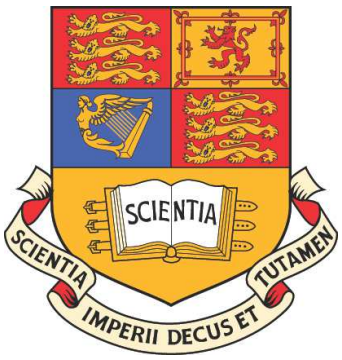
# Optimal Control and Robust Estimation for Ocean Wave Energy Converters

by

Edo Abraham




A Thesis submitted in fulfillment of requirements for the degree of
Doctor of Philosophy of Imperial College London

Department of Aeronautics
Imperial College London
2013

As required by the College, I hereby confirm that this thesis is the result of my own work, and that any ideas or quotations from the work of other people, published or otherwise, are fully acknowldged through standard referencing practices of the discipline.

Edo Abraham
Imperial College London
31 May 2013

# Abstract

This thesis deals with the optimal control of wave energy converters and some associated observer design problems. The first part of the thesis will investigate model predictive control of an ocean wave energy converter to maximize extracted power. A generic heaving converter that can have both linear dampers and active elements as a power take-off system is considered and an efficient optimal control algorithm is developed for use within a receding horizon control framework. The optimal control is also characterized analytically. A direct transcription of the optimal control problem is also considered as a general nonlinear program. A variation of the projected gradient optimization scheme is formulated and shown to be feasible and computationally inexpensive compared to a standard nonlinear program solver. Since the system model is bilinear and the cost function is not convex quadratic, the resulting optimization problem is shown not to be a quadratic program. Results are compared with other methods like optimal latching to demonstrate the improvement in absorbed power under irregular sea condition simulations.

In the second part, robust estimation of the radiation forces and states inherent in the optimal control of wave energy converters is considered. Motivated by this, low order $H_\infty$ observer design for bilinear systems with input constraints is investigated and numerically tractable methods for design are developed. A bilinear Luenberger type observer is formulated and the resulting synthesis problem reformulated as that for a linear parameter varying system. A bilinear matrix inequality problem is then solved to find nominal and robust quadratically stable observers. The performance of these observers is compared with that of an extended Kalman filter. The robustness of the observers to parameter uncertainty and to variation in the radiation subsystem model order is also investigated.

This thesis also explores the numerical integration of bilinear control systems with zero-order hold on the control inputs. Making use of exponential integrators, exact to high accuracy integration is proposed for such systems. New a priori bounds are derived on the computational complexity of integrating bilinear systems with a given error tolerance. Employing our new bounds on computational complexity, we propose a direct exponential integrator to solve bilinear ODEs via the solution of sparse linear systems of equations. Based on this, a novel sparse direct collocation of bilinear systems for optimal control is proposed. These integration schemes are also used within the indirect optimal control method discussed in the first part.

*To my family.*

# Acknowledgements

I would like to thank my supervisor Dr Eric Kerrigan for his guidance, support and inspiration through the course of my PhD. I would like to thank him for introducing me to many a fascinating subject within control and the computational sciences. He has been an excellent teacher who also allows you take your own routes and learn self-reliance. Most importantly, he has been a great source of knowledge, motivation and focus precisely at the times I needed them most. I would also like to express my sincere gratitude for his invaluable feedback both on the technical aspects of my work and on writing better manuscripts.

I would like to thank my colleagues from Eric's research group for their company and the pleasure to learn from their work. I am also grateful for their feedback on my research during group meetings. In particular, I would like to thank Paola Falugi, Shakil Ahmed, Bing Feng Ng, Juan Jerez, Andrea Suardi, Ammar Hasan, Stefano Longo, Zhe Feng and Ge Ming. Special thanks go to Zhe Feng for help with proofreading some parts of the thesis. I am also thankful to my past and present Aeronautics colleagues whose good company and discussions have been enjoyable.

Finally, I express my deepest gratitude to my family for their support and love.

# Contents

**Abstract** 3

**Acknowledgements** 5

**Contents** 6

**List of Figures** 9

**Abbreviations** 12

**1 Introduction** 13
- 1.1 Motivation . . . 13
- 1.2 Thesis Layout . . . 16
- 1.3 Related Publications . . . 18

**2 Ocean Wave Energy Conversion: Modelling and Control** 19
- 2.1 Gravity Surface Water Waves . . . 19
  - 2.1.1 Linear waves . . . 21
  - 2.1.2 Wave spectra . . . 22
- 2.2 Modelling WECs for Control . . . 22
  - 2.2.1 Wave-body interactions . . . 23
  - 2.2.2 State-space models for control . . . 25
  - 2.2.3 Modelling the wave excitation force . . . 27
- 2.3 Review of Existing Control Methods for WECs . . . 29
- 2.4 Short-Term Wave Prediction . . . 32

**3 Optimal Control and Optimization of a Wave Energy Converter** 34
- 3.1 Introduction . . . 34
- 3.2 System Dynamics . . . 38
- 3.3 Optimal Control Problem . . . 40
  - 3.3.1 The optimal control problem . . . 40
  - 3.3.2 Analysis of singular arcs . . . 43
  - 3.3.3 Optimal control algorithm: a gradient projection scheme . . . 45

3.3.4 Sufficient conditions of optimality and state inequality constraints . . . 49
3.3.5 Direct transcription and interior-point solvers . . . 50
3.3.6 Receding horizon control implementation . . . 53
3.4 Example Simulations . . . 55
3.4.1 Projected gradient algorithm performance . . . 57
3.4.2 Device optimization for control . . . 59
3.4.3 Prediction horizon sensitivity in a receding horizon implementation . . 60
3.5 Conclusion . . . 62

**4 Robust Estimator Design for Bilinear Systems with Bounded Inputs 65**
4.1 Introduction . . . 65
4.2 Problem Statement and Related Work . . . 67
4.3 Quadratic Stability and Performance for LPV systems . . . 70
4.3.1 Linear parameter varying systems . . . 72
4.3.2 Conservativeness of the PLDI representation . . . 73
4.4 Lower-order $H_\infty$ Filter for Bilinear Systems . . . 74
4.5 Solving the BMI problem (4.19) . . . 77
4.5.1 LMI-based algorithm and initial robustly feasible observer . . . 79
4.5.2 An upper bound on the optimality gap for Algorithm 2 . . . 79
4.6 Continuous-time Extended Kalman Filter . . . 81
4.7 Example Simulations . . . 82
4.7.1 Observer-based MPC and closed-loop receding horizon simulations . . 86
4.8 Conclusion . . . 89

**5 Matrix Exponential Methods for Integration of Bilinear Control Systems 92**
5.1 Exponential Integrators: Avoiding $\varphi$ Functions . . . 93
5.2 Exponential Integrators for Bilinear Systems . . . 95
5.2.1 Exact solution of bilinear control systems with ZOH . . . 95
5.3 Exponential Integrators with only Matrix-Vector Multiplication . . . 97
5.3.1 The Scaling and Squaring method for computing $e^X$ . . . 97
5.3.2 An efficient scheme by Al-Mohy and Higham [113] for computing $e^X y$ 100
5.3.3 A priori bounds on computational complexity for exponential integration of a bilinear system . . . 104
5.3.4 A Krylov subspace method for computing $e^X y$ . . . 106
5.3.5 Explicit classical Runge-Kutta methods . . . 110
5.4 Solving Bilinear ODEs via Sparse Linear Systems of Equations . . . 111
5.5 Example Simulations with Bilinear Systems . . . 119
5.6 Exponential Collocation for Direct Transcription . . . 126
5.7 Conclusion . . . 132

**6 Conclusions and Future Work 133**

6.1 Contributions . . . . . 133
6.2 Future Work . . . . . 136

**Appendices** **138**
A Adjoint system dynamics (3.12) for the OCP in (3.8) . . . . . 138
B Exact Solution of Semilinear Initial Value Problems with polynomial in time nonlinearity . . . . . 138
C Padé Approximation for $e^X$ . . . . . 139
D Bounds on Computational Cost of Runge-Kutta and Exponential Integrators . . 140

**References** **142**

# List of Figures


1.1 Some examples of WECs [8] . . . . . 15

2.1 Water wave propagating along the $x$-axis . . . . . 20
2.2 A heaving body in water . . . . . 23
2.3 (a) The oscillating WEC as a system with inputs $f_{exc}, f_c$ and position states $s$ and/or its derivatives as output (b) The subsystem $H_f$ is augmented to $H$ so that wave height $\eta(\cdot)$ is the input and not $f_{exc}(\cdot)$. . . . . 28
2.4 Latching control: the buoy is latched when its velocity is zero (red curve) and released at a favourable time so that it is in phase with the excitation force (blue curve). . . . . 30

3.1 A schematic of a heaving buoy point absorber WEC. . . . . 39
3.2 Receding horizon control . . . . . 55
3.3 (a) A radiation subsystem of order $n_r = 5$ appears good enough to approximate the radiation impulse response sufficiently. (b) A 6th order subsystem can be used to model the "causalized" excitation force; $n_{exc}$ is the order of the linear subsystem that approximates the excitation force response. . . . . 56
3.4 Convergence of the PGM algorithm (dotted, left) against that of a direct transcription solution using IPOPT (lines with marker) over a 50 s prediction horizon (a) for waves of the same typical period ($T_p$) and different significant wave heights($H_s$) (b) for waves of different typical period ($T_p$) but of the same significant wave height. . . . . 58
3.5 A plot of the device velocity against control commands and normalized excitation force under 'Method 1'. The blue plot switching between +1 and -1 is the active control. The green plot switching between 0 and +2 is the damping control times 2; it is multiplied by 2 for clarity of presentation. . . . . 59
3.6 Variations in average absorbed power (W) against parameters $G$ and $B_{pto}$ for: (a) Method 1 (b) Method 2, $T_p = 8$ s, $H_s = 2$m. . . . . 61
3.7 Average absorbed power against typical wave period with the different control methods, $B_{pto} \approx 280$ kN·s/m ($B_{pto} \approx 95$ kN·s/m for latching control), $G = M + \mu_\infty$, $T_p = 8$ s, $H_s = 2$m. . . . . 62

3.8 Variations in average absorbed power against prediction horizon lengths for 'Method 1'. JONSWAP wave with $T_p = 8$ s and $H_s = 2$ m used; $G = B = 0.3 * (M + \mu_\infty)$. . . . . 63

4.1 Convergence of $\gamma^{(k)}$ in synthesis via Algorithm 1. The lower (blue) line shows the best full-order filter performance as a bound for $\gamma^{(k)}$. . . . . 84
4.2 Displacement and velocity and their noisy measurements . . . . . 84
4.3 Comparison of radiation force estimates by the EKF and robust $H_\infty$ filters; $P(0) = I$, and $\hat{x}(0) = 0$. . . . . 85
4.4 Estimates of radiation force with a different initial state guess; $P(0) = I$, and $\hat{x}(0) = [1\ -1\ 0\ 0\ 0]^T$ . . . . . 86
4.5 Velocity estimates with observers based on radiation subsystems of order $n_{rad}$ equal to1 and 3 for the $H_\infty$ filter, and an EKF based on a 3rd order radiation subsystem. . . . . 87
4.6 Radiation force estimates with observers based on radiation subsystems of order $n_{rad}$ equal to1 and 3 for the $H_\infty$ filter, and an EKF based on a 3rd order radiation subsystem. . . . . 88
4.7 Cumulative extracted energy of observer-based receding horizon control. The order of the radiation subsystem for observer design is $nrad$. $T_p = 8$ s and $H_s = 2$ m used; $G = B = 0.3 * (M + \mu_\infty)$. . . . . 90

5.1 The sparsity pattern for the sparse dynamic matrix from a bilinear PDE and the sparsity of the matrix $D_l(2^{-s}X)$, $l = 13$ solved by the Padé approximation. . . . 100
5.2 The ZOH length can be many multiples of the integration step length. A snippet from a WEC simulation showing the velocity ($x$ here) against bilinear controller ($u$ here): $M_N = 100/10 = 10$. . . . . 114
5.3 The sparsity structure of $\boldsymbol{A}(\cdot)$ in (5.45): wave energy system model simulated with parameters $n = 5$, $m = 2$, $\text{nnz}(A) = 18$; $N = 10$, $M = 2$ . . . . . 115
5.4 The sparsity structure of $\boldsymbol{A}(\cdot)$ in (5.57): wave energy system model simulated with parameters $n = 5$, $m = 2$, $\text{nnz}(A) = 18$; $N = 10$, $M = 2$ . . . . . 118
5.5 Convergence of cost and gradient with integration step: (a) cost function (b) norm of gradient of cost with input sampling time of 0.1 and $T_h = 12$. . . . . 120
5.6 Logarithmic plots of the forward error of the solution against computational cost of three numerical methods in solving the bilinear system. The keys 'KRY', 'EXP', and 'RK' stand for the Krylov integrator (using the algorithm `expmvp` from [126]), adapting the method of [113] and the 4th order Runge-Kutta method, respectively: (a) the $x-$axis shows total CPU time of integration (b) the size $n_x$ of the linear systems solved by the direct EXP and RK methods are shown on the $x-$axis. . . . . 121
5.7 A repeat of the experiment in Figure 5.6 with the tolerance for 'EXP' set to IEEE single precision eps ($2^{-24}$) and all other parameters the same. . . . . 121

5.8 Computational cost of integration schemes in a closed-loop simulation. . . . . 122
5.9 Simulation profiles of a metal slab with two sides being cooled by two different control inputs. . . . . 124
5.10 Forward error of solution against computational cost of three numerical methods in solving the controlled metal cooling bilinear system. (a) the x-axis shows total CPU time of integration (b) the size of the linear systems solved by the direct EXP and RK methods are shown on the x-axis. . . . . 125
5.11 The sparsity structure of the Hessian when RK and EXP collocation schemes are used. . . . . 129
5.12 Comparison of the IPM solutions with the analytic solution for [135, Example. 3.2]. . . . . 131
5.13 Comparison of the IPM solutions with the analytic solution for [135, Example. 3.3]. . . . . 131

# Abbreviations

| | |
|---|---|
| WEC: | Wave Energy Converter |
| PTO: | Power Take-Off |
| MPC: | Model Predictive Control |
| ODE: | Ordinary Differential Equation |
| PDE: | Partial Differential Equation |
| LTI: | Linear Time Invariant |
| LPV: | Linear Parameter Varying |
| LQR: | Linear Quadratic Regulator |
| LMI: | Linear Matrix Inequality |
| BMI: | Bilinear Matrix Inequality |
| IPM: | Interior Point Method |
| RK: | Runge-Kutta Method |

# Chapter 1

# Introduction

This thesis is concerned with the optimal control of ocean wave energy converters for the maximization of power generation. This is vital to significantly improve the economic prospects for wave power and make it a viable contributor to the renewable energy mix. In addition to developing an algorithm for the synthesis of optimal controllers for wave energy converters, we solve some associated robust estimation problems. Here, we motivate the premise of the thesis; the technical background and relevant literature review are presented in the next chapter.

## 1.1 Motivation

The growing global need for energy and the urgent necessity to tackle climate change have motivated the much more rapid development of renewable energy technology in the last few decades. Modern biofuels, solar, wind, hydro and geothermal energy together constitute about 8.2% of the world energy production and are growing rapidly in capacity. In 2011 alone, of the approximately 208 GW of additional global electricity capacity, renewables contributed almost half [1]; for the European Union the renewable contribution goes up to 71%. However, one major renewable source remains untapped—ocean wave energy. After decades of dormancy, European and global targets to generate a higher percentage of 'clean energy' and the accumulation of knowledge within the field of ocean engineering have made ocean wave energy an active area of research again.

Ocean waves have the highest energy density (per unit area) of all renewable energy sources [2]. The total wave power that can be generated around the coasts of the world is of the order of 1 TW, similar to the current global electricity consumption [2, 3]. Moreover, harnessing the more powerful offshore waves may increase this by an or-

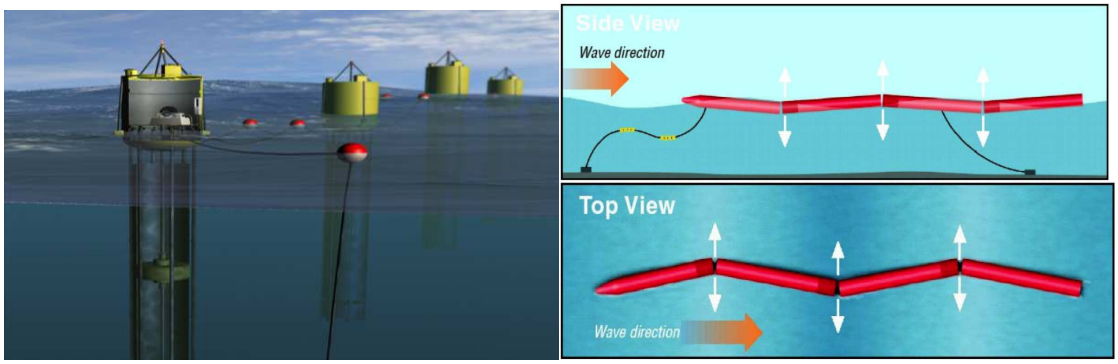
der to roughly the same level as the global total power consumption ($\sim$ 10 TW) [3]. In addition, recent research has also shown that nearshore waves are only marginally less energetic compared to deepwater waves [4]. This alleviates the possible technical demands necessitated by a need to go farther out to deeper waters like some offshore wind farms. For these reasons, the second generation of ocean wave energy converters (WECs), are primarily oscillating bodies planned for nearshore installation some kms from the shore (with depths of less than 50m) [5]. This is in contrast to the first generation of Oscillating Water Column devices built on shore, which absorb energy by coupling the ocean water oscillation to a gas turbine through the compression and expansion of a trapped air chamber.

**Wave Energy Converters**

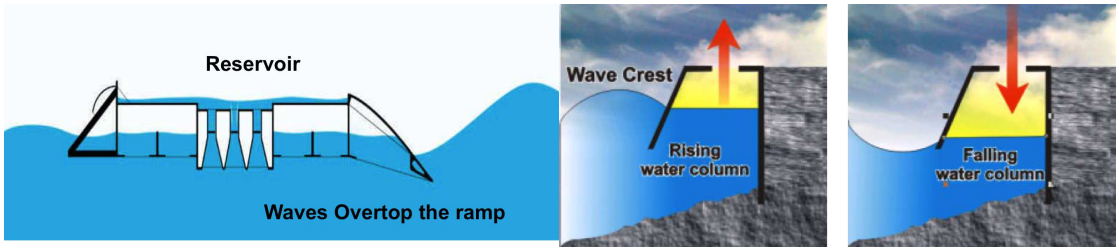
Over the last three decades, a variety of technologies have been developed with the aim to tap into this abundant renewable energy. In the process of conversion, the first stage is the transfer of wave energy to the mechanical (kinetic and/or potential) energy of an oscillating system. Despite the variety in the actual mechanics of the WECs, most are oscillating structures that resonate through the water-body interaction in or on ocean waves, which then drive a power take-off (PTO) mechanism. Some are floating or submerged solid oscillating body structures, oscillating water within a fixed structure or some membrane, or even an overtopping device like the Wave Dragon that elevates ocean water to a higher potential reservoir [2, 3, 6].

The next step of conversion involves changing the mechanical energy of the oscillating WEC to a useful form such as electricity or pressurized water. This stage is called the power take-off (PTO) mechanism and is to a large degree dependent on the kind of oscillating system used. For example, a heaving buoy is well suited to drive a linear generator or a high-pressure water pump (i.e. damper), whereas an oscillating water column could drive air turbine generators [7].

There are various ways of classifying WECs. Two of the main parameters of classification are device location (Onshore, Nearshore or Offshore) and physical dimensions (Point Absorbers, Attenuators or Terminators). These methods have some degree of overlap and additional qualifiers can be used for more clarity; for example, the PELAMIS [2, 6] is a nearshore attenuator device. Point absorber devices are usually symmetric about the vertical axis and have small horizontal dimensions in comparison to the wavelength of the ocean waves. Attenuators and terminators are collectively called line absorbers and have one large dominant horizontal dimension relative to the wavelength. An attenuator has its main axis oriented parallel to the direction of the wave propagation, whereas terminators have their axes normal to it. Some examples are shown in Figure 1.1. Sometimes devices are also described by the means of operation




(a) Point Absorber (b) Attenuator




(c) Overtopping (d) Terminator

Figure 1.1: Some examples of WECs [8]

of the oscillating system and PTO mechanism. For example, the LIMPET [2, 6] uses an oscillating water column driving a Wells (air) turbine.

As mechanical oscillators, these devices perform well only in sea conditions where the wave frequency is matched with their natural frequencies. Without dynamic control, the absorbed power would diminish significantly as the frequency of the ocean waves varies. Therefore, optimal control of WECs are vital to the future of wave energy and motivate this thesis. Although not the subject of this thesis, we briefly discuss the role that ocean wave energy can play in also powering desalination plants.

### Wave Energy for Desalination

A shortage of clean water for irrigation and human consumption is also as big and urgent a problem as climate change is for the world; 30% of global fatality is directly linked to this [2, p. 261]. As a result, many different desalination technologies have been developed and are currently growing in use. Thermal and physical (or membrane) processes are two broad means of separating salt from water. Although thermal distillation processes need little maintenance and produce high quality water, they are energy intensive ($\sim$ 10 KWh/m$^3$ of water produced [2]). The only physical process currently in use for desalination is Reverse Osmosis (RO) and is more efficient ($2-3$ KWh/m$^3$ [2, 9]). A semi-permeable membrane separates two solutions with a different concentration while

an osmotic pressure differential is created across the membrane by pumping the salty feed water at high pressure. Advances in membrane and energy recovery technologies in the desalination process, and its scalability have made RO desalination commercially feasible and promising [2, 3].

Both the high energy requirements and the remote nature of feed water sources from the grid had motivated many studies into the use of renewables in desalination. Renewable sources such as solar, wind, hydrostatic pressure and wave power have been investigated [2, 3]. Although most wave energy devices are designed with direct wave to electrical energy conversion in mind, most lend themselves to powering RO desalination plants by generating high-pressure feed water. Moreover, the presence of both the energy source and the feed water at the same site is another advantage. As a result, successful implementations and feasibility have been shown and some are at pre-commercial stages. The Delbuoy system is an example [2, 9, 10].

**Notation**

In the rest of this chapter, we outline the content of the subsequent chapters. Since the three main Chapters, Chapters 3, 4 and 5, cover different topics, it has not been possible to keep one consistent notation throughout the thesis. Where possible, the utmost attempt has been made to keep notation in each chapter consistent with literature. Therefore, the reader is reminded that each of these chapters have their own distinct notation, defined locally wherever necessary.

## 1.2 Thesis Layout

In Chapter 2, we introduce some technical background and literature review. We introduce gravity water waves and their linear approximation for wave energy applications. The modelling of wave energy converters for control will also be outlined considering a buoy moving in heave only. The derivation of state space models from hydrodynamic data will also be surveyed. We then review literature on existing control methods for WECs. Reactive and latching control methods will be discussed and the motivation for advanced optimal control methods outlined.

**Chapter 3: Optimal Control and Optimization of a Wave Energy Converter**

This chapter investigates model predictive control schemes applied to a point absorber wave energy converter. The system dynamics for a heaving buoy will be given. An indirect method is used to characterise the optimal solution of an energy maximization problem. To solve the optimal control problem, a computationally inexpensive variation of the projected gradient optimization scheme is shown to be feasible and computation-

ally inexpensive compared to an interior point solver. Example simulations are used to compare the proposed model predictive control with optimal latching. Simulations are also used to show how performance scales with active and damping control parameters. By implementing the controllers in a receding horizon fashion, the sensitivity of the controllers to prediction horizon length is assessed under irregular wave condition. The main computational cost of the proposed algorithm is the integration of bilinear systems, which motivates the analysis of integration schemes for bilinear systems discussed in Chapter 5. The motivation for robust observer design for radiation forces will also be stated and left as the subject of Chapter 4.

**Chapter 4: Estimator Design for Input-Constrained Bilinear Systems**

Motivated by dynamic estimation of radiation forces in control of WECs, Chapter 4 investigates observer design for bilinear systems with input constraints and associated synthesis algorithms. A discussion of linear parameter varying systems and quadratic stability and performance for polytopic linear differential inclusions will be presented. For a general bilinear system, the constrained nonlinearizing inputs are posed as linear parameters to get equivalent polytopic descriptions. We then pose an $H_\infty$ filter design problem using lower order models but with performance guarantees around the full-order model. The filter synthesis is posed as a nonconvex bilinear matrix inequality (BMI) problem. Algorithms for solving the BMI problem are discussed. A local LMI-based algorithm with some optimality gap bound is given. Example simulations with the WEC system of Chapter 3 are then used to assess the performance of the $H_\infty$ observer within a receding horizon control scheme. The $H_\infty$ filters are also compared with the Extended Kalman filter.

**Chapter 5:Matrix Exponential Methods for Integration of Bilinear Control Systems**

In Chapter 3, the use of an indirect method for the computation of an optimal controller was proposed. The main computational cost of the projected gradient method is the integration of bilinear systems. In Chapter 5, the use of exponential integrators for efficient solution of bilinear control systems will be detailed. It is shown that bilinear control systems with zero-order hold on the input can be integrated exactly to a high prescribed precision by computing the action of the matrix exponential on a vector. New results in the literature for efficient computation of the latter and their a priori error guarantees are exposed. Based on these, new a priori error bounds on the computational complexity of integrating bilinear systems with a zero-order hold are derived. We also look at the classical Runge-Kutta scheme and Krylov methods as alternative sparse methods for bilinear system integration.

We also propose a direct exponential integrator to solve bilinear ODEs via the solution

of a sparse linear system. The sparsity and computational complexity in solving the resulting linear system are analysed using our new bounds. It is then compared with a similarly implemented sparse fourth-order explicit Runge-Kutta scheme. Numerical experiments are also used to assess the advantages of the exponential integrators compared with the classical Runge-Kutta method. The method is applied to the bilinear system arising in the WEC control problem. A PDE heat transfer model for the controlled cooling of a metal slab is a second example used. Finally, the exponential integrator with our new bounds is used in a novel sparse direct transcription of optimal control problems with bilinear dynamics. Example optimal control problems are used to demonstrate the feasibility of the proposed approach.

In Chapter 6, we summarise the main results presented in this thesis along with contributions. Ideas for future work based on the results are also suggested.

## 1.3 Related Publications

Much of the work and results presented in this thesis are mostly based on the contents of the following publications:

- E. Abraham and E. C. Kerrigan, "Optimal Active Control and Optimization of a Wave Energy Converter," IEEE Transactions on Sustainable Energy, 2012.
- E. Abraham and E. C. Kerrigan, "Optimal Active Control of a Wave Energy Converter," In *Proc.* 51$^{st}$ *IEEE Conference on Decision and Control*, December 10 – 13, pp. 2415-2420. IEEE, 2012.
- E. Abraham and E. C. Kerrigan. "Estimator design for input-constrained bilinear systems with application to wave energy conversion," in Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on, Accepted.
- E. Abraham and E. C. Kerrigan, Robust estimator design for bilinear systems with bounded inputs, IEEE Transactions on Signal Processing, in preparation.

# Chapter 2

# Ocean Wave Energy Conversion: Modelling and Control

Although ocean waves have the highest energy density of all renewable energy sources, current wave energy conversion technology is not yet competitive with other renewables. Since power output from WECs can be significantly increased through the use of dynamic control, optimally controlled WECs are vital to the future of wave energy. In this chapter, we will introduce wave energy conversion and some relevant background, the WEC mathematical models and a review of existing control schemes.

## 2.1 Gravity Surface Water Waves

Ocean waves are generated by winds passing over the surface of the water body. These waves, whether created by local winds or created over longer ranges, can travel long distances without losing energy. It is this mechanical energy of ocean waves that wave energy conversion changes into a useful form. This is done through the interaction of waves with mechanical systems.

