpcNum);

hin = struct2array(load(MpcFile,'h'));

bname = sprintf('Mpc_%s_%03d_Constrained',TestID,MpcNum);

%%
hin(2) = hin(1);   % create and duplicate four cases
hin(3) = hin(2);
hin(4) = hin(3);
hin(1).name = 'Unconstrained';
hin(2).name = 'ConstrainedCase1';
hin(3).name = 'ConstrainedCase2';
hin(4).name = 'ConstrainedCase3';

Umax0 = unique(hin(1).MPC_UMAX);
hin(1).Umax = max(Umax0);          % Unconstrained limit (arbitrarily large)
hin(2).Umax = max(Umax0(1:end-1));% Constrained Case 1 limit
hin(3).Umax = max(Umax0(1:end-2));% Constrained Case 2 limit
hin(4).Umax = max(Umax0(1:end-3));% Constrained Case 3 limit

hin(2).valid = find(hin(2).MPC_UMAX == hin(2).Umax);
                                   % valid time range for Constrained Case 1
hin(3).valid = find(hin(3).MPC_UMAX == hin(3).Umax);
                                   % valid time range for Constrained Case 2
hin(4).valid = find(hin(4).MPC_UMAX == hin(4).Umax);
                                   % valid time range for Constrained Case 3
hin(1).valid = hin(2).valid - 300000;
                                   % valid time range for Unconstrained Case

%%
% plot control force, velocity, and power
%    for Unconstrained and Constrained Cases 1-3
figure('name','time domain','pos',[10 10 1400 1200])
ii = 0;
for h = hin
    ii = ii + 1;
    h.K2 = h.Kt.^2*2/3/(.484);
    ax(1) = subplot(3,1,1);   % plot 'Time vs. Control Force'
    grid on
    hold on
    ylabel('Control force [N]')
    title('(a)')
    plot(h.t(h(1).valid)-300*(ii-1),h.F_des(h.valid),'DisplayName',h.name)
    set(gca,'FontSize',14)

    ax(2) = subplot(3,1,2);   % plot 'Time vs. Velocity'
```

```
    grid on
    hold on
    ylabel('Velocity [m/s]')
    title('(b)')
    plot(h.t(h.valid)-300*(ii-1),h.v_enc(h.valid),'DisplayName',h.name)
    set(gca,'FontSize',14)

    h.pow = h.v_enc .* h.F_des + h.F_des.^2./h.K2./h.N.^2;
                                % compute electrical power
    h.pow_mean = mean(h.pow(h.valid));
    if ii == 1
        lstr{ii} = sprintf('%s, $\\bar{P} = %.2f$ W',h.name,h.pow_mean);
    else
        lstr{ii} = sprintf('$U_{max}=%3d, \\bar{P} = %.2f$ W',h.Umax,h.
   pow_mean);
    end
    hin(ii).pow = h.pow;

    ax(3) = subplot(3,1,3);    % % plot 'Time vs. Power'
    grid on
    hold on
    plot(h.t(h.valid)-300*(ii-1),h.pow(h.valid),'DisplayName',h.name)
    ylabel('Power [W]')
    xlabel('Time [s]')
    title('(c)')
    set(gca,'FontSize',14)
end

subplot(3,1,3);
legend(lstr,'interpreter','latex','location','best','FontSize',14)
linkaxes(ax,'x')
xlim(xtz)
export_fig([bname,'_td.pdf'],'-transparent')  % save figure file in pdf

end
```

B.3 PI gain variation

This code was used to process the results shown in Section 2.3 which look at the performance of a matrix of PI controller gains.

```
close all; clear; clc;

ID = {'3A','3B','3C'};    % test case #
Pi = [62, 65, 66];        % test ID

for i = 1:length(ID)
    PiNum(i) = Pi(i);
end

```

```
Wave3.A = load(sprintf('MASK2B_%03d',PiNum(1)));  % loading data
Wave3.B = load(sprintf('MASK2B_%03d',PiNum(2)));
Wave3.C = load(sprintf('MASK2B_%03d',PiNum(3)));

%%
Kt = Wave3.B.h.Kt;
K2 = Kt.^2*2/3/(.484);    % motor constant
N = Wave3.B.h.N;          % transmission ratio

T_s = 300+1.025;          % start time
T_f = [5400,5400,5400]+1.025;     % final time
valid = {(T_s*1000+1):(T_f(1)*1000),(T_s*1000+1):(T_f(2)*1000), ...
    (T_s*1000+1):(T_f(3)*1000)};  % time range to be plotted

phasing = {'A','B','C'};
figure(1);clf;
figure(2);clf;

for i = 1:length(phasing)
    Wave3.(phasing{i}).F.vals = ... % extract control force
        reshape(Wave3.(phasing{i}).h.I(valid{i})*N*Kt,300000,[]);
    Wave3.(phasing{i}).v.vals = ... % extract velocity
        reshape(Wave3.(phasing{i}).h.v_enc(valid{i}),300000,[]);
    Wave3.(phasing{i}).x.vals = ... % extract position
        reshape(Wave3.(phasing{i}).h.x_enc(valid{i}),300000,[]);