Like all fluid flows, the modelling of ocean water waves starts from the two basic equations that express conservation of mass and momentum. Let $\mathbf{v}(x,y,z,t)$ define the flow velocity vector at a point in the fluid $(x,y,z)$ and time $t$; the $y$ axis is into the page in Figure 2.1. Two hydrodynamic equations need to be satisfied everywhere in the fluid [11–13]. These are the principles of conservation of mass (continuity equation)

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \tag{2.1}$$

and conservation of momentum (Navier-Stokes equation)

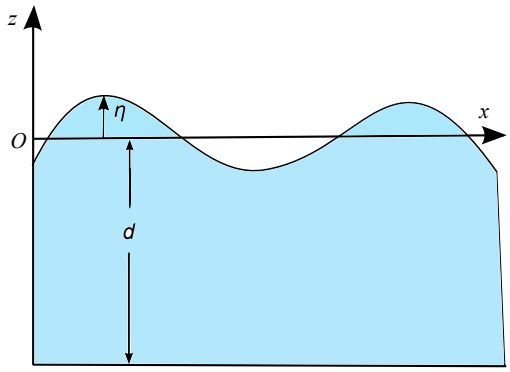



Figure 2.1: Water wave propagating along the $x$-axis

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{v} = -\frac{\nabla p}{\rho} + \nu \nabla^2 \mathbf{v} + \frac{\mathbf{f}}{\rho}. \tag{2.2}$$

Here $\rho$ is the mass density of the fluid, $p$ is the pressure in the fluid, and $\nu := \mu/\rho$ is the kinematic viscosity, where $\mu$ is the viscosity coefficient of the fluid. The fluid will be assumed inviscid ( $\nu \approx 10^{-6}\mathrm{m}^2\mathrm{s}^{-1}$ for water at 15°C, [13, p. 28]) and $\nu$ is set to zero in (2.2). The force acting per unit volume of fluid is represented by $\mathbf{f}$. Neglecting the effects of surface tension, we consider only gravitational force, $\mathbf{f} = \rho\ [0, 0, -g]^T$ with $g$ the gravitational acceleration.

Making the reasonable assumption that water is incompressible, that is $\partial\rho/\partial t = 0$, we get from (2.1) the condition $\nabla \cdot \mathbf{v} = 0$. *Irrotationality* of this ideal fluid flow ($\nabla \times \mathbf{v} = 0$) enables us to re-write the velocity as $\mathbf{v} = \nabla\phi$; $\phi$ is the so-called velocity potential. We denote the free surface of the fluid $z = \eta(x,y,t)$, where $\eta(\cdot)$ represents the water surface elevation from the mean free surface (that is, from the reference $z = 0$). By requiring that the particles at the free surface remain at the free surface as it deforms, i.e. the *kinematic condition* $z - \eta(x,y,t) = 0$, one can show the set of equations below define ocean gravity waves [11–13]. Assuming the bottom of the medium is of a uniform depth and flat or far away as in deep ocean, we can define ocean gravity waves by the set of equations:

$$\nabla^2\phi = 0, \quad \forall z,\ -d \le z \le \eta(x,y,t), \tag{2.3a}$$

$$\frac{\partial \phi}{\partial t} = -\frac{1}{2}\nabla\phi \cdot \nabla\phi - g\eta, \quad \forall z, = \eta(x,y,t), \tag{2.3b}$$

$$\frac{\partial \eta}{\partial t} = \frac{\partial \phi}{\partial z} - \nabla\phi \cdot \nabla\eta, \quad \forall z = \eta(x,y,t), \tag{2.3c}$$

where $\phi := \phi(x,y,z,t)$, $\eta := \eta(x,y,t)$, and $x$ and $y$ are defined on some domain of interest with appropriate boundary conditions. In finite-depth waters, the no penetration condition at the bottom adds to the above set of equations the boundary condition $\frac{\partial \phi}{\partial z} = 0$, $\forall z(x,y) = -d(x,y)$; $d(\cdot)$ can vary significantly with location $(x,y)$ in shallower

waters.

### 2.1.1 Linear waves

In modelling wave energy conversion, it is universally assumed that the wave amplitude is sufficiently small to apply linear wave theory. This is not an unreasonable assumption since control of WECs is sought in calmer wave conditions that are approximately linear [11]. We assume that the dynamic variables $\phi$, $\eta$ and all their derivatives are 'small'. By this we mean, $(\frac{\partial\phi}{\partial x})^2+(\frac{\partial\phi}{\partial y})^2+(\frac{\partial\phi}{\partial z})^2$ is negligible compared to the term $g\eta$ and we can similarly neglect product terms like $(\frac{\partial\phi}{\partial x})(\frac{\partial\eta}{\partial x})$ in (2.3c). The linearized set of equations we use are:

$$\nabla^2\phi = 0, \quad \forall z,\ -d \le z \le \eta(x,y,t), \tag{2.4a}$$

$$\frac{\partial\phi}{\partial t} = -g\eta, \quad \forall z = \eta(x,y,t), \tag{2.4b}$$

$$\frac{\partial\eta}{\partial t} = \frac{\partial\phi}{\partial z}, \quad \forall z = \eta(x,y,t), \tag{2.4c}$$

where $\phi$, $\eta$, $x$, and $y$ are as defined in (2.3).

We consider, for simplicity of notation, the two-dimensional case where there is no variation in the y-axis; $\frac{\partial\phi(\cdot)}{\partial y}=0$. By seeking a sinusoidal solution $\eta(x,t)=a_\eta\sin(kx-\omega t)$ and substituting for $\eta(\cdot)$ in (2.4c), reveals that the velocity potential should have the form $\phi=e(z)\cos(kx-\omega t)$, where $\omega$ is the temporal frequency and $k$ is the wave number. Since $\phi(\cdot)$ satisfies Laplace's equation (2.4a), the function $e(z)$ must satisfy:

$$\frac{\partial^2 e}{\partial z^2}(z) - k^2 e(z) = 0. \tag{2.5}$$

With the general solution $e(z)=ae^{kz}+be^{-kz}$, $k>0$, and applying the condition that the velocity potential should be bounded in the deep water approximation, $z\to-\infty$, we get the solution

$$\phi(x,z,t) = a_\phi e^{kz}\cos(kx-\omega t).$$

Substituting the solutions for $\eta$ and $\phi$ into (2.4b) and (2.4c), we solve for the unknown $a_\phi$ to get the solution

$$\phi(x,z,t) = \frac{a_\eta\omega}{k}e^{kz}\cos(kx-\omega t), \tag{2.6}$$

and the dispersion relation

$$\omega^2 = gk. \tag{2.7}$$

In light of this simple solution, the nonlinear terms in (2.3b) and (2.3c) reveal that

the linearity approximation is equivalent to the condition that the amplitude $a_\eta$ be small compared to the wavelength $\lambda = 2\pi/k$ [12]. Such a wave is also called *long-crested*.

### 2.1.2 Wave spectra

Assuming linear wave theory, a real sea state is described as a superposition of different harmonic components. The total average energy stored per unit sea surface area can be calculated using:

$$E = \rho g \overline{\eta^2(x,y,t)} = \rho g H_{m0}^2/16 = \rho g \int_0^\infty S(f)df \tag{2.8}$$

where $H_{m0}$ is called the significant wave height [3, 11]. The bar over the wave height $\eta(\cdot)$ indicates averaging over a unit area and over time. $S(f)$ is called the energy spectrum. It is a function of the wave heights at each frequency and has units $m^2$/Hz [11, sec. 4.5]. A different description of the wave spectrum $S(f)$ would also take account of the direction of incidence (the angle between the wave propagation direction and the $x$ axis) of each harmonic component; $S := S(f,\beta), -\pi \leq \beta \leq \pi$. Note that this is a temporal description of an ocean wave at a fixed point in space. Two semi-empirical spectra are mainly used to describe waves at various locations—the Pierson-Moskowitz (PM) and JONSWAP (Joint North Sea Wave Project) spectra. The PM spectrum, for example, is given as

$$S(f) = \frac{5H_{m0}^2 f_p^4}{16 f^5} \exp\left(-\frac{5}{4}\left(\frac{f_p}{f}\right)^4\right), \tag{2.9}$$

where $f_p$ is the peak frequency with a corresponding typical wave period, $T_p := 1/f_p$. This period varies with time and location usually in the range 6–15 seconds; this amounts to about an average of 50 KW of wave power per meter width of wavefront for the coastlines of northwest Europe [2].

## 2.2 Modelling WECs for Control

Among the many existing wave energy conversion technologies, point absorbers are the most studied and prime candidates for scaling up in wave energy farms with arrays of converters [2]. These axisymmetric devices can absorb waves from all directions and are preferred in the form of heaving semi-submerged buoys [14]. Therefore, the modelling of WECs for control will be introduced by focusing on a generic point absorber moving with one degree of freedom, namely heave.

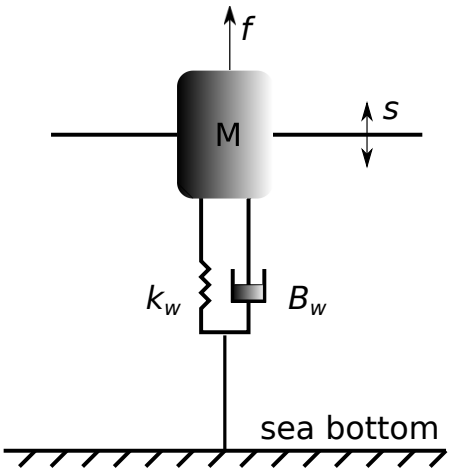



Figure 2.2: A heaving body in water

### 2.2.1 Wave-body interactions

Within the ocean engineering literature, most analysis and modelling is done in the frequency domain. To illustrate this *mechanical impedance analysis* of hydrodynamic systems [11], we will consider the motion of a heaving body in water. As shown in Figure 2.2, the net buoyant (or restoring) and dissipative forces are represented by a spring and damper, respectively. Now, if an external vertical force $f$ acts on the body of mass $M$, $s$ being the displacement from its equilibrium position, the equation of motion for the body, derived from Newton's second law of motion, is

$$M\ddot{s} + B_w\dot{s} + k_w s = f, \tag{2.10}$$

where the hydrostatic stiffness (or coefficient of the hydrostatic buoyancy force) $k_w$ is non-negative and $B_w$ represents viscous and other dissipative losses. The body's movement in water generates waves that radiate away. The effect is often called *wave radiation* and the waves radiated in turn exert a 'reaction' force on the body through their interaction. This force is called the radiation force and here we use $f_r$ to denote it. The external force $f$ is now a sum of the radiation force $f_r$ and other external excitation forces $f_e$. We can now rewrite (2.10) as:

$$M\ddot{s} + B_w\dot{s} + k_w s = f_e + f_r. \tag{2.11}$$

It is assumed that the external excitation force $f_e$ is harmonic (i.e. $f_e = \mathrm{Re}(F_e e^{j\omega t})$). The further assumption that the interaction between the water and the heaving body is linear will imply that the radiation force is also harmonic and with the same frequency in steady state (i.e. $f_r = \mathrm{Re}(F_r e^{j\omega t})$). For linear systems, we can define the mechanical impedance as the ratio of the Fourier transform of an external force to the Fourier trans-

form of the velocity response of the system $\hat{u}(j\omega) = j\omega \cdot s(j\omega)$ (over all frequencies where the linear time-invariance of the system is valid) [11, 15]. Taking the Fourier transform of (2.11), we get:

$$j\omega M\hat{u}(j\omega) + [B_w(\omega)]\hat{u}(j\omega) + \frac{k_w}{j\omega}\hat{u}(j\omega) = \hat{F}_e(j\omega) + \hat{F}_r(j\omega). \tag{2.12}$$

Now let $\hat{F}_r(j\omega) := -Z_r(j\omega) \times \hat{u}(j\omega)$ where $-Z_r(j\omega)$ is the radiation force impedance. The radiation impedance is a property of the WEC system and depends on the geometry of the body [11, 15, 16]. In general, $Z_r(j\omega)$ is a complex function of $\omega$ and can be written as:

$$Z_r(j\omega) := H_r(\omega) + jX_r(\omega), \tag{2.13}$$

where $H_r(\omega)$ and $X_r(\omega)$ are real functions of $\omega$. Often $X_r(\omega)$ is written as a function of the so-called added mass $M(\omega)$ [11, 16–18], $X_r(\omega) := \omega M(\omega)$. The term $B(\omega) := B_w(\omega) + H_r(\omega)$ is called the potential damping and represents the net damping effect on the body. Using these (2.12) can be re-written as:

$$j\omega[M + M(\omega)]\hat{u}(j\omega) + B(\omega)\hat{u}(j\omega) + \frac{k_w}{j\omega}\hat{u}(j\omega) = \hat{F}_e(j\omega). \tag{2.14}$$

In the wave energy and marine structures community, these frequency dependent added mass and damping coefficients are calculated, in some finite set of frequencies of interest, using commercial hydrodynamic software like WAMIT, DIODORE, and AQUAPLUS [5, 16–19]. Using what is called the infinite-frequency added mass, $\mu_\infty := \lim_{\omega\to\infty} M(\omega)$, (2.14) simplifies to

$$j\omega[M + \mu_\infty]\hat{u}(j\omega) + [B(\omega) + j\omega[M(j\omega) - \mu_\infty]]\hat{u}(j\omega) + \frac{k_w}{j\omega}\hat{u}(j\omega) = \hat{F}_e(j\omega). \tag{2.15}$$

Now let $K(j\omega) := B(\omega) + j\omega[M(j\omega) - \mu_\infty]$ and represent its inverse Fourier transform $k(t)$. The time domain representation of (2.15) is:

$$(M + \mu_\infty)\ddot{s}(t) + \int_{-\infty}^{t} k(t-\tau)\dot{s}(\tau)d\tau + k_w s(t) = f_e(t), \tag{2.16}$$

where the integration performs the convolution operation. Note that the impedance analysis and the expressions in (2.14) and (2.16) can be easily extended to a full six degrees of freedom motion by using matrix representation with cross-coupling between the different degrees of freedom as in examples considered in [11, 20]. For example, if all six modes of oscillation (surge, sway, heave, roll, pitch, yaw) are considered, the parameters would be 6x6 matrices and $s(t)$ and $f_e(t)$ become vectors of length 6. For the rotational modes, some elements of the mass matrix would represent corresponding

moments of inertia.

The term $k_w s(t)$ represents net bouyancy spring forces. When the submersed buoy is at equilibrium, the mass of the body is balanced by buoyancy forces due to displaced water. That is, $Mg = \rho g V(t)$, where $V(t)$ is the submersed volume at time $t$. Therefore, as the body heaves, the net bouyancy force is represented by $f_b = -\rho g \Delta_V(t)$. However, as is practice in linear modelling, the buoyancy force is linearized by assuming a constant waterline area for small waves and small relative motion; that is $f_b(t) = -\rho g a_w s(t)$, where $a_w$ is the constant cross-sectional area of the heaving buoy. The relevance of such a model is extensively assessed in [21].

### 2.2.2 State-space models for control

A time-domain approach makes use of the integro-differential equation of (2.16), also called the Cummins equation [22], since Cummins was the first one to make use of such a representation in the ocean engineering community. Let us consider again the WEC moving in heave only. The external force $f_e$ consists of $f_{exc}$ (the excitation force due to the water wave) and $f_c$ (force applied by the control and PTO systems). The equation of motion can be rewritten as

$$(M+\mu_\infty)\ddot{s}(t) + \int_{-\infty}^{t} k(t-\tau)\dot{s}(\tau)d\tau + k_w s(t) = f_{exc}(t) + f_c(t), \tag{2.17}$$

where the impulse-response $k(\cdot)$ could also be computed directly using time-domain simulations using boundary element method software like ACHIL3D [5]. In control design, for example, using MPC or optimal latching (see section 2.3 and Chapter 3), this integral equation would have to be solved at each time. For wave energy converters, it was first shown in [15] that this computation can be more efficiently calculated with an approximate state-space model for the convolution integral. Moreover, the state-space model is a very convenient method of time-domain analysis and the method of choice in the automatic control setting.

In approximating (2.17) by a finite order system in state space form, [15] considers using a linear subsystem of order $n_r$ to approximate the integral term. Taking the velocity $u(t) := \dot{s}(t)$ as the input of this subsystem and the integral approximation as the output $y_r(t)$, the subsystem is described as:

$$\dot{z}(t) = A_r z(t) + B_r u(t), z(0) = 0, \;\; y_r(t) = C_r x(t) \approx \int_{-\infty}^{t} k(t-\tau)\dot{s}(\tau)d\tau, \tag{2.18}$$

where $z(t) = [z_1(t)\, z_2(t)\, \ldots\, z_{n_r}(t)]^T$ and $A_r$ , $B_r$ and $C_r$ are assumed to have the observer

companion-form realization shown below. Note that $z_{n_r}(t) = y_r(t)$. The equation of motion becomes:

$$\begin{aligned}(M+\mu_\infty)\ddot{s}(t)+y_r(t)+k_w s(t) &= f_{exc}(t)+f_c(t), \\ \dot{z}(t) &= A_r z(t)+B_r\dot{s}(t), \\ y_r(t) &= C_r z(t),\end{aligned} \tag{2.19}$$

where

$$A_r = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 & -a_1 \\ 1 & 0 & 0 & \dots & 0 & -a_2 \\ 0 & 1 & 0 & \dots & 0 & -a_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & -a_{n_r-1} \\ 0 & 0 & 0 & \dots & 1 & -a_{n_r} \end{bmatrix}, \quad B_r = [b_1\ b_2\ \dots\ b_{n_r}]^T, \quad C_r = [0\,0\ \dots\ 0\,1]$$

Since $z(0) = 0$, then the $n_r^{\text{th}}$-order state space approximation of the radiation model has impulse response $k_{n_r}(t)$ is given by

$$k_{n_r}(t) = C_r e^{A_r t} B_r. \tag{2.20}$$

The $2n_r$ unknown parameters in (2.19) can then be estimated from $l$ values of $k(t)$ evaluated at discrete times $\{t_i : i = 1,2,\dots,l\}$ derived either via software simulations or from experiment; $l \gg n_r$. To estimate the matrices $A_r$, $B_r$ and $C_r$, the minimization problem solved is

$$\min_{A_r,B_r,C_r} \sum_{i=1}^{l} |k(t_i) - C_r e^{A_r t_i} B_r|^2. \tag{2.21}$$

This is a nonlinear least squares (NL-LS) problem or LS curve fitting of the impulse response and can be performed using the Matlab function `lsqnonlin`. Although the assumption in [15] is that the accuracy of the estimate would improve with increasing $n_r$, no qualification is given on the goodness of the accuracy as $k(t)$ is itself often known with limited accuracy. Moreover, due to its non-convex nature, the solution of this method is found to be very much dependent on the initial guess. An iterative procedure that uses solutions as an initial guess can be used.

The radiation subsystem identification can also be performed in the frequency domain [16, 17, 20]. In this method, the irrational[1] transfer function $K(j\omega)$ in (2.16) can be ap-

[1]The transfer function involving the radiation potential comes from the partial differential equations for water

proximated by a rational transfer function using a sampled data of $K(jw)$ over a finite frequency range of interest. That is, the Laplace transform of the impulse response function $k(\cdot)$ is approximated by a strictly proper rational transfer function of a given order using sampled frequency response data; this NL-LS frequency response data fitting is implemented in Matlab's `invfreqs` function.

Another state-space time-domain approximation of the equation of motion is based on realization theory and aims to find a minimal state-space realization approximation [16, 17]. It is normally a sampled impulse response (or input-output) data that we get from a simulation or experiment. This sampling enforces a discrete-time model on the radiation response data. From this discrete data, a linear discrete-time system approximation of the form below with impulse response $h_d(\cdot)$ is sought.

$$z_d(t_{k+1}) = A_d z_d(t_k) + B_d u(t_k), \quad y_d(t_k) = C_d z_d(t_k) + D_d u(t_k), \tag{2.22}$$

$$h_d(t_k) = C_d A_d^{k-1} B_d + D_d. \tag{2.23}$$

By directly computing the Hankel matrix from the impulse response, the singular value decomposition (SVD) based method of [23] serves to approximate the McMillan degree (i.e. order of the minimal linear system realization) and then the associated model for exact realization of the impulse response data. A continuous-time linear state space model is then computed using the Tustin transform. This system identification method is implemented in the Matlab function `imp2ss`; see [23, 24] for more details. The order of the resulting model depends on which Hankel singular values we deem insignificant and get rid of. For example, if not specified, by default `imp2ss` discards any singular value less than 0.01 times the maximum. The lower-order models come with an $H_\infty$ norm bound to the high-order model that exactly realizes of the impulse response data. Model reduction can also be applied to the high-order model to further reduce the order of the approximate model.

### 2.2.3 Modelling the wave excitation force

Let us assume that a plane wave propagates in the positive $x$ direction and that the WEC is at the point $x_B$. Having found a state space approximation for the integral in (2.17), the equation of motion can be represented as in Figure 2.3a, where the whole system $H$ has a state-space description with inputs $f_{exc}(t)$ and $f_c(t)$, and its states as output. Since $f_{exc}(\cdot)$ depends on the sea state near the WEC and cannot be directly measured, often it is modelled as a function of a measurable parameter like the wave height at or near the device [11, 15, 25]. By modelling the operator or system from the wave height $\eta(\cdot)$ to

waves and their interaction with the body in water and are irrational.

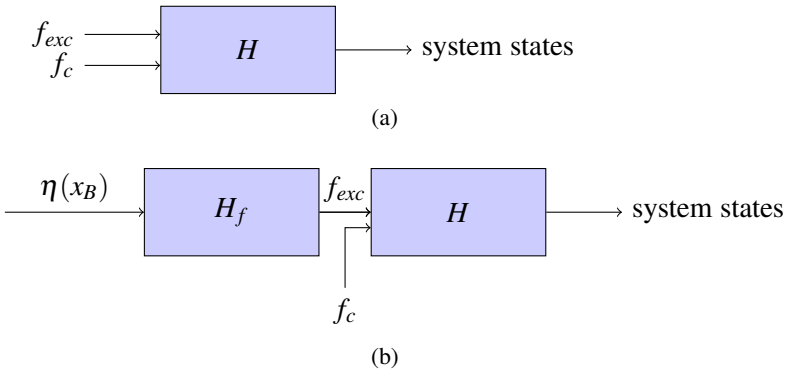



Figure 2.3: (a) The oscillating WEC as a system with inputs $f_{exc}, f_c$ and position states $s$ and/or its derivatives as output (b) The subsystem $H_f$ is augmented to $H$ so that wave height $\eta(\cdot)$ is the input and not $f_{exc}(\cdot)$.

$f_{exc}(\cdot)$ by $H_f$, we can augment this subsystem to $H$. We now have the wave elevation at the position of the WEC, $\eta(x_B,t)$ as the input and not $f_{exc}(t)$. Like the radiation force, the excitation force has an integral representation,

$$f_{exc}(t) = \int_{-\infty}^{\infty} k_{exc}(\tau)\eta(x_B,t-\tau)d\tau, \tag{2.24}$$

where $\eta(\cdot)$ is the wave elevation at the buoy and $k_{exc}(\cdot)$ the excitation force impulse response function [11], i.e. the impulse response of $H_f$ in Figure 2.3b.

Although wave propagation is a causal process, it is shown in [11, 25] that $k_{exc}(t)$ is non-causal for the dispersive gravity water waves. The lack of direct causal relationship between the force on a body in water and the wave height at a single point is shown in [25]. However, since the integral representation allows for system identification and state space modelling , it is the method of choice in the wave and hydrodynamics community. Here, the method of linear state-space model derivation discussed cannot be directly applied to the non-causal impulse response. In [25], the authors propose *causalizing* $k_{exc}(t)$ since it decays to zero very quickly for negative time $t$. It is shown that $k_{exc}(t)$ is negligible ( $\approx 0$) for $t < -t_c$ where $t_c > 0$ is a small number, then its causalized impulse response can be re-written as

$$k_{exc,c}(t) = \begin{cases} k_{exc}(t-t_c), & \text{for } t \geq 0 \\ 0, & \text{otherwise} \end{cases}. \tag{2.25}$$

A state space approximation of the causal impulse response $k_{exc,c}(\cdot)$ can then be computed the same way as for the radiation integral.

$$\dot{z}_f(t) = A_f z_f(t) + B_f \eta(x_B,t), X(0) = 0, \quad y_f = C_f z_f(t+t_c) \approx \int_{-\infty}^{\infty} h_f(\tau)\eta(x_B,t-\tau)d\tau \tag{2.26}$$

where $z_f$, $\eta(x_B,t)$ and $y_f(t)$ are the state, input and output of the subsystem $H_f$, respectively. We can see from (2.26) that future knowledge of the system states (and therefore the incident wave elevation at $x_B$ at a future time) is needed to calculate the output $f_{exc}$ at present. This, together with the need for optimal control, has motivated some research in short-term wave prediction [26]; see also Section 2.4.

## 2.3 Review of Existing Control Methods for WECs

Since point absorbers have a narrow bandwidth [11], for such a WEC without dynamic control the absorbed power would diminish significantly as the wave frequency varies. Early work focused on the use of mechanical impedance matching schemes to maximize the velocity and hence the absorbed power from sinusoidal (or regular) waves. By representing the PTO mechanism by a linear spring damper system and assuming linear wave-body interaction, simple frequency domain analysis was used to derive optimal amplitude and phase conditions on the velocity of the device with respect to a sinusoidal wave excitation force. Looking back at equations (2.17) and (2.14), by denoting the transfer function (or mechanical impedance) from the velocity to control force as $Z_c(jw)$, the equation of motion can be re-written as

$$\hat{u} = \frac{\hat{F}_e}{B(\omega) + j[\omega(M + M(\omega)) - \frac{k_w}{\omega}] - Z_c(j\omega)} \,. \tag{2.27}$$

We can assume the control system incorporates the PTO mechanism with no loss of generality. A linear control mechanism can be represented by a linear spring damper system; $Z_c(j\omega) = B_c(\omega) - jK_c(\omega)/\omega$. Simple impedance matching reveals that maximum average power is absorbed by the control PTO system when $B_c(\omega) = B(\omega)$ and $K_c(\omega) = \omega^2(M + M(\omega)) - k_w$. Because of the cancellation of the reactive (or imaginary) term in (2.27), the control satisfies the conditions that $\hat{u}$ and $\hat{F}_e$ have the same phase, and that the optimum amplitude condition is $\hat{u}(\omega) = \frac{\hat{F}_e(\omega)}{2B(\omega)}$. For these reasons, this frequency domain control is often called reactive control or optimal phase control [3, 11, 19, 27].

This method's theoretical optimal resonance condition, of having the velocity in phase with the sinusoidal force, has been shown to result in unrealistically large amplitudes and therefore unrealistically large two-way energy transfers between the body and PTO. Moreover, this amplitude condition does not allow for physical constraint handling [11, 28]. Another shortcoming is that the method is not applicable to systems with a nonlinear PTO.

From here on, we use the standard control engineering terminology *active control* to

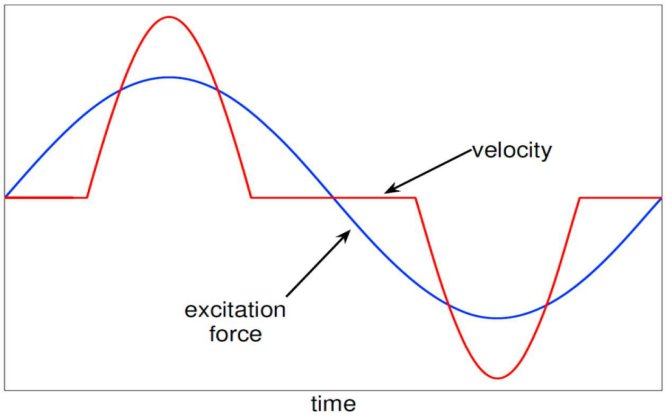



Figure 2.4: Latching control: the buoy is latched when its velocity is zero (red curve) and released at a favourable time so that it is in phase with the excitation force (blue curve).

refer to control mechanisms that are not passive. Consider a dynamical system with input $v(t)$ and output $y(t)$, where $y(t) = h(t, v(t))$, $t \in [0, \infty)$, $v, y \in \mathbb{R}^p$. The system is called passive if $v(t)^T y(t) \geq 0$, $\forall t \in [0, \infty)$ [29, Ch. 6]. The WEC controller can be seen as a dynamical system whose input is the WEC velocity and whose output is the control force it exerts on the WEC. A passive controller is a passive dynamical system. Control mechanisms that inject external energy like reactive control would then be active; compare this with passive methods like latching (see below) that do not use external energy to control the system.

In [30], the idea of passively meeting only the phase criteria of reactive control through a method called latching was considered. During its oscillation, the body is latched (i.e. prevented from moving) when its velocity vanishes and released at a favourable time. As illustrated in Figure 2.4, the body is latched when it comes to rest in its oscillation, and released at a 'favorable' time. In its implementation, this is equivalent to applying an infinite or very high damping $B_c(\omega)$. The work in [31] was the first to convert the problem of determining latching and releasing times into an optimal control problem and use Pontryagin's maximum principle (PMP) to solve it. For a buoy moving in heave only, assuming the wave excitation force is known some time into the future and neglecting radiation forces, a sequence of latching/unlatching commands were computed such that the energy absorbed by a linear damper is maximized. Latching is approximated numerically by a very large linear damper that can be switched on and off. It was shown that a large enough damping value in the numerical simulation approximates physical latching well. This approximation of latching is also used by all literature that followed [31].

Latching control is the most popular and studied choice in the wave energy community and has been shown in simulations to increase power yield by up to a factor of 8 in an

irregular sea compared to no control [31]. It has been applied to single DOF devices [5, 28, 32, 33] as well as WECs with multiple DOF [5, 19, 34, 35]. With deep water floating buoys in mind, optimal command latching for 3-body and 2-body systems with on-board motion controlled platforms is investigated in [34, 36]. In [36], time domain simulations are used to imply that power absorption for these devices is inferior to ones for conventional buoys with sea-bottom-fixed references.

A recent work in [19] also claims that, unlike the case of single body buoys moving relative to the sea bottom, latching does not give a substantial increase in absorbed energy for multibody floating WECs. In [19], latching does not constrain the motion of any of the moving bodies including the PTO with respect to the sea bottom; only the relative motion between the different bodies is constrained. A recent work in [35] uses a PMP formulation to compute optimal passive controllers. In [35], however, optimal latching is applied to a two-body device with nonlinear hydraulic circuitry. Optimal latching is also similarly formulated and applied to a single DOF device in [32] and later to a WEC with a four-DOF device called SEAREV [5]. However, just as is known for multiple DOF systems, the effectiveness of latching diminishes for an array of devices interacting with each other; as also noted by [37], the phase condition loses meaning and the optimal power absorption condition no more requires all bodies to have a velocity in phase with the excitation force [38, 39]. In addition, despite its high energy capture potential, the application of latching has been doubted because of the the excessive loads exerted on WECs in latching [19]. All these reasons have motivated some research in the application of advanced optimal control schemes to wave energy conversion.

Another passive method considers varying the damping coefficient of the PTO continuously in time; in practice, this is done in a discrete or pseudo-continuous way and results in a complex PTO with a lot of components. Motivated by the need to alleviate this problem, the work in [40] has shown via simulations that an on-off strategy with an optimal command gives more energy and therefore does at least as well as its continuous counterpart. Since this on-off strategy is practically implemented using a simple by-pass valve, this is called *declutching or unlatching* control. In Chapter 3, in addition to simulations, we show the on-off nature of the optimal control policy theoretically and within a more general PTO setting that also includes an active control element and with no assumptions on the regularity of the waves.

As in the classical reactive control case [11, 19, 28], injecting mechanical energy into the controlled WEC via the PTO machinery could help generate more energy on average. However, this two-way flow of energy needs to be efficient to justify its use; the more efficient the PTO or control mechanism is, the more applicable optimal active control will be. This has motivated some work in the design of high-efficiency hydraulic machines and novel electrical generator technology for WECs [41–43]. Recently, with

advances in the efficiency of power take-off systems for WECs, there is a growing interest in active control methods [7]. In Chapter 3, we consider model predictive control of a heaving point absorber using active and passive power take-off. We will review relevant literature and investigate model based optimal control for energy maximization within receding horizon control.

## 2.4 Short-Term Wave Prediction

It has been noted in the previous section that optimal control of WECs requires future knowledge of the wave energy excitation force. This is because optimization of energy absorption is performed over some horizon into the future. Moreover, the use of wave height in computing the excitation force requires knowing the wave height into the future (2.26).

One of the first works that attempted to use prediction for improving power yield from a heaving buoy is found in [30]. This work computes the excitation force on the WEC directly from a single pressure reading on the buoy and not from wave height data. By modelling the pressure dynamics as a damped second order system, a Kalman filter was used to make pressure predictions at future times. Although it was claimed in [30] that it offered some improvement in power yield by providing predictions for a sub-optimal unlatching control strategy, similar formulations have since been found too simplistic and not useful in a realistic sea wave environment [44].

A neural networks approach is taken in [45]; a nonlinear adaptive method is used to predict not excitation forces but 'events of interest' for latching control. Rather than predicting the future wave profile or excitation force, the work attempts to predict the ideal time for releasing a latched WEC, i.e. a next peak or trough in the wave excitation force, using only a past time-series of the wave height at a single position. The results in [45] show, using wave data from a Pierson-Moskowitz spectrum and a simple submerged buoy model, that this neural networks method performs well and could be useful. However, its utility is limited to a suboptimal latching controller and has also not been tried on real irregular sea data or other WECs. In fact, a recent study in [26] shows, via comparisons with a neural networks prediction method, that a relatively simple linear autoregressive (AR) model does much better. By modelling the wave height at a single point using AR systems of varying orders, predictions are made up to 10s of seconds into the future. Using a sampling frequency of 0.7 rad/s on real wave data from two coastal locations, [26] shows that AR models of order 12 and above can give reliable estimates of wave height up to two peak periods into the future. Based on these AR models, the prediction requirements for active control and the accuracy of the achiev-

able predictions are also assessed in [46]. The effects of the prediction errors on control performance are of great interest; these are reported as ongoing work by the authors of [44, 46].

In this chapter, we have given a review of wave energy conversion, existing modelling and control methods for wave energy devices as well as a brief review of wave prediction schemes. In the next chapter, we investigate model predictive control for a heaving buoy with active and passive control elements.

# Chapter 3

# Optimal Control and Optimization of a Wave Energy Converter

This chapter investigates model predictive control schemes applied to a point absorber wave energy converter. A variational formulation of the power maximization problem is adapted to solve the optimal control problem. It will be shown that the optimal active control method is of a bang-bang type for a power take-off mechanism that incorporates both linear dampers and active elements. We also consider a direct transcription of the optimal control problem into a general nonlinear program. A computationally inexpensive variation of the projected gradient optimization scheme is developed and shown to be feasible and computationally inexpensive compared to standard nonlinear program solvers. Results will be compared with an optimal command latching method to demonstrate the improvement in absorbed power. For all these methods, time domain simulations are presented under irregular sea conditions.

## 3.1 Introduction

Oscillating body wave energy converters (WECs) involve floating structures that oscillate because of water-body interactions with ocean waves. These then drive a power take-off (PTO) mechanism to generate useful power. As mechanical oscillators, these devices perform well only in sea conditions where the wave frequency is matched with their natural frequencies. Point absorbers are one such class of devices that are usually symmetric about the vertical axis and have small horizontal dimensions in comparison to the wavelength of the ocean waves. However, since they have a narrow bandwidth [11], without dynamic control the absorbed power would vary significantly as the sea wave characteristics vary at a given site. Therefore, optimal control of WECs has

been recognised as vital to the future of wave energy and has been studied since the early 1970s.

By representing the power take-off (PTO) by a linear spring-damper system and assuming linear wave-body interaction, simple frequency domain analysis was used to derive optimal amplitude and phase conditions on the velocity of the device with respect to a sinusoidal wave excitation force. Often called reactive control , this method's theoretical optimal resonance condition, of having the velocity in phase with the sinusoidal force, has been shown to result in unrealistically large amplitudes and large two-way energy transfers between the body and PTO. Moreover, this amplitude condition does not allow for physical constraint handling [11, 28]. In [30], the idea of passively meeting only the phase criteria through a method called latching is considered. The WEC is latched (i.e. prevented from moving) when its velocity vanishes and released at a favorable time.

Model based optimal latching was first suggested in [31]; an optimal sequence of latching/unlatching commands were computed to maximize extracted energy in a simplified model. As the most studied choice in the wave energy community, Optimal latching control has been applied to single DOF devices [5, 28, 32, 33] as well as WECs with multiple DOF [5, 34, 35]. However, just as is known for multiple-DOF systems, the effectiveness of latching diminishes for an array of devices interacting with each other; the phase condition loses meaning and the optimal power absorption condition no more requires all bodies to have a velocity in phase with the excitation force [38]. In recent years, this has motivated some research in the application of advanced active optimal control schemes to wave energy.

The works in [37, 44] consider the use of an active force within the framework of model predictive control (MPC) and so are of importance to our work. Both papers consider only an active element for the PTO and depend on the reformulation of an energy maximization problem as discrete-time model problems. As a time domain equivalent of reactive control, [44] considers an optimization problem where the difference between work done on the WEC by the excitation force and the energy radiated by it is maximized; this is not energy absorbed by the PTO since only excitation and radiation forces are considered. By making a state space approximation of the radiation force and using the velocity as the optimization variable, the discretized optimization problem is shown to be a semidefinite quadratic program in the discrete velocity values – a convex problem. Since the optimal velocity results in unrealistically large amplitudes even for small waves, the quadratic program is solved with box constraints on the heave amplitude of the buoy. By considering the same model but with a control input, the control force that gives the calculated optimal velocity profile of the uncontrolled model is sought. However, the usefulness of this controller is in doubt since it does not maximize the energy

absorbed by the PTO over prediction horizon; it merely maximizes the difference between the energy absorbed by the buoy (in the form of kinetic and potential energy) and the energy radiated away from the buoy. The optimisation problem that maximizes the energy absorbed by a PTO is finally defined and dismissed for giving less power and for resulting in "inaccurate solutions". Nonetheless, this should have been the more relevant optimization problem since the aim is to generate more useful energy via the PTO. The "inaccurate solutions" may also be a result of the large matrix inversions involved in the the algebraic formulae used in [44] to compute the control force from velocity values over the prediction horizon. These inverted matrices may become close to singular since they are shown only to be positive semidefinite. A final optimal control problem considered [44] is one of tracking the computed 'optimal velocity' trajectories. As also noted in [37], it seems unnecessary that only predicted velocities should be used as the optimization variable when the control force is the free variable. It is also not clear how robust the methods are, since they involve a number of big and possibly ill-conditioned matrix inversions. The relationship between the various formulations is also not clear. The work in [44] also examines the use of an extended Kalman filter based predictor and deem the prediction accuracy unsatisfactory for their MPC scheme and conclude better prediction methods need to be found.

Another model predictive control scheme is used by [37], where the objective is to maximize the energy absorbed by an ideal PTO force over the prediction horizon. The emphasis is on discretizing the system using a triangle-hold such that the objective function can be approximated with one where the optimization parameters become changes in the control input at each sampling time; the method employed allows the approximation of the objective function by a semidefinite quadratic program. This formulation allows easy integration of rate constraints on the control. It is also claimed, but not shown, that hard constraints on the buoy displacement, velocity and the control force can be reformulated as constraints on the new optimization parameters. The authors then use soft constraints on the control force and its rate so as to avoid "feasibility issues". However, it has to be mentioned that these quadratic penalties on the control would have to be very conservative to guarantee hard constraints are not violated. Moreover, as will be shown in this chapter, the optimal control is of bang-bang type when no displacement or velocity constraints are imposed. This would exploit the practical advantage that bang-bang controllers can be implemented with simple on-off machinery; a triangle-hold implementation would not make use of this advantage.

A recent work on the use of active control for a WEC [47] considers the synthesis of optimal causal controllers using the statistical characterisation of ocean waves. There, the emphasis is on the causality of the controller — the need to know future wave forces is alleviated. The WEC considered is a floating buoy moving in all degrees of freedom

and tethered to three rotary generators via a spring pulley system. The wave power spectral density is approximated by a linear time-invariant model driven by a white noise process and this is augmented to the buoy dynamic model. The objective to maximise the average extracted electrical energy results in a non-standard LQG optimal control problem. This formulation, however, is possible only because the PTO considered is an electrically driven generator/motor with a spring pulley system and would not apply to systems with hydraulic PTO components. The work also shows the controller to be non-robust to changes in the wave statistics and would require the characterisation of the wave and a subsequent periodic synthesis of an LQG controller within an adaptive control scheme. In addition, the method does not allow constraint handling.

In this work, we consider a general optimal active control problem for a heaving point absorber. It is general in the sense that it considers a PTO with a controlled damping element in addition to the active control force considered by [37, 44]. As such, it reduces to an optimal declutching type problem if we remove the active control command, and it reduces to the absorbed energy maximization problems considered in [37, 44] if the damping element is removed. Moreover, this formulation can be generalised in a straightforward manner to devices moving in more degrees of freedom and with various control elements. Actuation and physical constraints can also easily be incorporated in this setting. As in the previously discussed literature, we assume that the excitation force is known in the prediction horizon.

In Section 3.2, we will discuss the dynamics of a heaving buoy and touch upon how a state space model is derived for control. Section 3.3 presents a variational formulation of the optimal control problem inherent in a model predictive control scheme and discusses methods to solve it. Having shown that the optimal control for the state unconstrained problem is of a bang-bang type, we will present a numerical scheme for computing the control commands. We will formulate and use a globally convergent and computationally cheap gradient projection scheme and take advantage of the strictly bang-bang nature of the solution. We also employ a state-of the art interior-point optimization software to solve the resulting nonlinear program for comparison and validation. Finally, in Section 3.4 an example device is used to demonstrate the computational gains from using the projected gradient method. Control feasibility and the improvement that optimal active control delivers over optimal latching control is also presented. The method is applied under various irregular wave conditions.

## 3.2 System Dynamics

In this chapter, we consider a semi-submerged cylindrical point absorber constrained to move in heave only; see Figure 3.1. A rigid body interacting with an inviscid, incompressible and irrotational fluid flow is assumed. Considering the sea bottom as an inertial reference, the vertical displacement of the buoy from the equilibrium (in the absence of waves) is represented by $\zeta(t)$. Then, the buoy displacement at time $t$ is given as

$$M\ddot{\zeta}(t) = f_c(t) + f_h(t) + f_r(t) + f_{exc}(t), \tag{3.1}$$

where $M$ is the mass of the body and $f_c$ represents the vertical control force exerted on the buoy. The net hydrostatic restoring force due to buoyancy and gravity is given by $f_h$ and is proportional to the displacement

$$f_h(t) = -C_h\zeta(t) \tag{3.2}$$

where the hydrostatic stiffness $C_h := \rho g S$, with $\rho$ being the density of water, $g$ gravitational acceleration and $S$ the cross-sectional area of the buoy. The heave excitation force $f_{exc}$ is the force exerted on the stationary body at equilibrium due to the interaction with the oncoming waves. The radiation forces $f_r$ describe the forces due to the movement of the body itself in the absence of incident waves; changes in the momentum of the surrounding fluid and the resulting radiated waves give rise to net forces on the body. Assuming a linear water-body interaction and using velocity potential theory these forces can be linearly related to the displacement, velocity and acceleration of the buoy in the frequency domain; see [11] for the derivation of frequency domain transfer functions relating the velocities with radiation and excitation forces for some floating geometries in water.

As reviewed in Section 2.2.2, a time-domain approach models the radiation force as

$$f_r(t) = -\mu_\infty\ddot{\zeta}(t) - \int_{-\infty}^{t} k_r(t-\tau)\dot{\zeta}(\tau)d\tau, \tag{3.3}$$

also referred to as the Cummins equation [22]. The so-called infinite-frequency added mass $\mu_\infty$ represents an instantaneous force response of the fluid after an impulsive movement of the buoy. The convolution integral represents forces due to the transient fluid motion or radiated waves caused by the motion of the buoy. The impulse-response of the radiation force $k_r(\cdot)$ can be computed using time-domain simulations via software like WAMIT and ACHIL3D [5]. The equation of motion (3.1) can now be re-written as:

$$(M+\mu_\infty)\ddot{\zeta}(t) + \int_{-\infty}^{t} k_r(t-\tau)\dot{\zeta}(\tau)d\tau + C_h\zeta(t) = f_{ext}(t) + f_c(t). \tag{3.4}$$

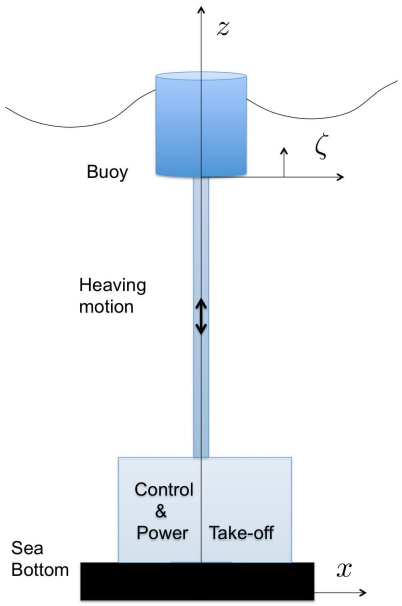



Figure 3.1: A schematic of a heaving buoy point absorber WEC.

In a control algorithm, (3.4) would have to be solved at each sample instant. However, this computation is more efficiently calculated with an approximate state-space model for the convolution integral [15]. Considering the velocity $\dot{\zeta}(t)$ as the input of a linear time-invariant continuous-time system of order $m$ and the integral approximation $y_r(t)$ as the output, we have:

$$\begin{aligned} \dot{z}_r(t) &= A_r z(t) + B_r \dot{\zeta}(t), \;\; z_r(0) = 0, \\ y_r(t) &= C_r z_r(t) \approx \textstyle\int_{-\infty}^{t} k_r(t-\tau)\dot{\zeta}(\tau) d\tau, \end{aligned} \tag{3.5}$$

where the state $z_r(t) \in \mathbb{R}^{n_r}$, $A_r \in \mathbb{R}^{n_r \times n_r}$, $B_r \in \mathbb{R}^{n_r \times 1}$ and $C_r \in \mathbb{R}^{1 \times n_r}$. As in [37], we call this the *radiation subsystem*.

With a radiation subsystem of order $n_r = n-2$ identified in (3.5), the WEC system dynamics can be re-written in state space form as:

$$\begin{aligned} \dot{x}_1(t) &= x_2(t), \\ \dot{x}_2(t) &= \tfrac{1}{M+\mu_\infty}[f_{exc}(t) + f_c(t) - C_r x_{3:n}(t) - C_h x_1(t)], \\ \dot{x}_{3:n}(t) &= A_r x_{3:n}(t) + B_r x_2(t), \end{aligned} \tag{3.6}$$

where the notation $x_{a:b}$ is to be interpreted as 'rows $a$ to $b$ of the column vector $x$' and the new state $x := [x_1 \; x_2 \; \ldots \; x_n]^T = [\zeta \; \dot{\zeta} \; z_r^T]^T \in \mathbb{R}^n$ with the appropriate initial conditions. See [17] and references therein for methods of system identification – we use the time-domain method implemented in the Matlab function `imp2ss` and discussed in [17].

Like the radiation force, the excitation force has an integral representation, $f_{exc}(t) = \int_{-\infty}^{\infty} k_{exc}(t-\tau)\eta(\tau)d\tau$, where $\eta(\cdot)$ is the wave elevation at the buoy and $k_{exc}(\cdot)$ the excitation force impulse response function. The non-causality of $k_{exc}(\cdot)$ and methods to "causalize" it are discussed in [15]. This problem can be rectified either by predicting the wave elevation at the buoy sometime into the future or by measuring the wave elevation some distance ahead of the buoy in the direction of the wave propagation. A state space approximation is computed the same way as for the radiation integral. In the following, we assume all states are known. The design of observers for the radiation forces from position, velocity and other sensor information is the subject of the next chapter.

## 3.3 Optimal Control Problem

The aim here is to examine the optimal control problem to be used within a model predictive control (MPC) scheme implemented in a receding horizon fashion. The underlying basis of MPC is an iterative, finite-time optimization of the plant model [48]. At any sampling instant, the measured state values are used as initial conditions to calculate an optimal input function or sequence and the associated future state trajectory. Therefore, at the root of MPC are an optimal control algorithm to find the optimal input sequence, and an ordinary differential equation (ODE) solver to calculate the state trajectory from the dynamics of the system. Here, we investigate the optimal control problem for maximizing energy extracted from a generic WEC. We assume that the excitation force is known over the prediction horizon into the future. In reality, this is not true and an estimate of the excitation force would be used. The topic of 'how good a given prediction scheme is or has to be for MPC' is beyond the scope of this thesis.

### 3.3.1 The optimal control problem

Let us consider the WEC of (3.6) again. Here we assume that power is taken off through a damping force proportional to the velocity (with the constant damping coefficient $B_{pto}$ being controlled proportionally through the input command $u_2(t) \in [0,1]$) and a bounded active forcing element; i.e. $f_c(t) = -B_{pto}u_2(t)x_2(t) + u_1(t)G$, where $u_1 \in [-1,1]$ and $G > 0$ is a (large) constant with a unit of force (N). Naturally, $G$ would be determined by the physical constraints of the actuation mechanism. The equation of motion (3.6)

can then be re-written as:

$$\begin{aligned}\dot{x}_1(t) &= x_2,\\ \dot{x}_2(t) &= \tfrac{1}{M+\mu_\infty}[f_{exc}(t)+Gu_1(t)-B_{pto}u_2(t)x_2(t)-C_r x_{3:n}(t)-Cx_1(t)],\\ \dot{x}_{3:n}(t) &= A_r x_{3:n}+B_r x_2(t),\end{aligned} \tag{3.7}$$

where $C := (C_h + k)$ and $k$ is the stiffness of an external spring system attached to the buoy, $u_1(t) \in [-1,1]$ and $u_2(t) \in [0,1]$. This dynamics is that of a *bilinear system*, i.e. it is linear in the input and linear in the state, but not jointly linear in both. Notice also that the damper is always opposing motion (when on) while the active control force can assist or oppose motion (i.e. can do positive work on the device or take away energy from the device at different times, while the damper always takes off energy when engaged). From here on, where convenient, we use the augmented input vector $u(t) := [u_1(t)\ u_2(t)]^T$ and the set $\mathbb{U} := \{u(\cdot) : u_1(t) \in [-1,1] \text{ and } u_2(t) \in [0,1],\ \forall t \in [t_0,\ t_f]\}$. The objective, at time $t = t_0$, is to maximize the energy extracted $\left[E := \int_{t_0}^{t_f}[B_{pto}u_2(t)x_2^2(t) - Gu_1(t)x_2(t)]dt\right]$ over a future time interval $[t_0,\ t_f]$; that is, we solve the optimal control problem:

$$\begin{aligned}OCP: \max_{u(\cdot)\in\mathbb{U}} \int_{t_0}^{t_f} &\{B_{pto}u_2(t)x_2^2(t) - Gu_1(t)x_2(t)\}dt.\\ &\text{subject to (3.7) and } x(t_0) = \hat{x} \text{ given.}\end{aligned} \tag{3.8}$$

Alternatively, we can define an objective function $\hat{E} := -E$ and solve a minimization problem. The dynamic constraint of (3.7) can be added to the minimization problem using a Lagrange multiplier $\lambda \in \mathbb{R}^n$ as

$$J := -E + \int_{t_0}^{t_f} \lambda^T(t)[f(x(t),u(t),t) - \dot{x}(t)]dt \tag{3.9}$$

where $f(\cdot)$ is a vector representation of the right hand side of (3.7) and $x(t)$ the state. The Hamiltonian associated with the minimization of $J$ then becomes [49, Sec. 2.3]:

$$\begin{aligned}H(x,u,\lambda,t) := &-B_{pto}u_2x_2^2 + Gu_1x_2 + \lambda_1 x_2 + \frac{\lambda_2}{M+\mu_\infty}\{f_{exc}(t) + Gu_1 - B_{pto}u_2x_2 -\\ &C_r x_{3:n} - Cx_1\} + \lambda_{3:n}^T(A_r x_{3:n} + B_r x_2).\end{aligned} \tag{3.10}$$

The $t$ in the time dependent variables $u$, $x$ and $\lambda$ is dropped for notational convenience.

Pontryagin's maximum principle considers the above formulation and derives necessary (and sufficient) conditions for optimality based on the idea that small variations of an optimal control $u$ should result in a non-decreasing value of the objective function of

the minimization problem. We consider an optimal input $u(\cdot) \in \mathbb{U}$ and an arbitrarily small admissible perturbation $\delta u(\cdot)$, i.e. $u(t) + \delta u(t) \in \mathbb{U}, \forall t \in [t_0, t_f]$ and $||\delta u(t)||_{L^1} < \varepsilon$, where for $v : [0,\infty) \to \mathbb{R}^l$, $||v||_{L^1} := \int_0^\infty \sum_{i=1}^{l} |v_i(t)| dt$ [50, Sec. 3.4]. The cost function can then be shown to satisfy:

$$J(u+\delta u) - J(u) = \int_{t_0}^{t_f} \{H(x, u+\delta u, \lambda; t) - H(x, u, \lambda; t)\} dt + O(\varepsilon), \tag{3.11}$$

where $\varepsilon$ is a small number and the vector of Lagrange multipliers (or costates) $\lambda$ satisfy the set of adjoint differential equations

$$\dot{\lambda}(t) = -\frac{\partial H^T}{\partial x}(x(t), u(t), \lambda(t), t), \tag{3.12}$$

and the terminal condition is

$$\lambda(t_f) = 0, \tag{3.13}$$

because the terminal cost is zero. Detailed derivations of this form of the PMP are available in [50, Ch. 3] and [51, Sec. 3.4]. We give the equations for the adjoint system dynamics of our optimal control problem in Appendix A.

Generally, through the weak form of the PMP, a candidate (locally) optimal control law $u(\cdot)^*$ can be derived from the first order necessary condition $H_u := \partial H(\cdot)/\partial u = 0$ and sufficient conditions are verified using $\partial^2 H(\cdot)/\partial u^2$ or by substituting $u(\cdot)^*$ into the objective function. However, since both the performance measure in (3.8) and the dynamics (3.7) are linear in the control input, $u$ does not appear in $H_u$. Therefore, it does not give us a candidate optimal control; we resort to the strong form of the Pontryagin's maximum principle [50].

A first order necessary condition for optimality, i.e. for $J(u+\delta u) - J(u)$ in (3.11) to be non-negative, is then [50, Thm 3.4.2]

$$H(x(t), u(t)^*, \lambda(t); t) \leq H(x(t), u(t), \lambda(t); t), \ \forall u(t) \in \mathbb{U}, \ \forall t \in [t_0, t_f], \tag{3.14}$$

where $H(\cdot)$ and $\lambda(\cdot)$ are as defined in (3.10), and (3.12)–(3.13), respectively. Simply put, PMP states that the optimal control, and its corresponding state and co-state trajectories, must minimise the Hamiltonian for all time $t \in [t_0, t_f]$ and for all "neighbouring" admissible inputs.