    Wave3.(phasing{i}).F.rms = sqrt(mean(Wave3.(phasing{i}).F.vals.^2));
                                    % rms of control force
    Wave3.(phasing{i}).v.rms = sqrt(mean(Wave3.(phasing{i}).v.vals.^2));
                                    % rms of velocity
    Wave3.(phasing{i}).x.rms = sqrt(mean(Wave3.(phasing{i}).x.vals.^2));
                                    % rms of position

    Wave3.(phasing{i}).P_mech.vals = ...
        Wave3.(phasing{i}).F.vals.*Wave3.(phasing{i}).v.vals;
                                    % compute mechanical power
    Wave3.(phasing{i}).P_elec.vals = ...
        Wave3.(phasing{i}).P_mech.vals+Wave3.(phasing{i}).F.vals.^2/K2/N^2;
                                    % compute electrical power

    Wave3.(phasing{i}).KI.vals = ...
        reshape(Wave3.(phasing{i}).h.KI(valid{i}),300000,[]);
                                    % tested K_I gain
    Wave3.(phasing{i}).KP.vals = ...
        reshape(Wave3.(phasing{i}).h.KP(valid{i}),300000,[]);
                                    % tested K_P gain

  Wave3.(phasing{i}).P_mech.avg = mean(Wave3.(phasing{i}).P_mech.vals);
                                    % mean of mechanical power
  Wave3.(phasing{i}).P_elec.avg = mean(Wave3.(phasing{i}).P_elec.vals);
                                    % mean of electrical power
  Wave3.(phasing{i}).KI.avg = mean(Wave3.(phasing{i}).KI.vals);
                                    % mean of K_I gains
  Wave3.(phasing{i}).KP.avg = mean(Wave3.(phasing{i}).KP.vals);
```

```
                                    % mean of K_P gains

    figure(1);   % plot test K_I and K_P gains
    plot(Wave3.(phasing{i}).KI.avg,Wave3.(phasing{i}).KP.avg,'-o')
    hold on;grid on

end
figure(1);xlabel('K_I');ylabel('K_P')
%%
KImat = [Wave3.A.KI.avg,Wave3.B.KI.avg,Wave3.C.KI.avg];
KPmat = [Wave3.A.KP.avg,Wave3.B.KP.avg,Wave3.C.KP.avg];
P_elec_mat = [Wave3.A.P_elec.avg,Wave3.B.P_elec.avg,Wave3.C.P_elec.avg];
P_mech_mat = [Wave3.A.P_mech.avg,Wave3.B.P_mech.avg,Wave3.C.P_mech.avg];
F_mat = [Wave3.A.F.rms,Wave3.B.F.rms,Wave3.C.F.rms];
v_mat = [Wave3.A.v.rms,Wave3.B.v.rms,Wave3.C.v.rms];
valid = 1:51;

figure(2)
% quadratic_surface_fitting
C_elec = paraboloid_estimation(KImat(valid)*1e-3, KPmat(valid)*1e-3, ...
    P_elec_mat(valid));    % estimating coefficients of quadratic surface
[XX, YY] = meshgrid(linspace(min(KImat(valid)),...
    max(KImat(valid)),250)*1e-3, linspace(min(KPmat(valid)),...
    max(KPmat(valid)),250)*1e-3);
ZZ_elec = paraboloid_evaluation(XX,YY,C_elec);
                            % quadratic surface with determined coefficients
hold on
surf(XX,YY,ZZ_elec,'edgecolor','none');colorbar;% plot of quadratic surface
plot3(KImat(valid)*1e-3,KPmat(valid)*1e-3,P_elec_mat(valid),'-*')
                            % 3D-plot of K_I, K_P, and corresponding power
cl = caxis;
K_opt = estimate_optimal_gains(C_elec);
                            % estimate optimal gains by finding vertex
plot3(K_opt(1),K_opt(2),...
    paraboloid_evaluation(K_opt(1),K_opt(2),C_elec),'r^')
                            % 3D-plot of optimal gains
xlabel('K_I(\times0.001)')
ylabel('K_P(\times0.001)')
zlabel('Power Capture, W')
grid on

function C = paraboloid_estimation(X,Y,Z)
%This function estimates coeffieicnts of quadratic surface
if length(Z) < 6
    error('Number of data points must be larger than 6')
end

X = X(:);
Y = Y(:);
Z = Z(:);
n = length(Z);

optfun = @(x)sum((paraboloid_evaluation(X,Y,x)-Z).^2);
C = fmincon(optfun,[1 0 1 0 0 0],[],[],[],[],[],[],@(x)optcon(x),[]);
```

```
end

function [c,ceq] = optcon(x)
%This function specifies equality/inequality constraints
c = (-4*x(1)*x(3) + x(2)^2);
ceq = [];
end

function Z = paraboloid_evaluation(X,Y,C)
%This function calculates quadratic surface with determined coeffs
Z = C(1).*X.^2 +  C(2).*X.*Y +  C(3).*Y.^2 +  C(4) .* X + C(5) .* Y + ...
    C(6) .* ones(size(X));
end

function K = estimate_optimal_gains(C)
%This function estimates optimal gains
A = [C(1), C(2)/2; C(2)/2, C(3)];
B = [C(4), C(5)];
K = -0.5*(A \ B');
end
```

DISTRIBUTION:

1 MS 0899 Technical Library, 9536 (electronic copy)


Sandia National Laboratories