In problems where the control is bounded, i.e. $\mathbb{U} := \{u(\cdot) : u(t) \in [u_{min}, u_{max}], \forall t \in [t_0, t_f]\}$, (3.14) allows us to show necessary conditions for optimality. An additional special property of the problem we are considering is that the Hamiltonian is affine in

the control, i.e. has the form:

$$H(x(t),u(t),\lambda(t);t) = l(x(t),\lambda(t),t) + \sigma(x(t),\lambda(t),t)^T\, u(t)\ \ \forall t \in [t_0,t_f], \tag{3.15}$$

where $l(x(t),\lambda(t),t) \in \mathbb{R}$ and $\sigma(x(t),\lambda(t),t) \in \mathbb{R}^m$, $\forall x,\lambda,t$. This is a direct result of both the system dynamics and the cost function being linear in the input. The necessary condition of (3.14) then reduces to:

$$\sigma(x(t),\lambda(t),t)^T\, u^*(t) \leq \sigma(x(t),\lambda(t),t)^T\, u(t)\ \ \forall u(t) \in \mathbb{U},\ \forall t \in [t_0,t_f]. \tag{3.16}$$

This further simplifies to conditions on the components of the optimal input, namely:

$$u_i^*(t) = \begin{cases} u_{min,i} & \text{if } \sigma_i(x(t),\lambda(t),t) > 0, \\ u_{max,i} & \text{if } \sigma_i(x(t),\lambda(t),t) < 0, \\ \text{undetermined} & \text{if } \sigma_i(x(t),\lambda(t),t) = 0, \end{cases} \tag{3.17}$$
$$\text{for } i = 1,\ldots,m,\ \forall t \in [t_0,t_f].$$

It is clear that $\sigma(\cdot) = H_u(\cdot)$. The components $\sigma_i(\cdot)$ are called switching curves; the optimal input components switch from one boundary to the other at the zero crossings of the corresponding function. We say a singular arc occurs if any of the switching functions $\sigma_i(\cdot), i = 1,\ldots m$, vanishes identically on an interval of nonzero measure in $[t_0,t_f]$. In such intervals, (3.17) does not determine the optimal input. In the next section, we show that the optimal control problem of (3.8) with only input constraints does not contain singular arcs and is of bang-bang type, i.e. takes only its boundary values. The algorithm is presented in Section 3.3.3.

### 3.3.2 Analysis of singular arcs

Here, we show that the optimal control problem (3.8) with only input constraints has no singular arcs. Simulations also show that under small (or normal) wave conditions, state constraints are actually never active. Therefore, the absence of singular arcs can be expected for the problem with state constraints under small wave conditions.

Let us consider the switching functions for (3.10). Along a singular arc, one or more of the switching functions vanish and so the linear necessary conditions of the PMP (3.16) are not adequate to determine a unique optimal control candidate; infinitely many admissible control trajectories, $u(\cdot) \in \mathbb{U}$, trivially meet the conditions. In order to determine a control law along the arc, a high order maximum principle (HMP) has to be considered. (The methods are called high order because they consider more complicated variations in the input to find necessary conditions. For example, the second order

maximum principle is derived by considering second order (quadratic) variations of the objective function with respect to the $L^1$ norm of the control variation $\delta^2 J/\delta u^2$, where $||\delta u||_{L^1} \leq \varepsilon$ and the first order variations give rise to the PMP necessary conditions which are identically zero over the arc) [49, Ch. 6], [52].

For a general problem with a Hamiltonian of the form (3.15) and a scalar input, suppose that the switching function $\sigma(x,\lambda;t) := H_u$ is identically zero over a finite interval $(t_1,\, t_2) \subset [t_0,\, t_f]$. Then all the time derivatives $\frac{d^q}{dt^q}H_u, q = 1,2,\ldots,\infty$, must also vanish over the same interval. By successive differentiation of the switching function, one may find the smallest integer $q$ such that:

$$\begin{array}{ll} \frac{d^i}{dt^i}H_u & = 0,\ \forall t \in [t_1,\, t_2],\ i = 0,\ldots,q \text{ and} \\ \frac{\partial}{\partial u}(\frac{d^q}{dt^q}H_u) & \neq 0, \text{ for some } t \in (t_1,\, t_2). \end{array} \tag{3.18}$$

If such a finite $q$ exists, it must be even [49, Sec. 8.4][52]. The variable $p$, with $q = 2p$ is called the order of singularity. Moreover, the candidate singular optimal control over $[t_1,\, t_2]$ and the corresponding $2n-q$ dimensional singular manifold of the $(x,\lambda)$-space are computed by substituting the state and adjoint dynamics (3.7) and (3.12), respectively, into (3.18). In addition, an admissible optimal control along a singular arc of order $p$ must satisfy (what is called) the generalized Legendre-Clebsch necessary condition [52], i.e.

$$(-1)^p \frac{d^{2p}}{dt^{2p}} H_u \geq 0,\ \forall t \in [t_1,\, t_2]. \tag{3.19}$$

The opposite inequality is valid for a maximization problem.

For problems with multiple inputs, (3.18) and (3.19) apply to each control input with some additional matrix necessary conditions for optimality; see [52, Thm. 6.2 and Cor. 6.3]. We use these results to prove that there cannot be singular arcs in the problem considered here.

**Proposition 3.1.** *For the optimal control problem* (3.8)*, every feasible solution* $(x,\lambda,u)$ *is regular, i.e. does not have singular arcs. That is, the optimal control contains only bang-bang arcs over* $[t_0,\, t_f]$*.*

*Proof.* Assume the contrary, i.e. that the optimal control contains singular subarcs. This implies that one or more of the switching functions vanish over a nonempty open interval of some $[t_1,\, t_2] \subset [t_0,\, t_f]$. We look at each such possibility. From (3.10), the switching

function vector is given as:

$$\begin{aligned}\sigma(x,\lambda,t) &:= [\tfrac{\partial H}{\partial u_1} \;\; \tfrac{\partial H}{\partial u_2}]^T \\ &= [G(x_2+\tfrac{\lambda_2}{M+\mu_\infty}) \;\; -B_{pto}x_2(x_2+\tfrac{\lambda_2}{M+\mu_\infty})]^T, \\ &\forall t \in [t_0,\, t_f].\end{aligned} \tag{3.20}$$

The only two cases under which an optimal singular arc may be possible are:

i) $\sigma_1(x,\lambda,t)=0$ over $(t_1,\, t_2)$ (this also implies $\sigma_2(\cdot)=0$ over the same interval).

Assuming that the wave excitation force $f_{exc}(\cdot)$ is sufficiently smooth, successive differentiations of $\sigma_1(\cdot)$ and substituting the state and adjoint dynamics (3.7) and (3.12), respectively, reveals that the minimum integer $q$ that satisfies (3.18) is $q=3$. This contradicts the fact such $q$ should be even and therefore case (i) is not possible.

ii) $\sigma_2(x,\lambda,t)=0$ over $(t_1,\, t_2)$ and $\sigma_1(\cdot)\neq 0$ except possibly at a finite number of points over the same interval, i.e. $x_2=0$ with $\lambda_2\neq 0$ except possibly at a finite number of points over $(t_1,\, t_2)$.

Taking a single time derivative of $\sigma_2(\cdot)$ reveals $\dot{x}_2=0\;\forall t\in(t_1,\, t_2)$. This implies $u_1(t)=\frac{C_r x_{3:n}(t)+Cx_1(t)-f_{exc}(t)}{G}$ over the same interval. However, since $\sigma_1(\cdot)\neq 0$ except possibly at a finite number of points over $(t_1,\, t_2)$, $u_1$ should only take its boundary values in this interval. Therefore, case (ii) is not possible and this proves our result. □

### 3.3.3 Optimal control algorithm: a gradient projection scheme

Although we had started Section 3.3.1 with the assumption that our control inputs can take a continuum of values in a bounded set, it has been shown that the optimal control inputs take values only on the boundary of the feasible set. With this in mind and assuming the digital control implementation will be piecewise constant, we solve an approximate finite-dimensional optimization problem. The new problem is approximate in the sense that we are seeking an input in a piecewise continuous and bounded subset of the original infinite dimensional feasible set $\mathbb{U}$. We outline the online control synthesis algorithm below.

Let the optimization interval $[t_0,t_f]$ be divided into $N\in\mathbb{Z}^+$ equal subintervals and $h:=\frac{t_f-t_0}{N}$ be the sampling period with sampling instants $t_j:=t_0+jh$ and piecewise constant control inputs, i.e. $u(t)=u(t_j)\in\mathbb{R}^m,\;\forall t\in[t_j,t_{j+1}),\;\forall j\in\{0,1,\ldots,N-1\}$. Let $u_{i,j}=u_i(t_j)$, then our aim is to find an optimal control sequence $\bar{u}=\left(u_{1:m,j}\right)\in\mathbb{V}\subset\mathbb{R}^{m\times N}$, where $\mathbb{V}=\{\left(u_{1:m,j}\right):u_{i,j}\in[u_{min,i},u_{max,i}]\subset\mathbb{R},\; i=1,...,m,\; j=0,1,\ldots,N-1\}$, $u_{min,i}$ and $u_{max,i}$ are the lower and upper bounds, respectively, on the $i^{th}$ control input, $m$ is

the number of control inputs ($m = 2$ in OCP) and $u_{1:m,j}$ is to be interpreted as the input vector at time $t_j$ .

Although more advanced schemes could be used (see Section 3.4.1), the method we adopt here is to iteratively improve our input sequence by minimizing the objective function (3.9) using a variation of the steepest descent method. This method has the nice property that it is globally convergent to stationary points (see [53]) under mild assumptions. The main advantage of our particular scheme is its smaller computational cost; it requires only a single state and costate evaluation at each iteration and converges within 10's of iterations (see Algorithm 1 and the simulations in Section 3.4).

With a feasible initial choice $\bar{u}^0$, traditional gradient methods, like the steepest descent method, seek iterates

$$\hat{u}^{k+1} := \bar{u}^k - s^k \nabla J(\bar{u}^k), \tag{3.21}$$

where $s^k$ is the step size at iteration $k$. The gradient, $\nabla J(\bar{u}^k) := \frac{dJ}{d\bar{u}}(\bar{u}^k)$, is an $m \times N$ matrix whose components in (3.22) below measure the variation of the cost function with respect to each input and within each sampling interval. From (3.11) we get:

$$\begin{aligned} \nabla J(\bar{u}^k)_{i,j} &= \int_{t_j}^{t_{j+1}} \frac{\partial H^T}{\partial u_i}(x(t), u(t), \lambda(t), t)dt|_{\bar{u}^k}, \\ &= \int_{t_j}^{t_{j+1}} \sigma_i(x(t), \lambda(t); t)dt|_{\bar{u}^k}, \end{aligned} \tag{3.22}$$

where $i = 1, \ldots, m$ refers to the input component and $j = 0, \ldots, N-1$ identifies the sampling interval in $[t_0, t_f]$.

The tenet of a projected gradient method (PGM) is that it keeps the iterates feasible [53]. In every iteration, a step in the direction of the anti-gradient is taken and the result in (3.21) projected onto the feasible set $\mathbb{V}$,

$$\begin{aligned} \bar{u}^{k+1} &:= P_{\mathbb{V}}(\hat{u}^{k+1}), \\ P_{\mathbb{V}}(\hat{u}) &:= \arg\min_{\bar{v} \in \mathbb{V}} ||\bar{v} - \hat{u}||. \end{aligned} \tag{3.23}$$

Although this projection operation can be computationally demanding with a substantial overhead for a general feasible set, it is easily computed for simple convex sets like the polyhedron (or box) set $\mathbb{V}$ considered here. The projection is a simple element-wise bounding

$$\bar{u}_{i,j}^{k+1} = \begin{cases} u_{max,i} & \text{if } \hat{u}_{i,j}^{k+1} > u_{max,i}, \\ u_{min,i} & \text{if } \hat{u}_{i,j}^{k+1} < u_{min,i}, \\ \hat{u}_{i,j}^{k+1} & \text{otherwise }, \end{cases} \tag{3.24}$$

$$\text{for } i = 1, \ldots, m,\ j = 0, \ldots, N-1$$

and therefore its computational demand is marginal.

To make use of existing PGM results, we make the following technical assumptions:

**Assumptions**

1. The objective function $J(\cdot;x_0)$ is continuously differentiable and bounded from below on the closed convex set $\mathbb{V}$.
2. The gradient $\nabla J(\cdot;x_0)$ is Lipschitz, i.e. $\exists\, L \geq 0$ such that $||\nabla J(\bar{u}) - \nabla J(\bar{v})|| \leq L||\bar{u} - \bar{v}||$, $\forall \bar{u}, \bar{v} \in \mathbb{V}$, where $||\cdot||$ can be any p-norm. We assume, of course, that $x_0$ is bounded and that the state and co-state trajectories stay bounded.

From the definitions of the dynamics and the objective function, it is trivial that the cost $J(\cdot)$ is continuous in the input and bounded from below. Using the standard result that the state and adjoint variables are continuous even under piece-wise continuous (or bang-bang) inputs [49], from (3.22), the gradient of the cost is continuous with respect to input variations in $\mathbb{V}$. The Lipschitz assumption results from the continuity and boundedness of $\nabla J(\cdot)$ over the compact set $\mathbb{V}$. This could also be inferred from the physical principle that we could not extract infinitely more energy from the device using a finite change in the control forces.

With these assumptions, one can show that the projected gradient method converges to a local minimum for various step-size rules [53, Sec. 2.3]. As in the steepest descent method, the limitation of this method is that it generally has poor convergence. Nonetheless, it is shown in [53] that fast (superlinear) convergence can be achieved using a combination of Armijo-type line search schemes and Newton and quasi-Newton methods. These, however, are complex algorithms performing line searches and associated function and Hessian evaluations at each iteration and have overheads comparable to complex nonlinear program (NLP) solvers. For convex problems, fast gradient methods that use a constant step-length with slight modifications can still achieve the best of either linear convergence rate (the initial residual of the cost is decreased exponentially) or a quadratic convergence rate can be achieved [54, 55]. Since the problem considered here is not convex and the aim is to avoid the overhead incurred in performing line searches and associated Hessian and function evaluations at each step, we opt for a constant step-length scheme, similar to the fast gradient methods of [54, 55].

**Theorem 3.1.** *With the above assumptions satisfied on the closed convex set $\mathbb{V}$, taking a constant step-size $s^k = s$, where $0 < s \leq \frac{2(1-\sigma)}{L}$, $0 < \sigma < 1$, the algorithm in* (3.23) *is globally convergent to a local minimum (or stationary point) and the following are*

*valid:*

$$J(\bar{u}^k) - J(\bar{u}^{k+1}) \geq \frac{\sigma}{s}||\bar{u}^k - \bar{u}^{k+1}||^2, \tag{3.25a}$$

$$J(\bar{u}^k) - J(\bar{u}^{k+1}) \geq \min\{1,s\}\frac{\sigma}{s}||\bar{u}^k - P_{\mathbb{V}}\{\bar{u}^k - s\frac{dJ^T}{d\bar{u}}(\bar{u}^k)\}||^2. \tag{3.25b}$$

*Proof.* The global convergence of the algorithm and the conditions in (3.25a)–(3.25b) are a standard result and relegated to [53, Sec. 2.3.2] or [56, Thm 4.1]. □

We have proposed a method that is globally convergent to a local minimum. Although only a sublinear global convergence rate can be guaranteed, the algorithm converges much quicker (at worst, within a few tens of iterations) than what is predicted by the convergence analysis. For a 'fine enough' sampling, the switching curves do not seem to have many zero crossings between sampling instants and so the gradient in (3.22) is element-wise bounded away from zero. As a result, the rate of reduction in the cost at each iteration in (3.25b) does not diminish until we get very close to a minimum, in which case all the inputs have converged for all practical purposes; each input can be rounded to its nearest boundary value. It should also be noted that the Lipschitz constant $L$ is usually unavailable, and so we cannot determine the range of feasible step sizes $s^k$ a priori. We will, however, demonstrate that choosing an appropriate $s^k$ via offline simulations under various conditions would suffice. A pseudo-code of the algorithm is given below in Algorithm 1.

**Algorithm 1** Projected gradient algorithm

Fix some algorithm termination parameters $\varepsilon_u$, and $max_{iter}$; $\varepsilon_u$ is sufficiently small to indicate convergence of the input and the cost functions, respectively. We carry out a maximum of $max_{iter}$ iterations.

**Step 0** Choose/guess some initial control sequence $\bar{u}^0$; $\bar{u}^0 = 0$, for example. Set the iteration number to $k = 0$.

**Step 1** With the control input $\bar{u}^k$, evaluate the state $x$ over $[t_0, t_f]$ by integrating the system dynamics (3.7) with the initial condition $x(t_0) = x_0$ and store it as $\bar{x}^k$.

**Step 2** Compute the costates $\lambda$ by backward integration of (3.12) over $[t_f, t_0]$ using $\bar{u}^k$, the $\bar{x}^k$ values already computed in **Step 1** and the final values $\lambda(t_f) = 0$.

**Step 3** Compute $\bar{u}^{k+1}$ using (3.21)–(3.23).

**Step 4** If $||\bar{u}^{k+1} - \bar{u}^k|| \leq \varepsilon_u$ OR $i + 1 > max_{iter}$, then stop, else increment $k$ by 1 and return to **Step 1**.

### 3.3.4 Sufficient conditions of optimality and state inequality constraints

In optimal control problems where the cost function is a convex function of the state and input, showing sufficiency is straightforward; the positivity of the second variation of the Hamiltonian with respect to the input verifies it. Although it would be interesting to investigate whether the bang-bang type controllers are locally optimal for the optimal control problems defined in this chapter, this is often a very hard task.

A globally optimal approach for solving general non-convex optimal control problems is dynamic programming. In problems where the system dynamics are defined by finite-dimensional ODEs, this reduces to solving the Hamilton-Jacobi-Bellman (HJB) partial differential equation [57]. In principle, the HBJ PDEs can be solved by discretizing the state-space over a bounded domain of interest in each dimension using $N$ grid-points (or $N$ elements). However, the complexity of the problem grows as $N^n$ if the state space lives in $\mathbb{R}^n$. This is called the curse of dimensionality and limits the possibility of solving the HJB equation only to convex problems like the standard LQR problem or, in the nonconvex case, to problems whose states are bounded and live in $\mathbb{R}^n$, $n \leq 3$. To the author's knowledge, [58] presents the only work where dynamic programming has been used for control of a WEC, albeit for a model with many simplifying approximations. In [58], a simplified second-order WEC model is considered and then approximated by a discrete-time model. By imposing upper and lower bounds on the state and control inputs, quantization of the state variables gives rise to an approximate finite dimensional optimization problem where the input is assumed to take only its boundary values. The resulting NLP is then solved using dynamic programming. In the OCP considered in this chapter, as in most models in the literature, the WEC system order is higher than two and so a dynamic programming approach is not a feasible option.

The alternative is to solve the necessary conditions for optimality. Sufficient conditions for local optimality are then derived using any special structure in the solution resulting from the necessary conditions. The work in [59] derives second order sufficiency conditions (SSCs) for bang-bang trajectories for problems with a Hamiltonian affine in the input and with no state constraints. Similarly to [59] and under some strict conditions, [60] develops numerical methods for the verification of local optimality and demonstrates sufficiency for some example problems. Assuming that the number of switching times (or bang-bang arcs) is finite and its structure known a priori, the infinite dimensional OCP is reformulated into an equivalent finite dimensional one (i.e. NLP); the switching times $\tau = [t_1, \ldots, t_n]^T$ and some unknown initial conditions are the optimization variables. With an initial guess for these variables, the new problem is solved using appropriate NLP solvers. Numerically checking the positivity of the Hessian of the objective function with respect to $\tau$ evaluated at a solution $\tau^*$ would reveal

its local optimality. Although an indefinite Hessian would rule out local optimality, semi-definiteness would add no information.

The method is extended in [61] to optimal control problems with state inequality constraints that result in a mixture of bang-bang and boundary arcs; the latter are open time intervals where one or more of the state inequalities are active. In [61] it is also assumed that one would know a priori the structure and number of arcs so that, with the necessary switching junction conditions, the optimization problem can be reformulated into an NLP. The number and sequence of the different arcs can be known a priori in some applications [60, 61]. This would allow one to explicitly derive the state and adjoint dynamics in each arc as a function of the switching times, and with the necessary junction conditions. The resulting multipoint boundary value problem could then be solved using a shooting method to find the switching times; if such an indirect optimization approach is possible it may not even be necessary to use a direct ('first discretize and then optimise') method [62].

In the OCP considered in this chapter, it is impossible to know a priori the number of bang-bang arcs and the switching structure between the different arcs, since these depend on the wave excitation force and could not be known without solving the OCP for each excitation force sequence. The unavailability of this a priori information makes it impossible to solve the optimization problem using indirect methods as in [61]. For these reasons, we are not concerned with proving sufficiency of solutions, but have solved the problem using first-order necessary conditions in an indirect-approach or direct methods that discretize the problem into an NLP and solve an approximate problem. In the example simulations, we only consider input constraints. In addition to the gradient projection scheme, a direct transcription with state-of the art interior point optimization software will be used to solve the problem for comparison and validation.

### 3.3.5 Direct transcription and interior-point solvers

Here, a fundamental difference to the indirect approach is that there is no need to derive the necessary optimality conditions, or their associated adjoint dynamics and transversality conditions. In direct collocation schemes, both the states and the inputs are discretized in time within the optimal control horizon. The values of the state and inputs at the collocation points are then simply treated as variables for an NLP. The resulting finite dimensional problem is solved using any appropriate NLP solver.

Assuming as in (3.8) that the initial and final times $t_0$ and $t_f$ are fixed, we divide the optimal control horizon into $N$ intervals with grid points $t_0,\ t_1,\ \ldots,\ t_N$, where

$t_{j+1} := t_j + h$ and $h := \frac{t_f - t_0}{N}$. Let us now define as the NLP variables

$$z := [u_0^T,\, x_1^T,\, u_1^T,\, \ldots,\, x_{N-1}^T,\, u_{N-1}^T,\, x_N^T]^T, \tag{3.26}$$

where each $x_i \in \mathbb{R}^n$ and $u_i \in \mathbb{R}^m$ are the state and control input evaluated at the collocation point $t_i$, the initial state $x_0$ is assumed known. In parameterizing the problem, any path constraints and bounds on the states and inputs are imposed only at the grid points. Depending on the discretization scheme used, the system ODEs (3.7) are replaced by a finite set of new constraints. For example, if the forward Euler method is employed, we get the approximation

$$\dot{x} = f(x,u) \approx \frac{x_{i+1} - x_i}{h}, \tag{3.27}$$

where $f(\cdot)$ is the right hand side of the system ODE. For the $i^{\text{th}}$ interval, this enforces the equality constraint

$$g_i(z) := x_{i+1} - x_i - hf(x_i,u_i) = 0. \tag{3.28}$$

Stacking these $g_i(z), i = 0,1,\ldots,N$, we get a vector equality constraint $g(z) = 0$ for the NLP. Of course, the system dynamics can be approximated using any one of the many implicit or explicit numerical methods for solving ODEs. A list of explicit expressions of these collocation constraints for the classical Runge-Kutta, Trapezoidal and Hermite-Simpson methods can be found in [62, Sec. 4.5]. See also Section 5.4 in this thesis.

The cost function $J(\cdot)$ of the OCP in (3.9) is approximated in the NLP using the appropriate quadrature rule of the collocation scheme used; [62, Chap. 4] discusses a number of collocation schemes and their quadrature implementation. The optimization problem may come with inequality constraints on the inputs and/or states, $q(z) \leq 0$. With the new optimization variable $z$, the cost $F(\cdot)$ approximating $J(\cdot)$, a set of equality constraints $g(z) = 0$ and inequality constraints $q(z) \leq 0$ at the collocation points, a general constrained NLP can be expressed as:

$$\begin{aligned} \min_{z,s} F(z) \\ \text{subject to: } g(z) &= 0, \\ q(z) - s &= 0, \\ s &\geq 0, \end{aligned} \tag{3.29}$$

where $F : \mathbb{R}^{n_z} \to \mathbb{R}$, $g : \mathbb{R}^{n_z} \to \mathbb{R}^{n_e}$, $q : \mathbb{R}^{n_z} \to \mathbb{R}^{n_i}$ and $n_z$, $n_e$ and $n_i$ stand for the number of NLP variables, equality and inequality constraints, respectively and $s$ is called the

slack variable and allows us to formulate the $q(z)$ using an equality constraint. For our problem, the equality constraints arising from the system dynamics $g(z) = 0$ end up becoming nonlinear (specifically, bilinear) in the optimization variable $z$. The cost function is also a nonconvex function of $z$.

A variety of interior point methods (IPM) exist in the literature; a detailed study of the different methods and all the latest advances can be found in the book [63]. In this study we use the open source IPM software package Ipopt (Interior Point Optimizer, pronounced "I–P–Opt"). It implements the minimization (3.29) as the barrier problem

$$\begin{aligned} \min_{z,s} & F(z) - \mu \sum_{i=1}^{n_i} \log s_i, \\ \text{subject to: } & g(z) = 0, \\ & q(z) - s = 0, \end{aligned} \tag{3.30}$$

where $\mu$ is a positive parameter called the barrier parameter and $\log(.)$ represents the natural logarithm. The condition $s \geq 0$ is removed as the log barrier function (i.e. the fact that $-\log(s_i) \to \infty$ since $s_i \to 0$) prevents the components of $s$ from becoming negative. Let the Lagrangian associated with the optimization problem (3.29) be defined as:

$$\mathcal{L}(z, s, \nu, \phi) := F(z) - \nu^T g(z) - \phi^T (q(z) - s), \tag{3.31}$$

where $\nu \in \mathbb{R}^{n_e}$ and $\phi \in \mathbb{R}^{n_i}$ are the Lagrange multiplier for the constraints.

In an IPM, the optimal solution is obtained by solving the nonlinear optimality condition $\nabla\mathcal{L}(\cdot) := \nabla F(z) - \nabla g^T(z)\nu - \nabla q^T(z)\phi = 0$ together with the primal and dual constraints in (3.29). Newton's method is used to solve these. Applying what are called the second order Karush-Kuhn-Tucker (KKT) optimality conditions to (3.29) gives a system of equations, on which an application of Newton's method results in a set of primal-dual equations

$$\begin{bmatrix} \nabla^2_{zz}\mathcal{L} & 0 & -\nabla g^T(z) & -\nabla q^T(z) \\ 0 & \Phi & 0 & S \\ \nabla g(z) & 0 & 0 & 0 \\ \nabla q(z) & -I & 0 & 0 \end{bmatrix} \begin{bmatrix} p_z \\ p_s \\ p_\nu \\ p_\phi \end{bmatrix} = \begin{bmatrix} \nabla F(z) - \nabla g^T(z)\nu - \nabla q^T(z)\phi \\ S\phi - \mu\mathbf{1} \\ g(z) \\ q(z) - s \end{bmatrix} \tag{3.32}$$

$\mathbf{1} = (1, 1, ..., 1)^T$, $\Phi := \text{diag}(\phi)$ and $S := \text{diag}(s)$. This equation is solved to find the search direction $p := (p_d, p_s, p_\nu, p_\phi)$ at each step. A basic interior-point algorithm per-

forms the iteration:

$$\begin{bmatrix} z_{k+1} \\ s_{k+1} \\ \nu_{k+1} \\ \phi_{k+1} \end{bmatrix} := \begin{bmatrix} z_k \\ s_k \\ \nu_k \\ \phi_k \end{bmatrix} + \begin{bmatrix} \alpha_s^{max} p_z \\ \alpha_s^{max} p_s \\ \alpha_z^{max} p_\nu \\ \alpha_z^{max} p_\phi \end{bmatrix} \tag{3.33}$$

where $p = (p_x, p_s, p_y, p_z)$ is determined by solving (3.32) at $(z_k, s_k, \nu_k, \phi_k)$. The step lengths $\alpha_s^{max}$ and $\alpha_z^{max}$ can be chosen using a line search method to make sure constraints are not violated at the next step while making sure there is a significant decrease in the objective function at each iterate. For example, the steps

$$\begin{aligned} \alpha_s^{max} &:= \max\{\alpha \in (0,1] : s + \alpha p_s \geq (1-\tau)s\} \\ \alpha_z^{max} &:= \max\{\alpha \in (0,1] : \phi + \alpha p_\phi \geq (1-\tau)\phi\} \end{aligned} \tag{3.34}$$

with $\tau \in (0,1)$ with a typical value of 0.995 prevents $s$ and $z$ from getting to their lower boundary of zero too rapidly [63, Sec. 19.2]. An error function measuring how close an iterate is to satisfying the first order KKT condition can be used as a stopping criteria for the algorithm:

$$\begin{aligned} err(z_k, s_k, \nu_k, \phi_k) := \max\{&\|\nabla F(z_k) - \nabla g^T(z_k)\nu_k - \nabla q^T(z_k)\phi_k\|, \\ &\|S\phi_k - \mu_k \mathbf{1}\|, \|g(z_k)\|, \|q(z_k) - s_k\|\}. \end{aligned} \tag{3.35}$$

As $\mu_k$ is decreased towards zero during the iteration, the condition $err(x_k, s_k, y_k, z_k) \leq \mu_{k+1}$ can be used as a stopping criteria.

In summary, at each iteration of the IPM, the algorithm requires the evaluation of the objective function $F(z)$, the gradient of the objective function $\nabla F(z)$, the constraint function values $g(z)$ and $q(z)$ and their Jacobians $\nabla g(z)$ and $\nabla q(z)$, respectively. Where quasi-Newton approximations are not used, the Hessian of the Lagrangian is also computed analytically if the assumption that the objective and constraints are twice continuously differentiable is valid. In addition, most of the computations in the algorithm are performed in solving the linear equation (3.32) to find a search direction. Depending on the original optimal control problem, the numerical conditioning of this linear system affects the convergence of the IPM. See [63, Chap. 19] for a thorough exposition on theoretical and algorithmic aspects of IPMs.

### 3.3.6 Receding horizon control implementation

In the last few sections, we have analysed a model-based optimal control problem and developed algorithms for computing an input sequence that maximises energy yield

from predictions of the WEC states. This nonlinear MPC is an advanced control technique that has been widely used in industry [48]. Its success arises from its ability to explicitly deal with constraints and nonlinearities in a straightforward and transparent fashion. State and input constraints, which are often imposed by safety, cost and physical limitations, can be explicitly incorporated in the optimal control problem.

In this work, we apply nonlinear MPC in a receding horizon fashion [64]. As shown in Figure 3.2, at each sampling instant $t$, using the current plant state measurements as an initial value, an open-loop optimal problem is solved to generate an input signal $u(\cdot)$ over the finite prediction horizon, $[t,\ t+T_p]$. This nonlinear optimal control problem optimizes some objective function as in (3.8) for the wave energy maximization problem.

In the optimization, a nominal model of the system is evolved to find its future behaviour under the given input sequence. The input that yields an optimal predicted output while simultaneously satisfying all the constraints is chosen. If there were no model uncertainties and external disturbances, then this input sequence could be applied over all the prediction horizon. However, external disturbances do exist and plant models are never perfect. As a result, as depicted in Figure 3.2, the predicted and actual system behaviours are different implying the need for some form of feedback. The dotted curves represent the predicted state trajectory and the open-loop input at a given time instant. The solid lines show the part of the optimal predicted input sequence applied and the actual state trajectory in closed-loop.

Receding horizon control accomplishes feedback by applying only part of the computed input sequence and then using feedback by updating the model states; the control is applied over what is called the control interval $[t,\ T_c)$; this can be one sampling period for the input but often much longer (many multiples of the sampling time depending on the application). At time $t+T_c$, the optimal control problem is solved again over the new prediction interval $[t+T_c,\ t+T_c+T_p)$ using the new state measurements as the initial condition. As the prediction horizon slides along, this process of state updates using measurements, prediction and optimal control recalculation is repeated.

**Stability of the closed-loop control system**

A distinguishing feature of nonlinear predictive control we derive by solving the OCP over a finite prediction horizon is that it is not a regulation problem. In standard regulation problems for linear and nonlinear systems, the cost function is often a Lyapunov function that enables one to show stability of a system equilibrium under robust feedback control laws. Nominal stability of a receding horizon implementation is often assured by applying terminal state costs, restricting the terminal state in some small set around the equilibrium or assuming infinite prediction horizons [64]. In fact, the

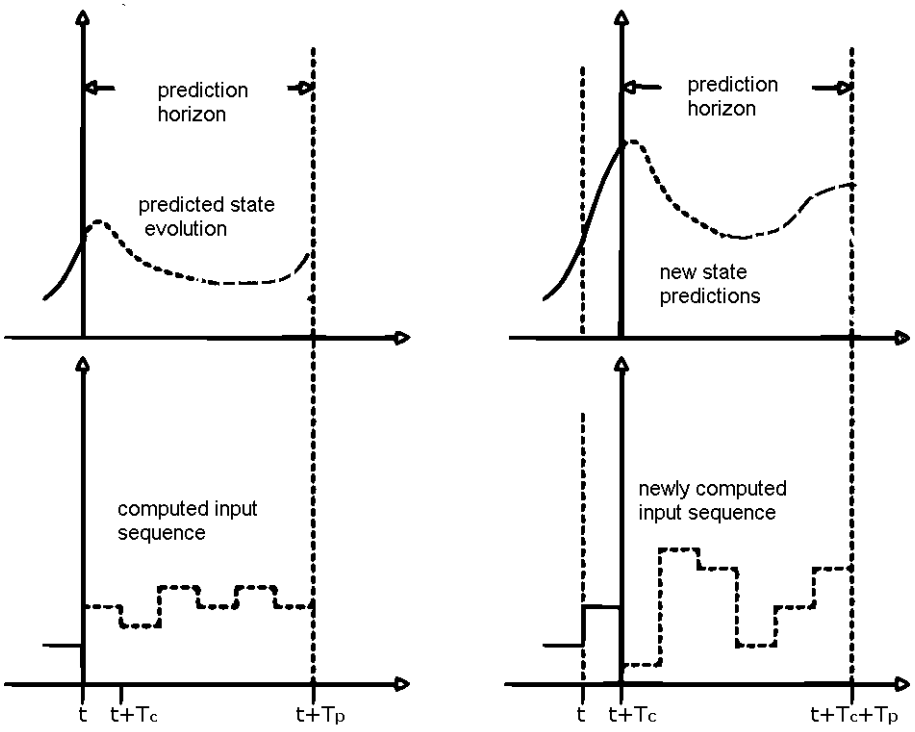


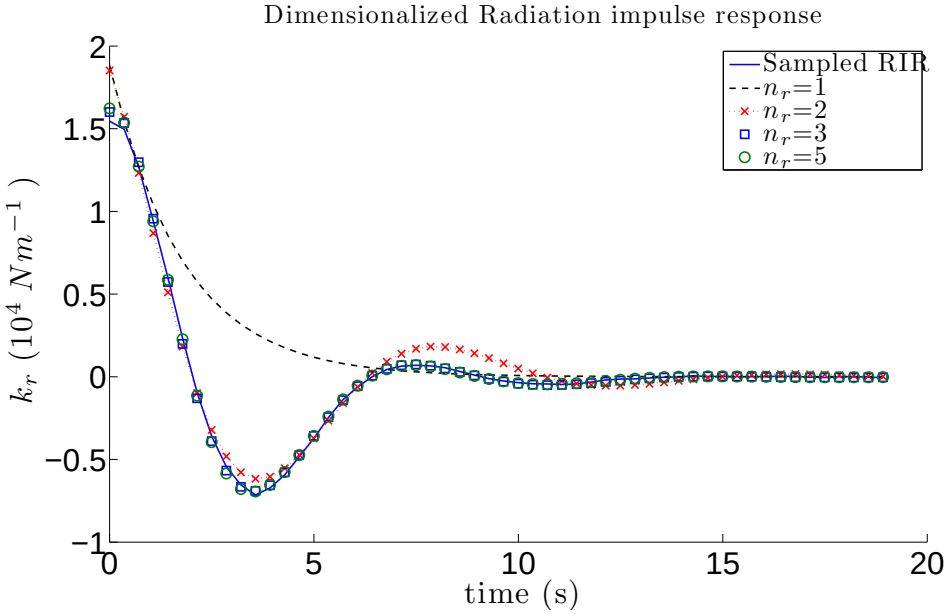

Figure 3.2: Receding horizon control

problem we solve can be considered as an example of an economic MPC problem [65]; we are maximising some economic performance with constraints that arise from the control actuation mechanism. In this setting, the objective is not regulation around an equilibrium point or path. The nonconvexity of the cost function and nonlinearities of the dynamics can mean that the best performing evolution of the states may not be an equilibrium point [66]. In fact, for our problem, it is known that the best trajectory is one that very much depends on the wave excitation force and results in the buoy moving 'in phase' with the exciting force.

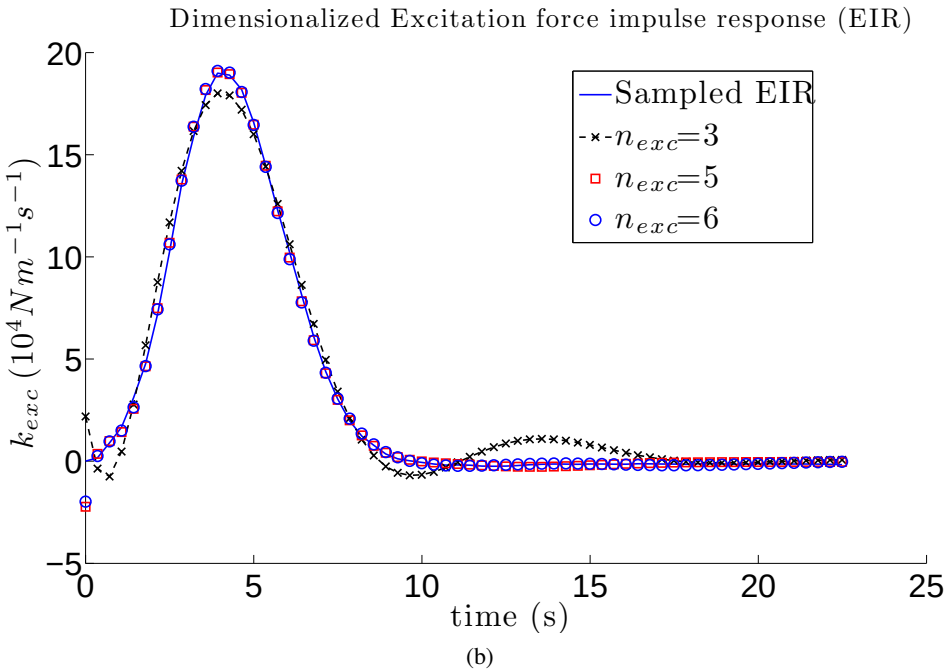
Generally, proving closed-loop stability is hard for receding horizon economic MPC problems and still remains a challenging research problem. We refer the reader to [65, 66] for discussions on proving stability for some economic MPC optimizing problems; the difficulty for general nonlinear systems with (nonconvex) economic MPC type costs is also discussed therein. Therefore, as often done in practice, we leave the assessment of closed-loop stability to simulation results.

## 3.4 Example Simulations

For the semi-submerged heaving cylinder that we consider, non-dimensionalized impulse response kernels for the radiation and excitation forces from [15] were used. We




(a)




(b)

Figure 3.3: (a) A radiation subsystem of order $n_r = 5$ appears good enough to approximate the radiation impulse response sufficiently. (b) A 6th order subsystem can be used to model the "causalized" excitation force; $n_{exc}$ is the order of the linear subsystem that approximates the excitation force response.

scale the problem to an appropriate size roughly comparable to the device in [5]; a cylinder of 5 m radius, and height of 20 m with a spring of stiffness $k = 240$ kN/m will be used. From the dimensionalization relation used in [15], we calculate the draught to be 9 m in 42.85 m deep waters. The spring is assumed slack at equilibrium (no wave), and the mass of the device is $M = 707$ t (tonnes) with $\mu_\infty = 0.345 \times M \approx 244$ t from the relation in [15] and water density $\rho = 1000$ kg/m$^3$.

Figure 3.3 shows that even a 3$^{\text{rd}}$ order radiation subsystem is enough to approximate the sampled radiation impulse response; the behaviour of the model impulse response changes little as we increase the order beyond 3. We also generate the excitation force using a 6$^{\text{th}}$ order state space model approximating the "causalized" excitation impulse response and driven by the wave height data at the buoy [15].

### 3.4.1 Projected gradient algorithm performance

The projected gradient method was tested to investigate its convergence properties under different wave conditions. The results were compared to the results from solving a direct transcription of the optimal control problem using a state of the art open-source optimization software (IPOPT, version 3.9.2) [67]. Euler, trapezoidal and Hermite-Simpson collocation schemes were successfully used for the IPOPT implementation (see [62, Sec. 4.5]). The explicit Runge-Kutta (4,5) and a variable order solver based on numerical differentiation formulas (Matlab's `ode45` and `ode15s`, respectively,) were used to integrate the dynamics and adjoint dynamics within the PGM implementation. Our implementation in IPOPT adds the input constraints to the objective function using barrier functions to form a Lagrangian. At each iteration, gradient and Hessian computations of the Lagrangian, as well as a number of line searches are performed (see Section 3.3.5). On the other hand, the PGM method described in Section 3.3.3 requires only a single state and adjoint state evaluation; the gradient computation and the projection onto the feasible input set add only marginal computational cost.

In all the simulations, the control inputs have a sampling period of 0.1s; the states and adjoint states are resolved at a 5 times finer rate. IPOPT was set to use an adaptive barrier parameter update strategy since it resulted in better convergence in simulations. For the PGM, a value for the constant step-size was chosen a posteriori from simulations. A value of $s = \frac{20}{G}$ was found to work sufficiently well under all the wave conditions presented. It can be seen from Figure 3.4 that the PGM converges more quickly than IPOPT to the same local optima. The test was done under different wave conditions, parameter values and WEC initial conditions to confirm similar performances. A snippet of the device response under the controller is shown in Figure 3.5. An interesting point to note is that the damper is off (or bypassed) when the active controller is aiding the

motion (i.e. when the control input $u_1$ has the same sign as the velocity $\dot{\zeta}$) and engaged when the active element resists motion.

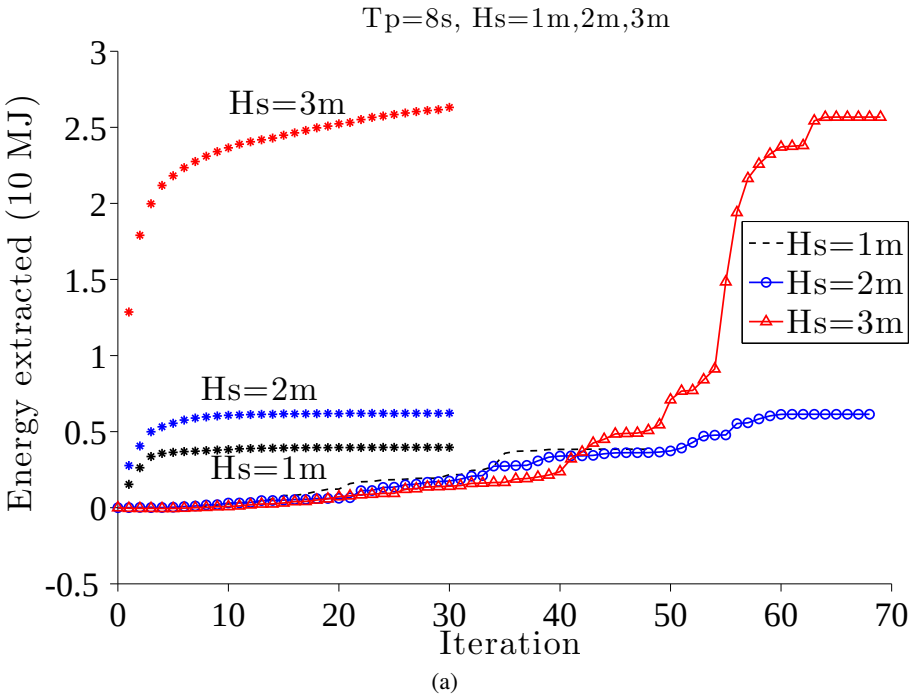



Figure 3.4: Convergence of the PGM algorithm (dotted, left) against that of a direct transcription solution using IPOPT (lines with marker) over a 50 s prediction horizon (a) for waves of the same typical period ($T_p$) and different significant wave heights($H_s$) (b) for waves of different typical period ($T_p$) but of the same significant wave height.

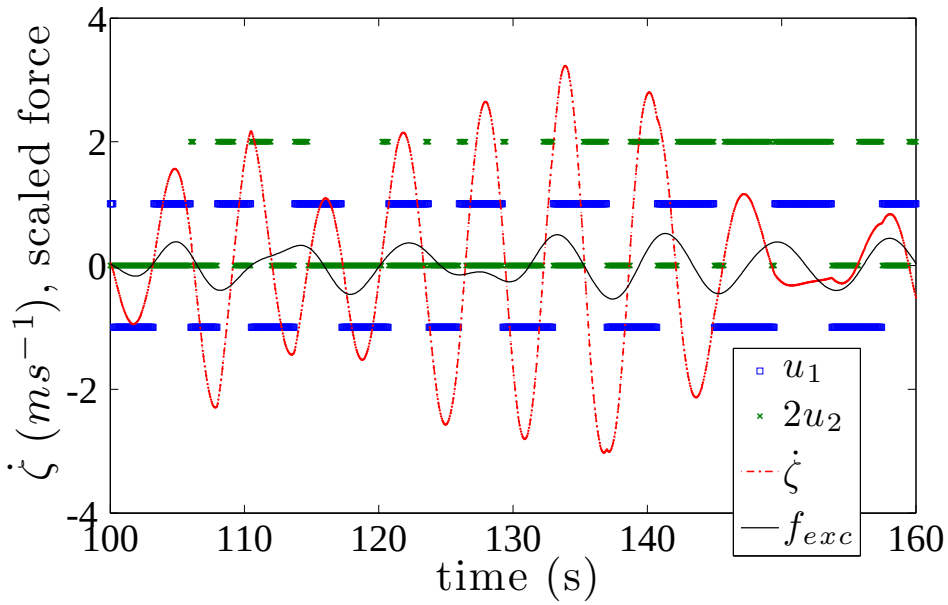



Figure 3.5: A plot of the device velocity against control commands and normalized excitation force under 'Method 1'. The blue plot switching between +1 and -1 is the active control. The green plot switching between 0 and +2 is the damping control times 2; it is multiplied by 2 for clarity of presentation.

### 3.4.2 Device optimization for control

In addition to being optimized for a given wave climate, the parameters of a PTO should be designed with a specific control scheme in mind; it is shown here that a design that is optimal under one control scheme may not be optimal under a different control scheme. For example, [68] shows that, under a given sea condition, the optimal damping coefficient of a generic point absorber is very different depending on whether latching control is used or the device is uncontrolled. We have made simulations to choose the control parameters $B_{pto}$ and $G$ in an optimal way for a given wave environment and control scheme.

We will label the active control scheme with control over both the damping and active PTO elements, and discussed in Section 3.3.1, 'Method 1'. Similarly to this, we also consider an actuation mechanism where the damper of the PTO is not controlled and is always on. A similar optimal control problem formulation and numerical scheme was used; only one input is considered, i.e. the active control input $u_1$; $u_2(t) = 1$, $\forall t$. The resulting optimal controller can be shown to be bang-bang. We call this case where the linear damper is always engaged 'Method 2'. We also compare these two control schemes with optimal latching control and the uncontrolled system. The optimal command latching scheme employed is exactly as in [5]. A parameter optimization for the latching control revealed that $B_{pto} \approx 95.1$ kN·s/m gives the best results under the same

wave conditions stated.

For Method 1, simulations indicate that energy yield increases with both parameters until it flattens. Figure 3.6a shows a sweep of the parameter values against average power delivered when the active controller of the previous section is used over 50 s prediction horizons; here the computed controls are applied over the whole 50 s. The simulations were carried out over 1000 seconds with a wave from a JONSWAP spectrum of typical period $T_p = 8$ s.

Figure 3.6b shows a parameter sweep for Method 2. The optimal damper value is $B_{pto} \approx 280$ kN·s/m for the range of $G$ shown. Here the surface is concave and further increasing the damping coefficient decreases yield, unlike in the case where the damping element is controlled. It seems that higher values of a damper element do not result in higher yield unless the element is controlled. This, perhaps, explains why the optimal damping coefficients in declutching control tend to be much higher than ones for latching control (as is apparent from results in [69]). Thus, the optimization of PTO parameters should heavily depend on the control scheme intended.

The power function in Figure 3.7 shows both active controllers and the optimal latching result in an increase in extracted energy compared to the hydraulic PTO with no control. As expected, the latching controller enlarges the bandwidth only towards low frequencies. Unlike latching control, the active methods widen the bandwidth of the WEC in both directions around the resonant frequency; latching is effective only at frequencies lower than that of the buoy [5]. Method 1 also shows consistently better performance compared with Method 2. Since linear dampers can be switched on and off using a simple by-pass valve, optimally controlling passive PTO elements for better performance can be justified.

### 3.4.3 Prediction horizon sensitivity in a receding horizon implementation

In the preceding discussion, a 50 s prediction horizon was used for device parameter optimization under ideal conditions where the whole predicted control is applied — no disturbances were applied and the wave height was assumed known. In a real implementation, optimal controllers are computed over a prediction horizon and then only part of the control is applied over the control interval $T_c$; this can be one sampling period but often much longer (many multiples of the sampling time depending on the application). Here, we investigate the performance of the method against varying prediction and control horizons.

Figure 3.8 shows that the power output increases with prediction horizon length until it flattens around three times the typical period value. We also show the use of different

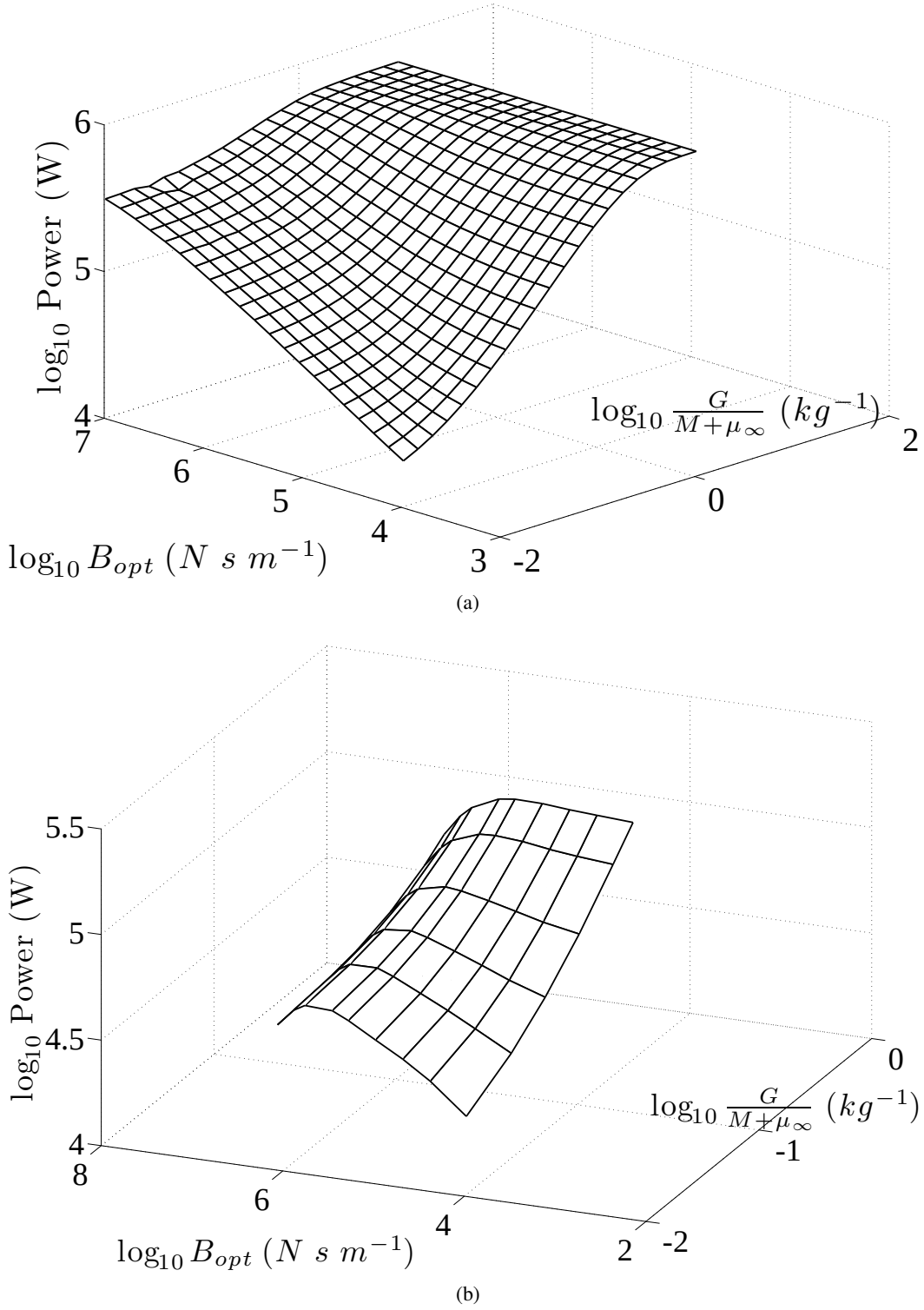



Figure 3.6: Variations in average absorbed power (W) against parameters $G$ and $B_{pto}$ for: (a) Method 1 (b) Method 2, $T_p = 8$ s, $H_s = 2$m.

control intervals ($T_c$). Although decreasing $T_c$ gives better performance in general, its effect can be negligible for large prediction horizons; compare the 20kW increase in

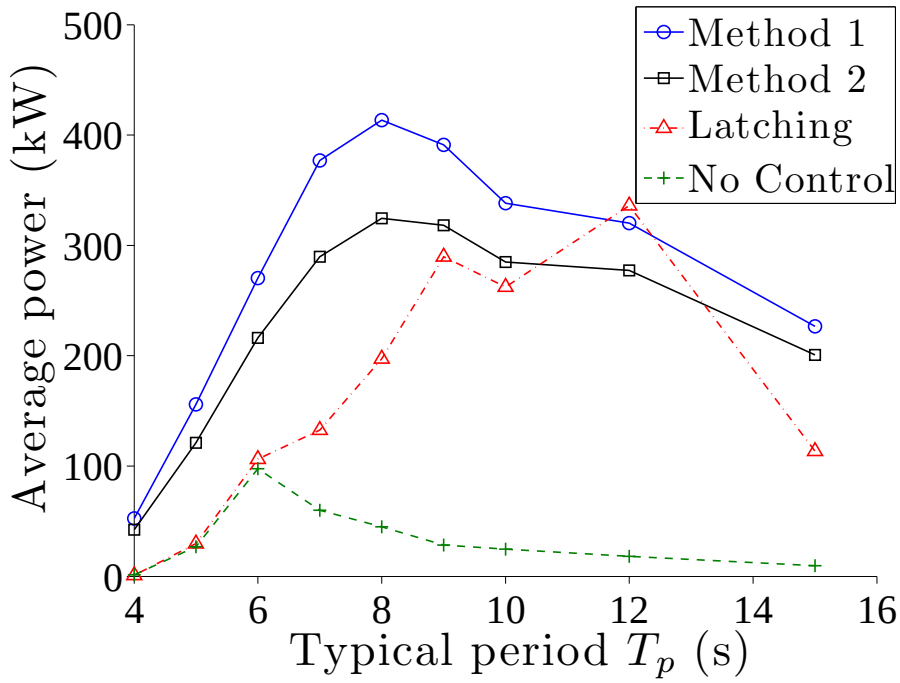



Figure 3.7: Average absorbed power against typical wave period with the different control methods, $B_{pto} \approx 280$ kN·s/m ($B_{pto} \approx 95$ kN·s/m for latching control), $G = M + \mu_\infty$, $T_p = 8$ s, $H_s = 2$m.

power when $T_c$ is halved at $T_p = 3$s with the 0.1 kW difference at $T_p = 8$s. The integration of multi-step wave excitation prediction schemes in this sensitivity analysis, as done using an extended Kalman filter in [44], can also be considered.

## 3.5 Conclusion

In this chapter, model predictive control strategies were considered for implementation in a receding horizon fashion. A state space model of a generic point absorber, whose power take-off includes a linear damper and an active element, was formulated and used. By considering a variational formulation of the optimal control problem with constraints only on the inputs, the solution was shown to be a bang-bang type.

A computationally inexpensive and globally convergent numerical scheme was developed for solving the power maximization problem. A variation of the projected gradient method (PGM) was exploited and shown to converge in a small number of iterations under various wave conditions. Its performance has been compared to solving a directly collocated version of the problem using an interior point solver, IPOPT. It was shown that the PGM requires only a single state and costate evaluation at each iteration and is far less computationally demanding compared to a general NLP solver. For example, at each iteration IPOPT performs gradient and Hessian computations of the Lagrangian,

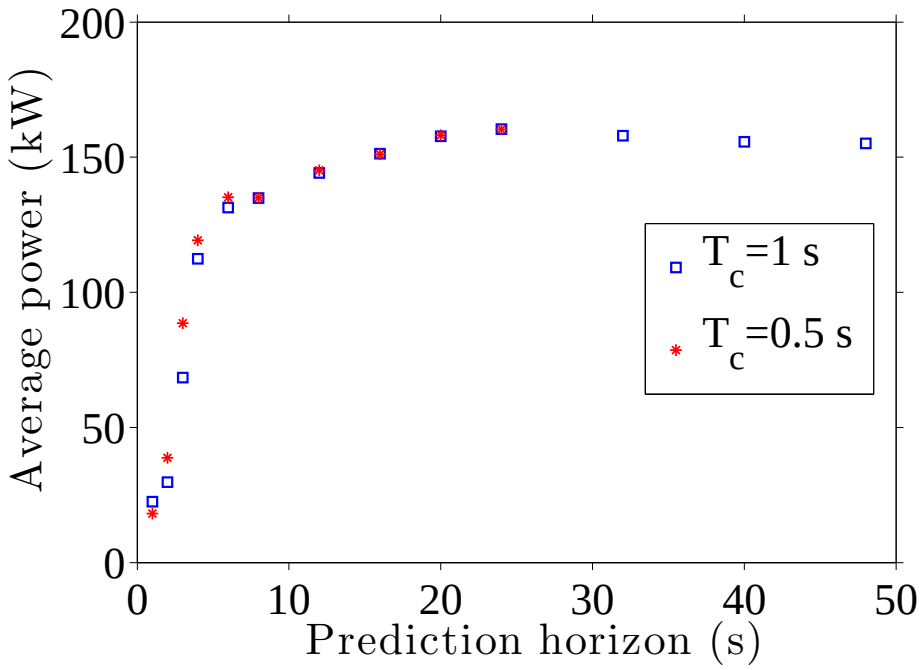



Figure 3.8: Variations in average absorbed power against prediction horizon lengths for 'Method 1'. JONSWAP wave with $T_p = 8$ s and $H_s = 2$ m used; $G = B = 0.3 * (M + \mu_\infty)$.

as well as a number of line searches by solving a KKT system (see Section 3.3.5). On the other hand, the PGM method described in Section 3.3.3 requires only a single state and adjoint state evaluation; there is only marginal computational cost associated with the gradient computation or the input projection onto the feasible set . For this reason, to further reduce the cost incurred in the solution of the system and adjoint ODEs, we consider the use of exponential integrators in Chapter 5.

Time-domain simulations have also been used to evaluate the performance of the controllers developed in a receding horizon scheme. The active methods have been compared with the case of optimal command latching and the system with no control. The controllers developed were shown to widen the bandwidth of the wave energy converter and increase extracted power significantly. Unlike latching control, the active methods widen the bandwidth of a WEC in both directions around the resonant frequency; latching is effective only at frequencies lower than that of the buoy.

The PTO system optimization was shown to be important and highly dependent on the control scheme used. Although more energy could potentially be produced using larger actuation forces, physical constraints related to the maximum pressures and loads that actuation systems can withstand or deliver should be taken into account. In practice, the efficiency of the PTO should also be considered in the control problem and device optimization.

Similarly to other works in the literature, the availability of future excitation force information and radiation forces has been assumed; this is not true in practice. The synthesis of observers for the radiation force and their use is also the subject of this thesis and is discussed in Chapter 4.

# Chapter 4

# Robust Estimator Design for Bilinear Systems with Bounded Inputs

This chapter investigates low-order observer design for bilinear systems with input constraints. A bilinear Luenberger-type observer with an $H_\infty$ performance is formulated and the resulting synthesis problem is posed as a matrix inequality optimization for a linear parameter varying system. The resulting (nonconvex) bilinear matrix inequality problem is then solved with an LMI-based algorithm to find low-order nominal and robust quadratically stable observers. The performance of these observers is compared with that of an extended Kalman fiter. In addition to alleviating the need to know the noise spectrum and its lower real-time computational burden, the $H_\infty$ filter is shown to be robust to model uncertainties. The online radiation force estimation problem for a wave energy converter with bilinear dynamics is considered as an example. A small order radiation subsystem is used to approximate the radiation process and the robustness of the observer to both model order and parameter uncertainties is investigated. Closed-loop receding horizon simulations are also used to assess the effectiveness of the observers under noisy measurements.

## 4.1 Introduction

Bilinear systems have dynamics that are linear in the input and linear in the state, but not jointly linear in both. These systems arise in a variety of modelling processes, including the discretization of certain partial differential equations with boundary actions or the bilinearization of some nonlinear systems, within various application areas such as plasma physics, quantum physics, biomedicine, metallurgy, economics, ecology and so on [70]. Motivated by estimation problems in wave energy conversion [71], in this chap-

ter we investigate the design of observers for bilinear systems with input constraints. We consider observer design for such systems where physical actuator constraints give rise to convex input constraints [71]. Motivated by this problem, we will investigate the design of observers for bilinear systems with bounded inputs.

The literature on full-order $H_2$ and $H_\infty$ filters for linear systems is vast. The synthesis of full-order $H_2$ filters is a well studied convex problem; the analysis and synthesis of $H_2$ filters in the presence of polytopic uncertainty is discussed in [72, Ch. 9] and the literature therein. However, the filter performance degrades for a general uncertainty in the model or when the spectral density of the noise is not perfectly known — the optimal minimal error variance in the estimation error is no longer guaranteed. In comparison, $H_\infty$ filters require no statistical knowledge of the input or disturbance, other than that its energy be bounded. In addition to being inherently robust to model uncertainty, the analysis and synthesis of $H_\infty$ filters also allows the explicit incorporation of various uncertainty types [73–75]. These cited works consider a system with structured and unstructured uncertainty and synthesize full-order $H_\infty$ filters with respect to the given uncertainty sets. This problem can be posed as a linear matrix inequality (LMI), which is a convex optimisation problem and so can be solved efficiently due to recent advances in the discipline [72]. As we review in the following, LMI based $H_\infty$ analysis and synthesis tools have been generalized to some nonlinear systems and bilinear systems. However, these standard methods produce observers that have the same order as the model. For high order systems, this may pose a high computational burden or even infeasibility for real-time implementation. In the following, we investigate the design of full-order and lower-order $H_\infty$ filters for bilinear systems.

This chapter is organised as follows. In Section 4.2, we define the bilinear systems of interest and review existing literature on observer design for such systems. Stability and quadratic performance of LPV systems is reviewed in Section 4.3. In Section 4.4, we formulate the bilinear system with input constraints as a linear parameter varying (LPV) system and show that the lower-order $H_\infty$ filter synthesis can be posed as a bilinear matrix inequality (BMI) problem. Section 4.5 presents an LMI-based coordinate descent algorithm for solving the BMI locally. We also give the dynamics for an implementation of an extended Kalman filter for comparison with the $H_\infty$ filter. In Section 4.7, we compare the two filters by examining the robust radiation force estimation problem for a wave energy converter (WEC) in [71].

## 4.2 Problem Statement and Related Work

We consider continuous-time bilinear systems of the form:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + \sum_{i=1}^{m} u_i(t)N_i x(t) + Bu^l(t) + Rw(t), \\ y(t) &= Cx(t) + Dv(t), \\ z(t) &= Lx(t), \quad x(0) = x_0, \end{aligned} \tag{4.1}$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $y(t) \in \mathbb{R}^p$ is the measured output and $z(t) \in \mathbb{R}^r$, $r \leq n$, is a linear functional of the states to be estimated. The disturbance signal $d(t) := [w(t)^T \; v(t)^T]^T \in \mathbb{R}^q$ is a vector containing both process noise $w(t)$ and measurement noise $v(t)$. The control input $u(t) := [u_1(t) \; \ldots \; u_m(t)]^T \in \mathbb{R}^m$ is known and enters the dynamics through the bilinear terms whereas $u^l(t) \in \mathbb{R}^k$ represents known control inputs entering the dynamics linearly, hence the superscript $l$ to denote this. In (4.1), $A, \{N_i, \; i = 1, \ldots, m\}, B, R, C, D, L$ are constant real matrices with compatible dimensions. Our objective is to design a dynamic estimator that will produce estimates of $z$ from the measurement $y$.

An important assumption we make here is that the control inputs $u(t)$ are bounded, i.e. $u(t) \in \mathbb{U}$, $\forall t \geq 0$, where $\mathbb{U} := \{u(t) \in \mathbb{R}^m : |u_i(t)| \leq 1, \forall t \geq 0, \; i = 1, \ldots, m\}$. This unit constraint is general enough to include any finite bounds on the component-wise inputs since any such bounds can be reduced to this form with a simple linear transformation. If $u_i(t) \in [a_i, b_i]$, $\forall t$, and some $a_i, b_i \in \mathbb{R}$, substituting $\tilde{u}_i(t) \in [-1, 1]$ for $u_i(t)$ such that $u_i := \frac{b_i - a_i}{2}\tilde{u}_i + \frac{b_i + a_i}{2}$ gives a bilinear system in the standard form. The new control input $\tilde{u}(t) \in \mathbb{U}$, $\forall t \geq 0$ and the system matrices change appropriately; $N_i \leftarrow \frac{b_i - a_i}{2} N_i$ and $A \leftarrow A + \sum_{i=1}^{m} N_i \frac{b_i + a_i}{2}$.

The LMI setting has been used in full-order $H_\infty$ filter design for a variety of Lipschitz nonlinear systems and bilinear systems. The works in [76–79] make use of Lyapunov quadratic functions both to check the existence of and synthesize stabilizing Luenberger-type nonlinear observers. Sufficient conditions for the existence of such a filter are given in [76] for a nominal system with no disturbances. By using input-to-state stability (ISS) theory, [77] gives LMI conditions for guaranteeing asymptotic stability of the estimation error; the same LMIs are then adapted for filter design in the presence of process and measurement noise. In [78], in addition to Lipschitz nonlinearities and disturbances, linear time-varying uncertainties are also considered. An $H_\infty$ observer is designed using LMI techniques; a multi-objective convex optimization problem is solved to guarantee a given error decay rate and $H_\infty$ disturbance attenuation level for a given Lipschitz uncertainty set. Similarly, the work in [79] considers a weighted

convex minimization problem; the minimum eigenvalue of a Lyapunov matrix is maximized while minimizing the induced $L_2$ gain from the disturbance to the estimation error. Although these methods are applicable to full-order filter design for the bilinear case (the bilinear terms with bounds on the input are Lipschitz nonlinearities in the states), recent literature has rather pursued a less conservative approach using polytypic linear system representations for the bilinearities [80, 81].

The works in [80, 81] and preceding works by the same authors and those referenced therein consider what they call low-order functional $H_\infty$ filters. They propose a filter of the form

$$\begin{aligned} \dot{\eta}(t) &= H_0\eta(t) + \sum_{i=1}^{m} u_i H_i \eta(t) + J_0 y(t) + \sum_{i=1}^{m} u_i(t) J_i y(t) + Gu(t), \\ \hat{z}(t) &= \eta(t) + Ey(t), \end{aligned} \tag{4.2}$$

where the state vector of the observer $\eta(t) \in \mathbb{R}^r$, $r \leq n$, is the same size as $z(t)$ and all the observer system matrices in (4.2) are to be determined.

This formulation has some attractive properties. Firstly, the order of the observer dynamics is $r$, which can be much less than $n$ depending on the problem. Moreover, the Lyapunov approach is used to transform quadratic stability and $H_\infty$ disturbance attenuation conditions on the error system into LMIs, which are solved using efficient interior-point and semidefinite program (SDP) solvers. In addition, this framework also allows the explicit incorporation of robustness against other linear parameter uncertainties.

One of the main constraints in [80, 81] is the objective to make the filter unbiased, i.e. the dynamics of the filtering error ($e = z - \hat{z}$) is independent of the state $x$ and input $u$ of the system (4.1). This assumption on the filter is used to derive a number of algebraic conditions (a set of Sylvester equations) that must be solvable to guarantee the existence of such a stabilizing observer. These equations admit a solution if and only if some rank constraints [80, Eq. 15] are satisfied; this condition roughly requires that the space spanned by $(LA_0, \ldots, LA_m, CA_0, \ldots, CA_m, L, C)$ be the same as the one spanned by $(CA_0, \ldots, CA_m, L, C)$. This purely technical condition may not be met by many systems of the form (4.1); our motivating example (see Section 4.7) does not satisfy this condition.

For systems where these nominal unbiasedness constraints are met, the observer matrices are parameterized by the solutions of the unbiasedness rank constraints, which results in an error dynamics with nonlinear terms in the unknown parameters. These nonlinear terms are linearized by arbitrarily setting some nonlinear terms to zero to formulate the quadratic stability and $L_2$ gain attenuation conditions of the error dynamics

as an LMI. In [81], by considering the control inputs as unknown but bounded uncertainties, the error dynamics is reduced to a linear system with some structured uncertainties in the $A$-matrix. In addition to the nominal case, norm bounded uncertainties on the system matrices are also considered in [81], resulting in non-convex matrix inequality problems whose relaxations are solved. In [80], only the nominal $H_\infty$ functional filter is considered, i.e. no model uncertainties are considered. However, the real-time measurable inputs are used to pose the bilinear system as a linear parameter varying (LPV) system; this reduces the conservatism inherent in the formulation of [81] where the inputs are treated as bounded (but unknown) structured uncertainties. Moreover, by considering a boundedness assumption also on the derivative of the control $\dot{u}$, parameter-dependent Lyapunov matrices are used in [80] to further reduce conservativeness.

The approach taken in this thesis will use a combination of ideas from the literature discussed above. As in [76, 77, 79], a reasonable structure for the observer is to use a copy of the system dynamics plus an output feedback term—a bilinear Luenberger-type observer. Similarly to [80], we use an LPV formulation of the bilinearities in the analysis of the error dynamics and synthesis of an observer; this should avoid the conservatism introduced in [81]. Nonetheless, since we are considering applications with possibly arbitrarily fast control rates, an example is the case where bang-bang type control inputs are used as in [71], we do not require that the derivative of the control input to be bounded. Unlike in [80], our LPV formulation will guard against arbitrarily fast input variations by using constant Lyapunov matrices.

In the method we propose, we do not require the low-order model used in observer design to be stable. Unlike in standard $H_\infty$ filter synthesis for LPV systems, we require only that the error dynamics of the observer be stabilizable by output feedback. This is very advantageous in cases where the low-order model comes from model reduction or a coarse discretization of PDEs [82] and stability may not be preserved for some low-order models.

It can also be shown that the algebraic conditions that guarantee unbiasedness are not preserved when there is uncertainty in the model [83]; since most systems will have uncertainty in the model, its utility is limited. In our work, we do not enforce an unbiasedness condition on the error dynamics. We show in Section 4.4 that a quadratically stable observer can be derived without the need for unbiasedness. This leads to an alternative synthesis method for lower-order $H_\infty$ filter design. In the next section, we review quadratic Lyapunov stability and performance and their application to LPV systems.

*Notation:* In the rest of this chapter, $P^T$ denotes the transpose of the matrix $P$, $P < 0$ ($P \leq 0$) represents a symmetric negative definite (semidefinite) matrix. $L_2[0,T)$ is the

standard (time-domain) Lebesgue space $L_2[0,T) := \{f : [0,T) \to \mathbb{R}^n | \int_0^T \|f(t)\|_2 dt < \infty\}$, with Euclidean norm $\|f\|_2 := \sqrt{(f^T f)}$.

## 4.3 Quadratic Stability and Performance for LPV systems

For linear time invariant (LTI) systems, it has long been known that notions of asymptotic stability and quadratic stability are equivalent; proving the existence of a quadratic Lyapunov function is equivalent to proving asymptotic stability. Similarly, performance measures have also been posed as equivalent quadratic Lyapunov function based conditions; see [84] for $H_\infty$ and $H_2$ performance criteria posed as LMIs.

However, this equivalence does not hold in the case of nonlinear systems, linear time-varying systems or linear parameter varying systems. Unlike for LTI systems, quadratic stability and performance analysis give only sufficient conditions. See [84, p. 73] and references therein for examples of polytopic linear differential inclusion (PLDI) systems that are asymptotically stable but not quadratically stable; the same has been confirmed for switched systems, which can be viewed as a subset of PLDIs with arbitrarily fast parameter variations [85]. However, as explained in [86], even for an autonomous linear polytopic system, if the parameters vary arbitrarily fast, determining asymptotic stability using general (non-quadratic) methods is an NP-hard problem. It is also known that [86, Thms 3–4] an LPV system with arbitrary time variations is asymptotically stable if and only if "a trajectory dependent quadratic Lyapunov function" exists. Since computing general (nonquadratic) Lyapunov functions or trajectory dependent Lyapunov functions (i.e. for all possible trajectories) is computationally infeasible, most literature adopts quadratic Lyapunov function based analysis and synthesis [84–86]. Although only sufficient, these give us a computable set of conditions for stability in the form of LMIs.

Similarly, quadratic Lyapunov theory gives us a way to compute the exact $L_2$-gain performance for LTI systems (i.e. $H_\infty$ norm for transfer functions of LTI systems). On the other hand, only upper bounds for the $L_2$ gain of PLDIs, LPVs and some nonlinear systems can be derived using quadratic Lyapunov functions [84]. Below, we give some definitions and propositions that will be useful in the analysis of the bilinear system we consider.

**Proposition 4.1.** *[87, Prop. 1.19] Let $f : \Delta \to \mathbb{R}$ be a convex function where $\Delta = \text{conv}(\Delta_0)$. Then, $f(\delta) \leq \mu$ for all $\delta \in \Delta$ if and only if $f(\delta) \leq \mu$ for all $\delta \in \Delta_0$.*

This proposition plays a vital role in quadratic Lyapunov based analysis and synthesis

of for PLDI systems; it reduces the verification of stability and performance conditions to the vertices, which are of finite number for these systems. To define some useful terms, we consider the linear differential inclusion (LDI)

$$\begin{aligned} \dot{x}(t) &= F(t)x(t)+B_d(t)d(t), \quad \zeta(t)=C_\zeta(t)x(t), \\ & x(0)=x_0, \quad \begin{pmatrix} F(t) & B_d(t) & C_\zeta(t) \end{pmatrix} \in \Omega_F \times \Omega_B \times \Omega_C, \end{aligned} \tag{4.3}$$

where $d(t)$ is an exogenous input, $\zeta(t)$ a linear function of the state $x(t)$ and $\Omega_A \times \Omega_B \times \Omega_C$ is a polytope.

In the case where $d(t)$ is identically zero, a sufficient condition for the asymptotic stability of the LDI (4.3) is the existence of a quadratic Lyapunov function.

**Definition 4.1.** *[84, p. 61] For $d=0$, the system (4.3) is called quadratically stable if there exists a constant matrix $P=P^T>0$ such that*

$$F(t)^T P + PF(t) < 0, \text{ for all } F(t) \in \Omega_F. \tag{4.4}$$

Of course, it is easily verified that the existence of such a symmetric positive definite matrix $P$ guarantees that the positive quadratic function $V(x)=x^TPx$ is a Lyapunov function; $\frac{dV(x(t))}{dt} = x^T(F(t)^TP+PF(t))x < 0$ for all nonzero trajectories if and only if $F(t)^TP+PF(t)<0$.

In order to characterise the performance of the estimator with respect to disturbances, we give the following definition as a generalisation of the $L_2$ gain for linear systems.

**Definition 4.2.** *[29, Def. 5.1]*

*Consider the LDI (4.3) as a mapping from $d$ to $\zeta$. If there exist nonnegative constants $\gamma$ and $\beta$ such that*

$$\int_0^\infty \|\zeta(t)\|_2^2 dt \le \int_0^\infty \gamma^2 \|d(t)\|_2^2 dt + \beta, \tag{4.5}$$

*for all $d \in L_2[0,\infty)$ and all bounded $x(0)$, then we say the $L_2$ gain is less than or equal to $\gamma$. The mapping has $L_2$-gain $< \gamma$ if there exists some $\hat{\gamma} \ge 0$ such that (4.5) holds for $\hat{\gamma} < \gamma$. The smallest $\gamma$ for which there is a $\beta$ such that (4.5) holds is called the $L_2$ gain of the system.*

**Proposition 4.2.** *Suppose there exists a quadratic Lyapunov function $V(x) := x^TPx$, $P > 0$ for the system (4.3) and some $\gamma \ge 0$ such that $\forall t \ge 0$*

$$\frac{dV(x(t))}{dt} + \zeta(t)^T\zeta(t) - \gamma^2 d(t)^T d(t) \le 0 \quad \text{for all trajectories of (4.3) with any bounded initial condition } x(0)=x_0. \tag{4.6}$$

*Then, $\gamma$ is an upper bound for the $L_2$ gain for the system* (4.3) *as in Definition 4.2.*

*Proof.* Integrating (4.6) and using the facts $V(x(t)) \geq 0$, $\forall t \geq 0$, gives in the limit

$$\int_0^\infty \left(\zeta(t)^T \zeta(t) - \gamma^2 d(t)^T d(t)\right) dt \leq V(x_0). \tag{4.7}$$

Putting $V(x_0)$ in place of the term $\beta$ in (4.5), the fact $V(x_0) \geq 0$ proves the claim. □

Simply, substituting for $\frac{dV(x(t))}{dt}$ in (4.6), the bounded real lemma [84] gives us the equivalent matrix condition below.

**Proposition 4.3.** *(The bounded-real lemma [84]) For some $\gamma \geq 0$, if there exists $P = P^T > 0$, such that $\forall t \geq 0$*

$$\begin{bmatrix} F(t)^T P + PF(t) + C_\zeta(t)^T C_\zeta(t) & PB_d(t) \\ B_d(t)^T P & -\gamma^2 I \end{bmatrix} \leq 0, \tag{4.8}$$
$$\begin{pmatrix} F(t) & B_d(t) & C_\zeta(t) \end{pmatrix} \in \Omega_F \times \Omega_B \times \Omega_C,$$

*then $\gamma$ is an upper bound for the $L_2$ gain for the system* (4.3).

### 4.3.1 Linear parameter varying systems

LPV systems can be defined as a subset of systems described by the LDI of (4.3), where the time variations in the system matrices depend on an exogenous parameter $\delta(t)$ that is assumed to be known and available from online measurements [86, 87]. Consider the LPV system

$$\dot{x}(t) = F(\delta(t))x(t) + B_d(\delta(t))d(t), \quad \zeta(t) = C_\zeta(\delta(t))x(t), \quad x(0) = x_0, \quad \delta(t) \in \Delta. \tag{4.9}$$

Here we further restrict attention to the usual case where the system matrices in (4.9) can be represented by an affine function of the parameters. For example, consider the matrices $F_0, F_1, \ldots, F_m$, $F_i \in \mathbb{R}^{n\times n}, i = 1, \ldots, m$, and $\delta(t) \in \mathbb{R}^m$ such that

$$F(\delta(t)) = F_0 + \delta_1(t)F_1 + \ldots + \delta_m(t)F_m, \quad \delta(t) \in \Delta, \tag{4.10}$$

where the parameters $\delta_i(t)$, $i \in \{1, \ldots, m\}$ take values in some bounded intervals, i.e.

$$\Delta := \{(\delta_1, \ldots, \delta_m) | \delta_i \in [\underline{\delta}_i, \bar{\delta}_i],\ i = 1, \ldots, m\}. \tag{4.11}$$

With no restrictions on the parameters $\delta_i$ other than the finite upper and lower bounds,

the set $\Delta$ is a bounded (convex) polytope with the set of its vertices defined (by simple vertex enumeration) as:

$$\Delta_0 := \{(\delta_1, \ldots, \delta_m) | \delta_i \in \{\underline{\delta}_i, \bar{\delta}_i\},\ i = 1, \ldots, m\}. \tag{4.12}$$

By the Main Theorem of Polytope Theory [88, p. 356], it simply follows that $\Delta$ is a convex hull of its vertices, i.e. $\Delta = \text{conv}(\Delta_0)$. Note that the finite set $\Delta_0$ now has (up to) $2^m$ elements; "up to" because, in some fortunate cases, a number of the vertices would be strictly in the interior of $\Delta$ and can be dropped reducing the number of elements of $\Delta_0$.

For our purposes, the most useful feature of this representation is the fact that $F(\delta(t))$ and the LMIs (4.4) and (4.8) are affine (and therefore convex) functions of the parameters $\delta$. By Proposition 4.1, it follows that it is necessary and sufficient to check the stability and performance guarantees (4.4) and (4.8), respectively, only at the vertices. In fact, the quadratic stability and performance analysis of PLDI systems in [84] make use of this property exactly. Moreover, if the system matrices $F(\delta(t)), B(\delta(t)), C(\delta(t))$ in (4.9) are affine in the parameters $\delta \in \Delta$ in (4.11), then an equivalent PLDI representation of the system is:

$$\begin{aligned} &\dot{x}(t) = F(t)x(t) + B_d(t)d(t), \quad \zeta(t) = C_\zeta(t)x(t), \\ &\quad x(0) = x_0, \quad \begin{pmatrix} F(t) & B_d(t) & C_\zeta(t) \end{pmatrix} \in \Omega \quad \forall t, \\ &\quad \Omega = \text{conv}\{\begin{pmatrix} F(\delta) & B_d(\delta) & C_\zeta(\delta) \end{pmatrix} | \delta \in \Delta_0,\ \Delta_0 \text{ is as defined in } (4.12)\}. \end{aligned} \tag{4.13}$$

This follows simply from the fact that affine functions are both convex and concave and as a result $\Gamma(\text{conv}(\Delta_0)) = \text{conv}(\Gamma(\Delta_0))$ for any affine operator $\Gamma$ [89, p. 67]. In the next section, we make use of this LPV or its equivalent PLDI representation to synthesise bilinear $H_\infty$ filters.

### 4.3.2 Conservativeness of the PLDI representation

It is usually claimed that design or analysis based on a PLDI representation is too conservative since this convex hull would overbound the LPV system in question. Often, the time-varying parameters in an LPV have a finite rate of change and seeking a common Lyapunov function (CLF) that stabilises a convex hull of the polytopic LPV can be conservative; such a CLF based analysis aims to show stability for parameters that can change arbitrarily fast. As a result a wealth of literature does now make use of parameter dependent Lyapunov functions — the less stringent "affine quadratic stability" from [90] are, for example, applied in the filter design with input-rate constraints of [80]. Formu-

lating the quadratic Lyapunov function as a function of the inputs, bounds on both the input and its derivative are used by [80] to find parameter dependent Lyapunov matrices that guarantee stability of the error dynamics.

However, in our motivating wave energy problem, we mainly consider optimally controlled systems with bang-bang control arcs. When modelling the bilinear system using an LPV formulation, this amounts to parameters that switch arbitrarily fast, taking the system from one boundary to another (i.e between vertices of the PLDI). Therefore, considering stability with a CLF is necessary to guard against arbitrarily fast input variations; our approach in this case is, therefore, not a conservative one when compared to affine quadratic stability methods that make use of rate bounds on parameters.

## 4.4 Lower-order $H_\infty$ Filter for Bilinear Systems

Consider the following (lower-order) Luenberger-type bilinear observer for the system of (4.1):

$$\begin{aligned} \dot{x}_f(t) &= \tilde{A}x_f(t) + \sum_{i=1}^{m} u_i\tilde{N}_i x_f(t) + \tilde{B}u^l(t) + G(y - \tilde{C}x_f), \\ \hat{y}(t) &= \tilde{C}x_f(t), \\ \hat{z}(t) &= \tilde{L}x_f(t), \quad x_f(t_0) = 0, \end{aligned} \tag{4.14}$$

where $\hat{z} \in \mathbb{R}^r$ is the estimate of $z$. The state of the observer is $x_f \in \mathbb{R}^{\tilde{n}}$ and $r \leq \tilde{n} \leq n$, $\hat{y}(t) \in \mathbb{R}^p$ approximates the measured output $y$ and $u \in \mathbb{U}$ and $u^l$ are the control inputs as defined for (4.1). Here we take the low-order system matrices $\left(\tilde{A}, \tilde{N}_i, \tilde{B}, \tilde{C}, \tilde{L}\right)$ as given and $G$ is to be determined in the filter synthesis. As in the dual control design problems, what is often done is to extract a low order approximation of the system using model reduction [91, 92] or via successively finer spatial discretization of systems described by PDEs [82]. Using these low-order models, [82] designs low-order controllers with performance guarantees around the full-order model. Similarly, we base our filter design on such low-order models and with closed-loop $L_2$ error attenuation guarantees around the full-order model.

Note that posing all the observer system matrices $\left(\tilde{A}, \tilde{N}_i, \tilde{B}, \tilde{C}, \tilde{L}\right)$ as unknowns results in a nonlinear matrix inequality problem with trilinear monomials. Even arbitrarily setting these nonlinear terms to zero as in [80, 81], it is not clear how to solve this problem for a general low-order model with $\tilde{n} < n$. Even for a linear autonomous system with disturbance inputs only, posing the observer dynamics as unknowns results in difficult non-convex conditions that have proved difficult to be reduced to an equivalent LMI by the projection lemma or congruence transformations [93].

Let $\eta(t) := [x(t)^T \; x_f(t)^T]^T \in \mathbb{R}^{n+\tilde{n}}$ and $d(t) \in \mathbb{R}^q$ denote the extended states and disturbance signals, respectively. The augmented/extended closed-loop system from inputs $u^l(t)$ and $d(t)$ to the estimation error $e(t) := z(t) - \hat{z}(t)$ is then given by:

$$\begin{aligned} \dot{\eta}(t) &= \bar{A}(t)\eta(t) + \bar{B}_1 d(t) + \bar{B}_2 u^l(t), \\ e(t) &= \bar{C}\eta(t), \quad x(t_0) = x_0,\, x_f(t0) = 0 \end{aligned} \tag{4.15}$$

where

$$\bar{A}(t) := \begin{bmatrix} A + \sum_{i=1}^{m} N_i u_i(t) & 0 \\ GC & \tilde{A} + \sum_{i=1}^{m} \tilde{N}_i u_i(t) - G\tilde{C} \end{bmatrix},$$

$$\bar{B}_1 := \begin{bmatrix} R & 0 \\ 0 & GD \end{bmatrix}, \; \bar{B}_2 := \begin{bmatrix} B \\ \tilde{B} \end{bmatrix}, \text{ and } \bar{C} := \begin{bmatrix} L & -\tilde{L} \end{bmatrix}.$$

**Problem 4.1.** *The $H_\infty$ filtering problem is to design a gain $G$ such that, for the observer of the form* (4.14) *we have:*

*(I) error dynamics* (4.15) *that are quadratically stable for all $u(t) \in \mathbb{U}$, when $[d^T \; (u^l)^T]^T = 0$,*

*(II) given $\gamma > 0$, the $L_2$ gain from the disturbance and control signals $[d^T \; (u^l)^T]^T$ to the estimation error $e$ is less than $\gamma$. In other words, we want to bound by $\gamma$ the $L_2$ gain from the signal $[d^T \; (u^l)^T]^T$ to $e$ as in Definition 4.2.*

To solve this problem, we will use the following Lemma.

**Lemma 4.1.** *For $u(t) \in \mathbb{U}$, the error dynamics* (4.15) *can be represented by the equivalent polytypic linear differential inclusion (PLDI):*

$$\begin{aligned} \dot{\eta}(t) &\in \bar{A}(t)\eta(t) + \bar{B}_1 d(t) + \bar{B}_2 u^l(t), \\ e(t) &= \bar{C}\eta(t), \end{aligned} \tag{4.16}$$

*where*

$$\bar{A}(t) \in \bar{\mathcal{A}}(G) := \text{conv}\{\hat{A}(G) | \hat{A}(G) = \hat{A}_0(G) + \alpha_1 \hat{A}_1 + \ldots + \alpha_m \hat{A}_m\},$$

$$\hat{A}_0(G) := \begin{bmatrix} A & 0 \\ GC & \tilde{A} - G\tilde{C} \end{bmatrix}, \; \hat{A}_i := \begin{bmatrix} N_i & 0 \\ 0 & \tilde{N}_i \end{bmatrix}, \quad \alpha_i \in \{-1, 1\}, i = 1, \ldots, m. \tag{4.17}$$

*Proof.* See Section 4.3.1 or [91]. □

A consequence of Lemma 4.1 is that the quadratic stability of (4.15) is equivalent to the

quadratic stability of the PLDI (4.16)–(4.17), which can be easily verified using LMI tools.

**Lemma 4.2.** *For a given $\gamma > 0$, if there exists a gain $G$ and $X = X^T > 0$ such that*

$$\begin{bmatrix} \bar{A}_i(G)^T X + X\bar{A}_i(G) & X\bar{B} & \bar{C}^T \\ \bar{B}^T X & -\gamma I & 0 \\ \bar{C} & 0 & -\gamma I \end{bmatrix} < 0, i = 1, \ldots, 2^m. \tag{4.18}$$

*where $\bar{A}_i(G)$ is the $i^{th}$ element of the $2^m$-tuple* [1] *$\left(\hat{A}_0(G) \pm \hat{A}_1 \pm \ldots \pm \hat{A}_m\right)$ with $\hat{A}_0(G)$, $\hat{A}_1, \ldots, \hat{A}_m$ as defined in* (4.17)*, and with the matrices $\bar{B} := [\bar{B}_1\ \bar{B}_2]$ with $\bar{B}_1$, $\bar{B}_2$ $\bar{C}$ as given in* (4.16)*,*

*then the observer in* (4.14) *solves Problem 4.1.*

*Proof.* By Lemma 4.1, the bilinear error dynamics (4.15) is equivalently represented by the PLDI of (4.16)–(4.17). By the strict bounded real lemma [72, p. 256], for a given $G$, (I) and (II) in Problem 4.1 are true for the PLDI of (4.16)–(4.17) if (4.18) is satisfied. Note that the matrix condition of (4.18) is another form of the one given in (4.8) of Proposition 4.3; the Schur complement lemma is used to express it as a linear function of $\gamma$ and the matrix $\bar{C}$. □

Since we have a matrix inequality feasibility condition for the existence of an appropriate filter gain, we pose an optimization problem to find such a gain $G$ and minimise the bound on the $L_2$ gain $\gamma$ at the same time.

$$\begin{aligned} &\min_{X,G,\gamma}\ \gamma \\ &s.t. \\ &X = X^T > 0, \\ &\gamma > 0, \\ &\begin{bmatrix} \bar{A}_i(G)^T X + X\bar{A}_i(G) & X\bar{B} & \bar{C}^T \\ \bar{B}^T X & -\gamma I & 0 \\ \bar{C} & 0 & -\gamma I \end{bmatrix} < 0, i = 1, \ldots, 2^m. \end{aligned} \tag{4.19}$$

Unlike in the case of $H_2$ filters, the nominal $H_\infty$ filter has been shown to have inherent

[1] An $n$-tuple is an (ordered) set of $n$ elements. For example, the set $S := (\pm p_1 \pm p_2 \pm \ldots \pm p_n)$ with a lexicographic order, starting with $s_1 = -p_1 - p_2 - p_3 - \ldots - p_{n-1} - p_n$ and ending with $s_{2^n} = -s_1$ has $2^n$ elements.

robustness against parametric variations in the system model. However, performance will not be guaranteed under uncertainties not explicitly considered in synthesis [74]. Therefore, it is important to remark that any parametric uncertainties in the bilinear system to be estimated can also be incorporated into the PDLI representation, possibly resulting in more vertices. As such, the lower-order $H_\infty$ filter designed by solving the above optimization problem can be made robust to such uncertainties in a straightforward manner; polytopic uncertainties of the type given in [81] are easily treated by the analysis and synthesis methods given here.

Another point to note is that, although the optimization problem in (4.19) is linear in the unknowns $X$, $G$ and $\gamma$ separately, it is not jointly linear in $X$ and $G$. If we partition $X$ according to the size of $A$ and $\tilde{A}$ as:

$$X := \begin{bmatrix} X_{11} & X_{12} \\ X_{12}^T & X_{22} \end{bmatrix}, \tag{4.20}$$

where $X_{11} \in \mathbb{R}^{n\times n}$ and $X_{22} \in \mathbb{R}^{\tilde{n}\times\tilde{n}}$, it is clear that the problem is not an LMI in the design parameters $X$ and $G$ together — we have the complicating terms $X_{22}G$ and $X_{12}G$, which result in a non-convex bilinear matrix inequality (BMI).

## 4.5 Solving the BMI problem (4.19)

Although LMIs are convex and can be solved efficiently using either interior-point methods or SDP solvers, optimisation problems involving BMIs have been shown to be NP-hard [94]. Unfortunately, a number of control problems (and their dual observer design problems) end up with BMIs; see [95] for many classic robust and optimal control problems with BMIs and for which a convex LMI formulation has not yet been found. Although a number of schemes have been suggested for various specific structures, no method is effective at solving general BMIs. The global branch-and-bound based methods [96] have combinatorial complexity and are, therefore, computationally feasible only for very small (academic) problems. The sum-of-squares based and 'asymptotically global' solvers in [97, 98] are less demanding (the size of successive LMI relaxations grow as the square of the problem state size) but have only been shown to be useful in control problems where only a few complicating bilinear variables are present [99, 100]. Examples include the static and PID controllers parameterised with up to three scalars and with $H_\infty$ performance guarantees in [97].

The only publicly available and benchmark-tested general software for solving BMI inequality constraints is the commercial software PENBMI [101]. This software is based on an augmented Lagrangian formulation for

general nolinear semidefinite programming (PENNON [102]) specialised to BMI constrained problems. For theoretical exposition guaranteeing convergence to local optimal points and algorithm implementation details, see [102, 103] and references therein. Although this method fails for some large order test cases and/or ill-conditioned problems, it has been shown [103] to return a local minimum for most of the $H_2$ and $H_\infty$ benchmark nonlinear SDP and BMI problems in $Compl_eib$ : COnstrained Matrix Optimization Problem library [104]. We had attempted to use PENBMI for observer design for our wave energy problem with no success, since it struggled to solve the problem from feasible initial conditions.

We remark that any nonlinear program solver, including PENBMI, can guarantee only a local solution, since the problem is nonconvex. Moreover, the solution very much depends on the initial condition. In addition to using nonlinear program solvers to find local solutions using various methods, a solution is often sought by solving a convex relaxation of the BMI or by taking advantage of a structure in the specific problem. For example, a typical relaxation is to restrict the space of Lyapunov matrices so that the problem becomes amenable to a congruence transformation or a change of variables that result in convex (or LMI) problems [105]. A simple relaxation for problem (4.19) is to restrict the Lyapunov functions considered in (4.20) by assuming $X_{12} = 0$. Then, the BMI can be changed into an LMI with a simple change of variables $Y = X_{22}G$. However, preliminary simulations show this method can sometimes be infeasible even for the case where the optimization problem has a solution. Therefore, we seek a different alternative.

As often done in control design problems with BMI constraints (see [100, 106] and references therein) we pursue an LMI-based coordinate descent scheme. This will allow one to solve BMI problems using one of a number of open-source LMI solvers (see [107] for a list) where a BMI solver may fail or is not accessible . By fixing one of the complicating BMI variables, $G$ or $X$ in (4.19), we solve a set of LMIs alternatively from a feasible initial solution. These alternate LMIs have solutions in a subset of the BMI problem solutions; so, if we start from a feasible solution, this method remains feasible and is monotonically non-increasing. As detailed in [100], optimality of the alternating LMIs does not imply optimality in both variables — the method can get stuck before reaching a local minima. However, although not theoretically proved due to the nonconvexity of constraint sets, local optimal solutions are in practice reached if each coordinate descent leads to a decrease in the objective function [108]. In our implementation, part of the Lyapunov-related matrices can also be simultaneously optimized with the feedback gain $G$ so as to aid an improvement in each iteration, following the suggestion in [108].

**Algorithm 2** Lower-order $H_\infty$ filter synthesis for bilinear systems

Fix some algorithm termination parameters $\varepsilon$, and $max_{iter}$. $\varepsilon$ is sufficiently small to indicate convergence of successive disturbance attennuation bounds $\gamma^{(k)}$. We carry out a maximum of $max_{iter}$ iterations.

**Step 0:** Synthesize a quadratically stabilizing $H_\infty$ filter for the lower-order model by solving an LMI problem; this gives some gain $G^{(0)}$ that is quadratically stabilizing for the PLDI in Lemma 4.2. Set $k = 1$.

**Step 1:** Fix $G$ at $G^{(k-1)}$ and solve (4.19) (i.e. an LMI) to find new $\gamma^{(k)}$ and $X^{(k)}$.
**Step 2:** Fix $X$ (or only the $X_{12}$ and $X_{22}$ components of $X$) at $X^{(k)}$ and solve (4.19) (i.e. an LMI) to find new $\gamma^{(k)}$ and $G^{(k)}$ (or new $\gamma^{(k)}$, $X_{11}^{(k)}$ and $G^{(k)}$).
**Step 3:** If $\frac{|\gamma^{(k)}-\gamma^{(k-1)}|}{|\gamma^{(k-1)}|} \leq \varepsilon$ OR $k+1 > max_{iter}$, then stop, else set $k \leftarrow k+1$ and return to **Step 1**.

### 4.5.1 LMI-based algorithm and initial robustly feasible observer

From (4.16)–(4.17) and the assumption that the full-order model is quadratically stable, it can be shown that the estimation error dynamics are quadratically stable if $G$ stabilizes the lower-order $2^m$–tuple PLDI $\left(\tilde{A} \pm \tilde{N}_1 \pm \ldots \pm \tilde{N}_m\right), i = 1, \ldots, m$. Therefore, we can find a feasible filter by solving the standard convex full-order $H_\infty$ problem around the lower-order model. Although this filter may not have optimal performance around the full-order model, it will be quadratically stabilising for the error dynamics (4.15) and therefore would suffice as an initialization for Algorithm 2. Using the solution as an initial condition for $G$, we can alternately solve LMIs until the noise attenuation bound $\gamma_k$ no longer improves. It is important to note here that we are not discussing stability of a closed-loop control system (eg. the receding horizon model predictive control implemented in Chapter 3), but merely stability of the error dynamics as an input-output block. For comments on closed-loop stability of our receding horizon control, see Section 3.3.6.

### 4.5.2 An upper bound on the optimality gap for Algorithm 2

A very desirable property for an asymptotically global method or solutions via convex relaxations is being able to either certify global optimality or computing the global optimality gap [98]. The method of Algorithm 2 is only a local method and so the globaly optimal $L_2$ gain bound $\gamma^*$ is unknown for the $\tilde{n}^{th}$ order filter, where $\tilde{n} < n$ and the problem is a nonconvex BMI. However, a full-order filter (i.e. a filter with $\tilde{n} = n$ based on exactly the full-order model) can be expected to give the the best performing filter around the full-order model. Luckily, the full-order $H_\infty$ synthesis problem is a convex problem and can be solved via the bounded real-lemma LMI.

We consider the same Luenberger-type bilinear observer as before but a full-order one:

$$\begin{aligned} \dot{\hat{x}}(t) &= A\hat{x}(t) + \sum_{i=1}^{m} N_i \hat{x}(t) u_i(t) + Bu^l(t) + G(y(t) - C\hat{x}(t)), \\ \hat{z}(t) &= L\hat{x}(t), \quad x(t_0) = x_0, \end{aligned} \tag{4.21}$$

where $\hat{x}(t) \in \mathbb{R}^n$ and $u(t) \in \mathbb{U}$.

In another form, substituting for $y(t) = Cx(t) + Dv(t)$, (4.21) becomes

$$\begin{aligned} \dot{\hat{x}}(t) &= A\hat{x}(t) + \sum_{i=1}^{m} N_i \hat{x}(t) u_i(t) + GCx(t) - GC\hat{x}(t) + Bu^l(t) + GDv(t), \\ \hat{z}(t) &= L\hat{x}(t), \quad x(t_0) = x_0, \end{aligned} \tag{4.22}$$

where $\hat{x}(t) \in \mathbb{R}^n$ and $u(t) \in \mathbb{U}$.

Let $\xi(t) := x(t) - \hat{x}(t)$ and $e(t) := z(t) - \hat{z}(t)$. Then the error dynamics can be written as:

$$\begin{aligned} \dot{\xi}(t) &= \left( A + \sum_{i=1}^{m} N_i u_i(t) - GC \right) \xi(t) + \begin{bmatrix} R & -GD \end{bmatrix} \begin{bmatrix} w(t) \\ v(t) \end{bmatrix}, \\ e(t) &= L\xi(t), \quad \xi(t_0) = \xi_0, \end{aligned} \tag{4.23}$$

The design objective is to find $G$ such that $A + \sum_{i=1}^{m} A_i u_i(t) - GC$ is quadratically stable $\forall u(t) \in \mathbb{U}$ and the $L_2$ gain from $[w\ v]^T$ to $e$ is minimized. Writing the PLDI that includes (4.23) as before, we can use the following lemma.

**Lemma 4.3.** *For a given $\gamma > 0$, if there exists a gain $Y$ and $X = X^T > 0$ such that for $i = 1, \ldots, 2^m$*

$$\begin{bmatrix} \bar{A}_i^T X + X\bar{A}_i - C^T Y - Y^T C & [XR \quad -YD] & L^T \\ [XR \quad -YD]^T & -\gamma I & 0 \\ L & 0 & -\gamma I \end{bmatrix} < 0, \tag{4.24}$$

*where $\bar{A}_i$ is the $i^{th}$ element of the $2^m$-tuple $(A \pm N_1 \pm \ldots \pm N_m)$ and $(A, N_1, \ldots, N_m)$, $R$, $D$, $L$ are as in* (4.23),

*then the observer in* (4.21) *has quadratically stable error dynamics* (4.23) *and the $L_2$ gain from $[w^T\ v^T]^T$ to $e$ is less than $\gamma$.*

*Proof.* The proof follows from the strict bounded real lemma [72, p. 256] if the substitution $Y = XG$ is made. Here, $X = X^T \in \mathbb{R}^{n \times n}$ and $Y \in \mathbb{R}^{n \times p}$. □

For the full-order $H_\infty$ filter above, an LMI feasibility problem is derived using a change

of variables. This resulting problem is convex since it is linear in all the unknowns $X$, $Y$ and $\gamma$. Minimizing this linear problem with a convex LMI constraint will give us the lowest achievable quadratic $L_2$-gain performance for the PLDI; because the Lyapunov matrix $X$ is positive definite, the gain can be derived from the inverse relation $G = X^{-1}Y$. Let this optimal gain be $\gamma_n^*$, with $n$ denoting the fact that the filter is full-order. This can be used to upper bound the distance between the performance $\gamma_{\tilde{n}}^{(k)}$ of the order $\tilde{n}$ filter from the $k^{\text{th}}$ iterate of Algorithm 2 and the global solution for the optimization problem in (4.19), $\gamma_{\tilde{n}}^*$:

$$(\gamma_{\tilde{n}}^{(k)} - \gamma_{\tilde{n}}^*) \leq (\gamma_{\tilde{n}}^{(k)} - \gamma_n^*). \tag{4.25}$$

Where the relative optimality gap bound $\frac{\gamma_{\tilde{n}}^{(k)} - \gamma_n^*}{\gamma_n^*}$ is small, the solution from Algorithm 2 will be close, in performance, to the optimal order $\tilde{n}$ bilinear model-based $H_\infty$ filter.

## 4.6 Continuous-time Extended Kalman Filter

The Extend Kalman filter (EKF) is by far the most widely and successfully used filter for state and parameter estimation of nonlinear systems [109]. As an extension of the well known minimum error variance Kalman filter for linear systems, the EKF and its higher order versions can estimate the noise covariances and give us locally optimal estimates for nonlinear systems. Nonetheless, the EKF has some associated drawbacks. Firstly, the stability of the error dynamics cannot be guaranteed for a general nonlinear system. For systems with bounded nonlinearities, like the ones discussed in this chapter and some of the literature we have reviewed, the continuous-time EKF can guarantee stability with some modifications such as [109, 110]. The other disadvantage is the computational burden imposed in the integration of the noise covariance matrices.

Given the system and measurement noises in (4.1), we re-write the bilinear system dynamics in a general nonlinear system form [109, p. 401]:

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t), w(t), t), \quad w \sim N(0, Q_w), \\ y(t) &= h(x(t), v(t), t), \quad v \sim N(0, R_v), \\ z(t) &= g(x(t), t) \end{aligned} \tag{4.26}$$

where $w$ and $v$ are the system and measurement disturbances, which are assumed to be

white noise with a given covariance. The EKF is then implemented as:

$$\begin{aligned}
\dot{\hat{x}}(t) &= f(\hat{x}(t),u(t),0,t)+K(t)[y(t)-h(\hat{x}(t),0,t)], \quad \hat{x}(0)=\hat{x}_0,\\
K(t) &= P(t)C(t)^T\hat{R}^{-1},\\
\dot{P}(t) &= F(t)P(t)+P(t)F(t)^T-K(t)C(t)P(t)+\hat{Q}, \quad P(0)=P_0,\\
\hat{z}(t) &= g(\hat{x}(t),t),
\end{aligned} \tag{4.27}$$

where $F(t) := \frac{\partial f}{\partial x}|_{(\hat{x}(t),u(t))}, C(t) := \frac{\partial h}{\partial x}|_{(\hat{x}(t),u(t))}, W(t) := \frac{\partial f}{\partial w}|_{(\hat{x}(t),u(t))}, V(t) := \frac{\partial h}{\partial v}|_{(\hat{x}(t),u(t))}$, $\hat{Q} := WQ_wW^T$, $\hat{R} := VR_vV^T$ and $P(t) \in \mathbb{R}^{n\times n}$ is the covariance of the state estimation error $x(t)-\hat{x}(t)$. Note that we integrate an $n^2$-size vector dynamics to find $P(t)$. A lower-order EKF can be implemented as in (4.27); it is based on a lower-order model of the system dynamics (4.26). In the next section, we compare the performance of the low-order model based EKF with the low-order $H_\infty$ filter.

## 4.7 Example Simulations

In this section, we consider the heaving point absorber WEC from Chapter 3 where a power take-off mechanism with controlled damping and active force elements is used. A state space representation was derived where the kernel representing forces due to radiated waves is approximated arbitrarily well by a finite-dimensional state-space model of sufficient order. An optimal control problem was then posed to determine a real-time optimization based control to be used in a receding horizon manner. The active and passive power take-off forces are controlled proportionally. It was shown that, with a radiation subsystem of order $n_{rad} = n-2$ identified, the complete WEC system dynamics can be re-written in state space form as:

$$\begin{aligned}
\dot{x}_1(t) &= x_2,\\
\dot{x}_2(t) &= \tfrac{1}{M+\mu_\infty}[f_{exc}(t)+Gu^l(t)-\\
&\quad B_{pto}u(t)x_2(t)-C_rx_{3:n}(t)-Kx_1(t)],\\
\dot{x}_{3:n}(t) &= A_rx_{3:n}+B_rx_2(t),
\end{aligned} \tag{4.28}$$

where the $u^l(t) \in [-1,1]$ represents the active force ($u_1(t)$ in the notation of (3.7)) and $u(t) \in [0,1]$ controls the damper force ($u_2(t)$ in (3.7)). The state $x := [x_1\ x_2\ \dots\ x_n]^T = [\zeta\ \dot{\zeta}\ z_r^T]^T \in \mathbb{R}^n$ with the appropriate initial conditions was evolved for predicting system behaviour. Of course, (4.28) is a bilinear system of the form considered in (4.1); it has a bilinear term in the form of $u_2(t)x_2(t)$ where $u_2$ is bounded.

As in all the wave energy literature, the radiation force $f_r(t) \approx C_r x_{3:n}(t)$ was assumed known in the dynamic optimal control algorithm of Section 3.3.3; this is not true in practice since we cannot measure this force directly. The measurements of the velocity used in determining the control will also have errors in practice. For the model considered in the example simulations, sufficiently high order systems can approximate the radiation kernel well enough in simulation. Here, we take a $7^{\text{th}}$ order system model (i.e. $n = 7$ and the radiation subsystem is of order 5) as the full-order model and synthesise a robust lower-order observer based on a $3^{\text{rd}}$ order radiation subsystem model ($\tilde{n} = 5$). We consider all the parameters as in Figure 3.8. In addition, we take a nominal value of $B_{pto} = 304$ kNs/m. We then construct a low-order EKF based on this model. We consider it to be the case that the parameter $B_{pto}$ is known with some uncertainty. Here, we construct a low-order $H_\infty$ filter with a parameter uncertainty of $\pm 10\%$ — the true system has $B_{pto} = 275$kNs/m. The parametric uncertainty is explicitly included in the LPV model used in (4.16). The performances of the EKF and $H_\infty$ filters are compared under this parametric uncertainty. We assume that velocity and displacement measurements are available. The objective is to attenuate the $L_2$ gain from the control inputs, excitation force and disturbances to the estimation error. A normally distributed random noise with a bound equal to the mean of the absolute actual displacement was added to the measured signals; $R = 0.1I_n$ and $D = 0.1I_p$. The EKF is based on the actual noise covariances to show how well the low-order $H_\infty$ observer does.

All computations were performed on a 2.4 GHz Intel Core 2 Duo CPU machine in MATLAB using YALMIP [107], and SDPT3 [111] — a Matlab software package for semidefinite programming. The system matrices are too large to display here.

The high order simulations for the WEC are generated by the optimal bang-bang controller as shown in Figure 3.5. The convergence of the closed-loop disturbance attenuation bound $\gamma$ from Algorithm 2 is shown in Figure 4.1; Figure 4.2 shows a noisy measurement of the displacement and velocity. The low order model is based on a $3^{\text{rd}}$ order radiation subsystem approximation. In Figure 4.3a and Figure 4.3b, we show a snippet of the estimates and estimation errors, respectively. The low-order model based $H_\infty$ filter estimates converge to the true value and have average relative absolute errors of less than 10%; the low-order EKF has average relative absolute error of about 35%. The $H_\infty$ filter shows better performance in this sense. Moreover, whereas the real-time implementation of the EKF requires the integration of size $O(\hat{n}^2)$ state dynamics, the low-order $H_\infty$ observer integrates an $\hat{n}$ state dynamics (3 in this example). In addition, it does not require knowledge of the statistics of the noise signals. Figure 4.4 shows how the EKF performance degrades for different non-zero initial state estimates. Simulations from many random initial state estimates confirm the robustness of the $H_\infty$ filter with less than 10% average error; the EKF shows average relative errors

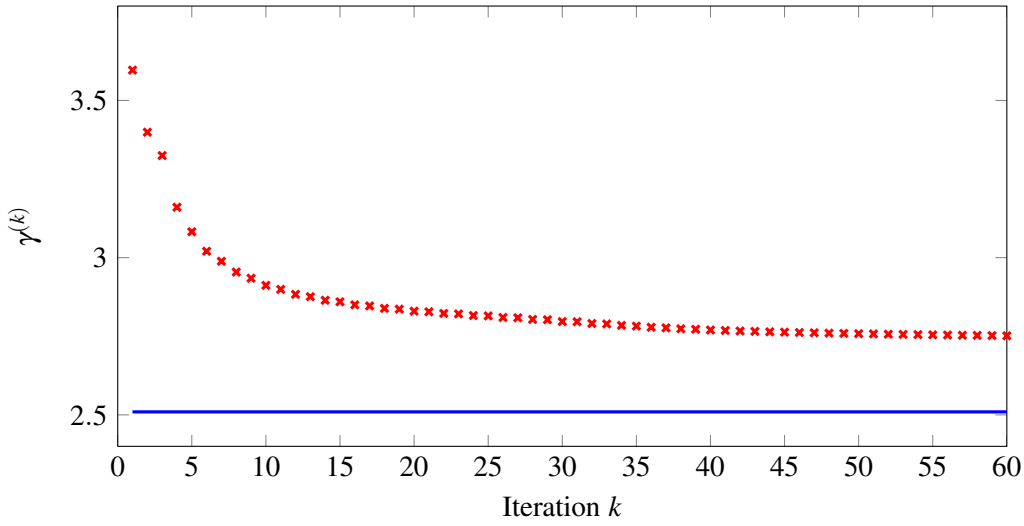



Figure 4.1: Convergence of $\gamma^{(k)}$ in synthesis via Algorithm 1. The lower (blue) line shows the best full-order filter performance as a bound for $\gamma^{(k)}$.

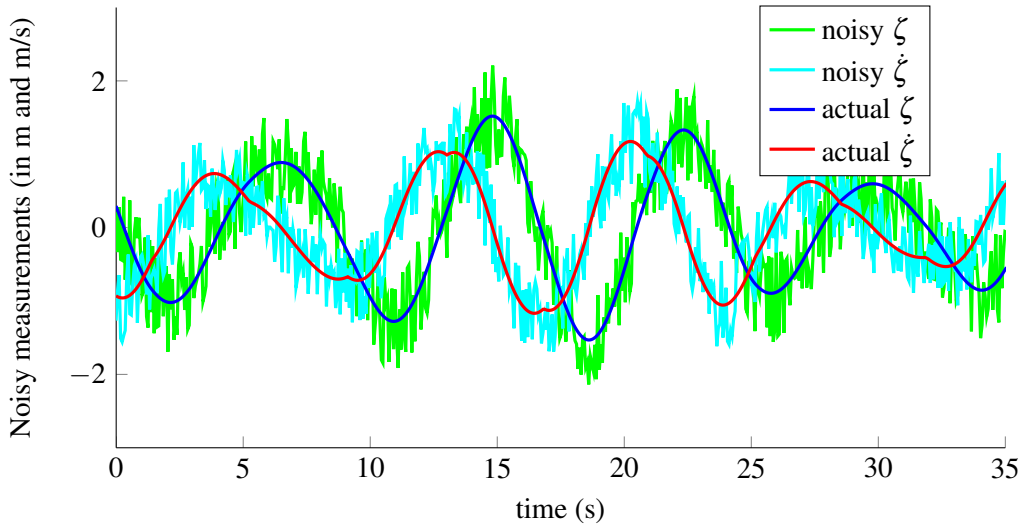
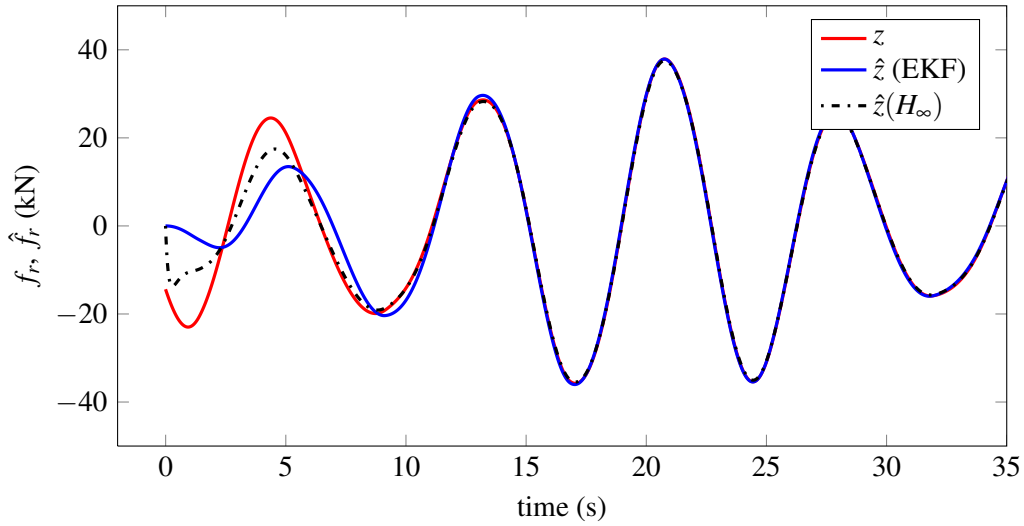


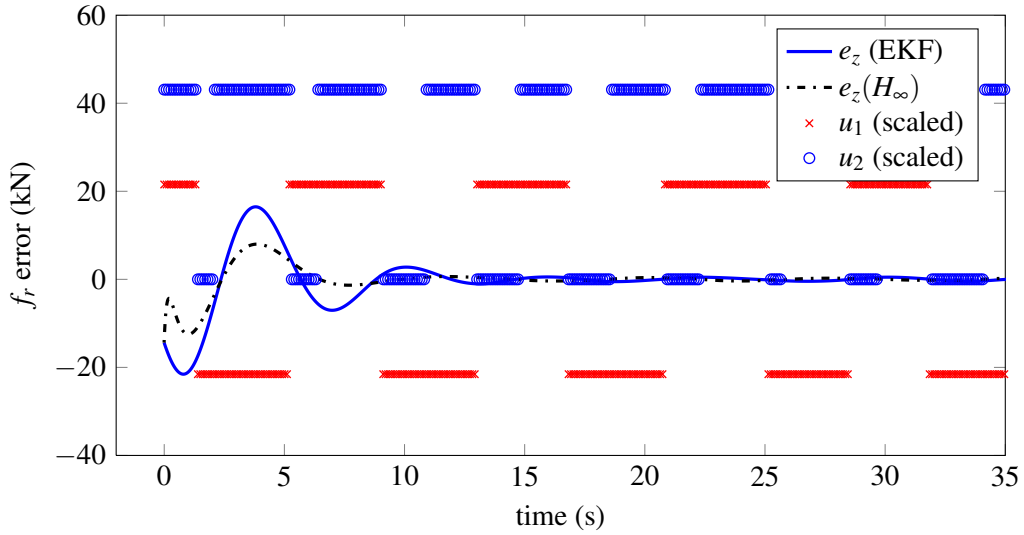

Figure 4.2: Displacement and velocity and their noisy measurements

of more than 100%.

Considering both filtering methods, it was also observed that the estimates deteriorate as the order of the model used in the synthesis decreases. This is not surprising since a lower order model perhaps captures less of the system dynamics. In Figures 4.5 and 4.6, we can see estimates using measurement data of velocity and displacement from a receding horizon simulation. From these and other simulations, it appears that under some initial conditions and when the velocity is high, compared to a 3$^{\text{rd}}$-order $H_\infty$ observer, the 1st-order $H_\infty$ observer and the 3$^{\text{rd}}$-order EKF poorly estimate the radiation force. The latter is also poor at estimating the velocity. It may seem from these that a 1st order radiation subsystem gives a filter with unsatisfactory $L_2$-gain bounds for the error




(a) estimates of the radiation force




(b) errors in estimates, and scaled control inputs

Figure 4.3: Comparison of radiation force estimates by the EKF and robust $H_\infty$ filters; $P(0) = I$, and $\hat{x}(0) = 0$.

dynamics. It also has an error dynamics that converges much slower compared to a 3rd order radiation subsystem based-model.

However, a more useful comparison is to assess the filter performances within a closed-loop framework. In such a setting, a filter may have an error dynamics that performs similarly or only marginally worse on average. For our purposes, the ultimate measure of performance for an observer is how much less energy is delivered by the observer-based controller. Below we consider using observer based estimates of velocity and radiation forces within the optimal control scheme developed in Section 3.3.3.

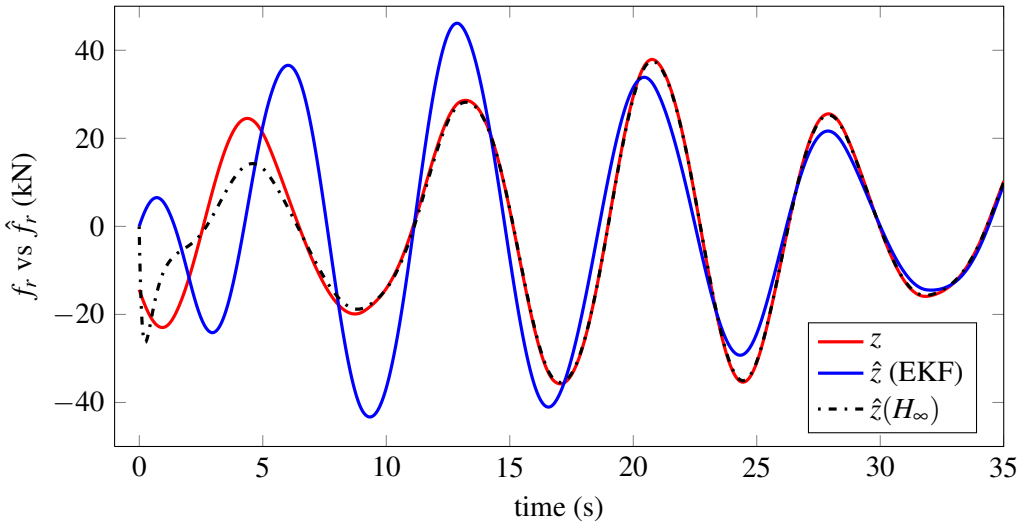



Figure 4.4: Estimates of radiation force with a different initial state guess; $P(0) = I$, and $\hat{x}(0) = [1\ -1\ 0\ 0\ 0]^T$

### 4.7.1 Observer-based MPC and closed-loop receding horizon simulations

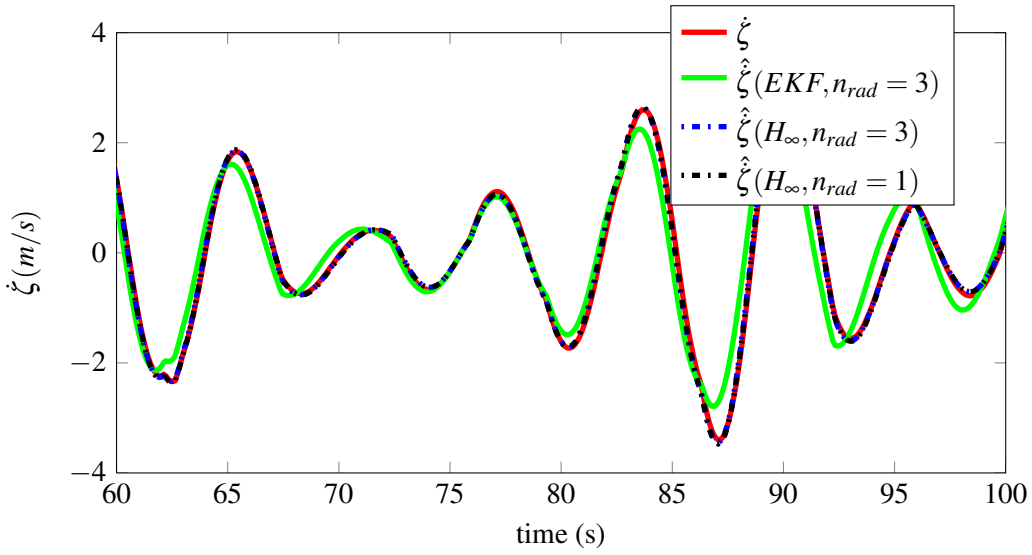
Although we have commented on the performance of the filters in estimating the radiation force and velocity, a more appropriate measure of their performance should consider closed-loop behaviour of the observer-based controller. Here, we consider again the optimal control problem solved at each iteration of the receding horizon control.

#### Optimal Control Design

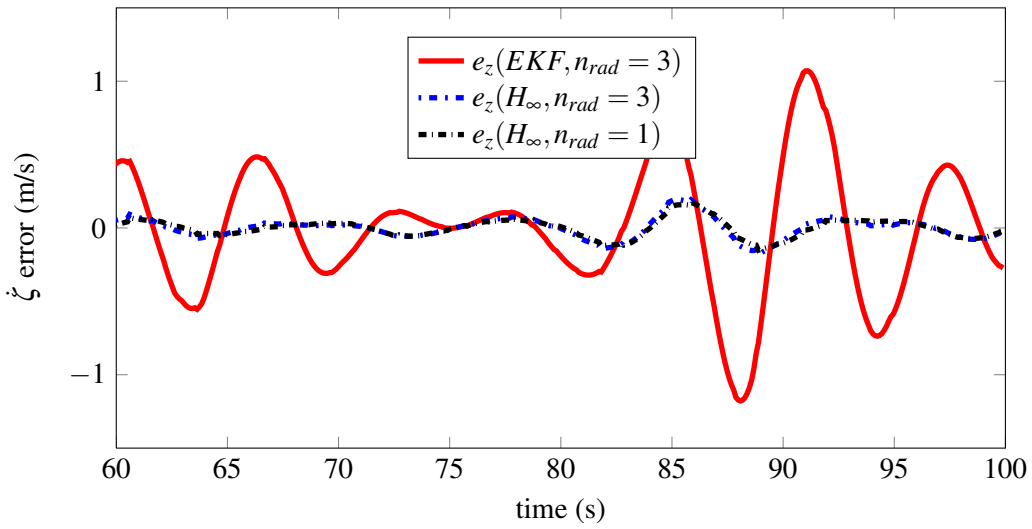
We use the same indirect optimal control method as in Section 3.3.1. The main difference here is that we do not have access to either the radiation force or exact values of the device velocity or displacements. Based on measured outputs, estimates from observers will be used. The Hamiltonian of the optimal control problem can be re-written as:

$$\begin{aligned} H(x,u,\lambda,t) := &- B_{pto}u_2x_2^2 + Gu_1x_2 + \lambda_1x_2 + \frac{\lambda_2}{M+\mu_\infty}\{f_{exc}(t) + \\ &Gu_1 - B_{pto}u_2x_2 - f_r - Cx_1\}. \end{aligned} \tag{4.29}$$

Taking the derivative of the Hamiltonian with respect to the inputs, the switching functions and hence the gradient of the cost can be shown to be the same as in (3.20) and (3.22), respectively. The full-order plant model is no longer used for prediction. Instead, the observer dynamics are evolved to find the estimates $\hat{x}_2$ and $\hat{f}_r$. These esti-




(a) velocity estimates



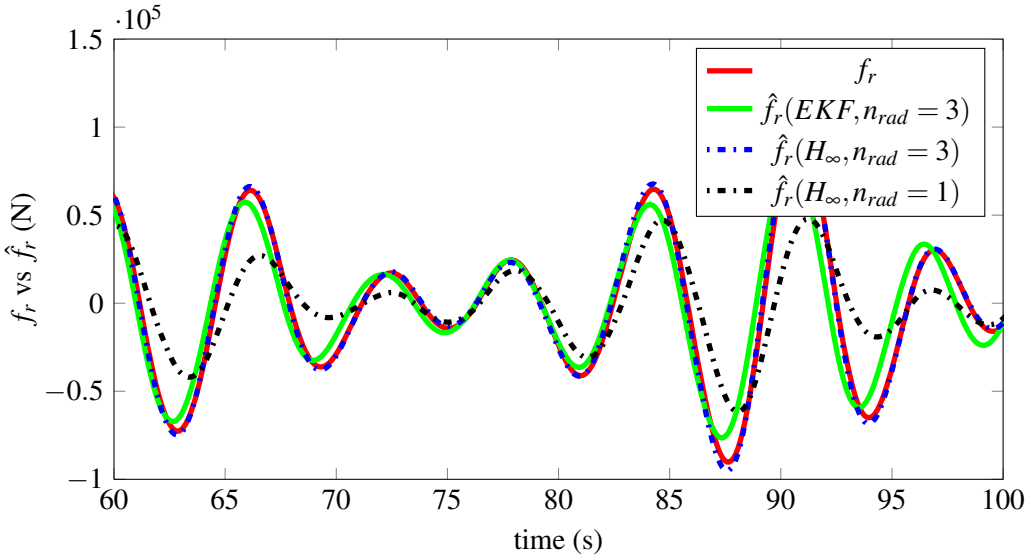

(b) errors in velocity estimates

Figure 4.5: Velocity estimates with observers based on radiation subsystems of order $n_{rad}$ equal to1 and 3 for the $H_\infty$ filter, and an EKF based on a 3rd order radiation subsystem.

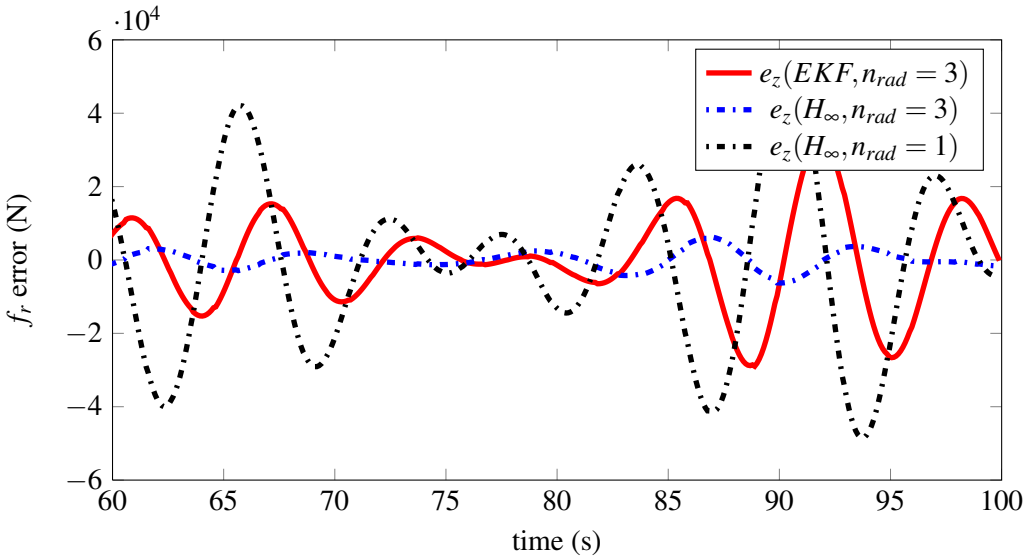
mates are then used in the adjoint dynamics:

$$\begin{aligned} \frac{d\lambda_1}{dt} &= \frac{C\lambda_2}{M+\mu_\infty}, \\ \frac{d\lambda_2}{dt} &= 2B_{pto}u_2x_2 - Gu_1 - \lambda_1 + \frac{B_{pto}\lambda_2 u_2}{M+\mu_\infty}, \end{aligned} \tag{4.30}$$

where $\dot{\lambda}_i(t) = -\frac{\partial H}{\partial x_i}(x(t), u(t), \lambda(t), t)$. The adjoint state $\lambda = [\lambda_1^T \ \lambda_2^T]^T$ has final condition $\lambda(t_f) = 0$, as before, because the terminal cost is zero.




(a) radiation force estimates




(b) errors in radiation force estimates

Figure 4.6: Radiation force estimates with observers based on radiation subsystems of order $n_{rad}$ equal to1 and 3 for the $H_\infty$ filter, and an EKF based on a 3[rd] order radiation subsystem.

In Step 1 of Algorithm 1, rather than evolving the state dynamics of the full-order model, we now **estimate** the observer state $\hat{x}$ over $[t_0, t_f]$ by integrating the low-order observer dynamics with the initial condition $\hat{x}(t_0) = \hat{x}_0$ and no innovation term. Similarly, in Step 2 of Algorithm 1, the second order costate dynamics $\lambda$ is computed by backward integration of (4.30) over $[t_f, t_0]$ using the estimated observer dynamics and the final values $\lambda(t_f) = 0$.

We present sample simulations to show the performance of the observers in the optimal

control scheme of Section 3.3.1. Based on the result in Figure 3.8, a prediction horizon of 12 s with 1 s control interval $T_c$ was used. The effect of model-mismatch on the estimator based receding horizon control is shown, considering both the EKF and $H_\infty$ filters of various orders. Figure 4.7 shows the cumulative energy extracted when various observers are used. Similarly, all these observers were also tested under parameter uncertainty for the damper value $B_{pto}$; the filters are based on a value of $B_{pto} = 304$kNs/m and the true value is now $B_{pto} = 275$kNs/m. Table 4.1 shows the average power extracted under all these situations. The filter order is $n_{rad} + 2$, where $n_{rad}$ is the order of the radiation subsystem used.

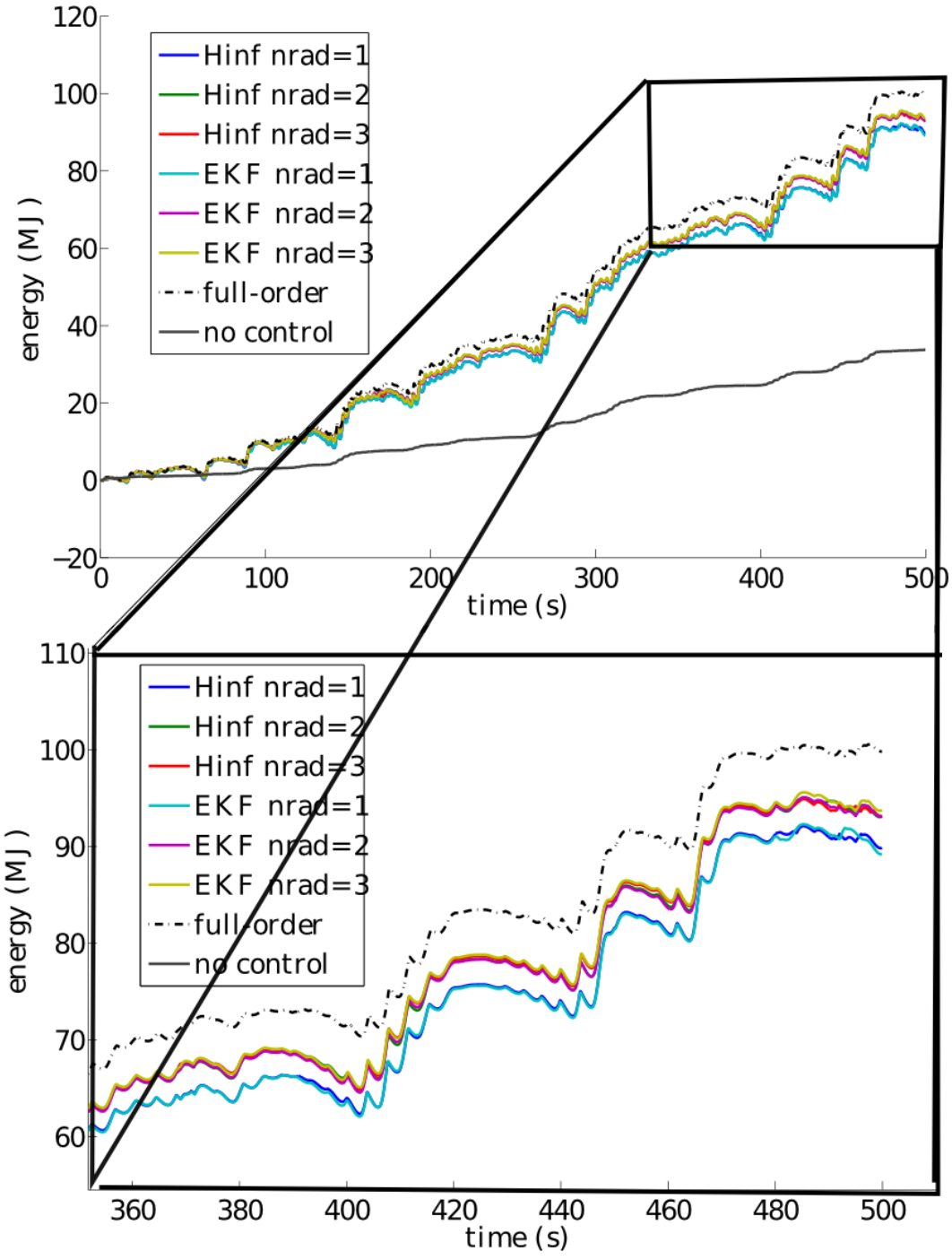
The lowest-order observers show the most suboptimal power. However, the worst performing observer-based controllers deliver only about 9% less power than the full knowledge controllers. Under the given measurement noise, performance does not improve much for filters with $n_{rad}$ higher than 2 or 3. The $H_\infty$ filters generally perform similarly to the same order EKF filters in closed-loop. We can, therefore, conclude that a low-order filter for velocity and radiation force estimation can be effectively used in a receding horizon controller. Such an implementation is shown to deliver about 3-times more power compared to an uncontrolled WEC system; see Figure 4.7. Random noise with a normal distribution and a bound equal to 50% of the maximum of the actual displacement and velocity absolute values was used; the standard deviation for each measurement noise is less than 0.5 times its bound. These bounds are used in the EKF.

Table 4.1: Performance of the filters in average extracted power (kW) under closed-loop simulations. Under parametric uncertainty, the filters are based on a value of $B_{pto} = 304$kNs/m where the true value is now $B_{pto} = 275$kNs/m; the same measurement noise is applied in all cases. The "Full-order" column shows the performance when, rather than an observer, the full-order model with complete state knowledge is used.

| Filter | Nominal Performance (kW) | | | | Under parametric uncertainty (kW) | | | |
|---|---|---|---|---|---|---|---|---|
| $n_{rad}$ | $H_\infty$ | EKF | Full-order | No control | $H_\infty$ | EKF | Full-order | No control |
| 1 | 186.1 | 184.8 | 204.2 | 68.6 | 176.8 | 175.4 | 199.6 | 67.5 |
| 2 | 191.5 | 193.2 | 204.2 | 68.6 | 186.6 | 187.2 | 199.6 | 67.5 |
| 3 | 194.0 | 194.4 | 204.2 | 68.6 | 187.6 | 187.7 | 199.6 | 67.5 |
| 4 | 194.6 | 194.6 | 204.2 | 68.6 | 188.3 | 187.9 | 199.6 | 67.5 |
| 5 | 194.6 | 194.6 | 204.2 | 68.6 | 188.3 | 188.3 | 199.6 | 67.5 |

## 4.8 Conclusion

In this chapter, the low order $H_\infty$ filter design for bilinear systems with bounded inputs has been investigated. Since the robust unbiased functional filters in the literature are




(a)

Figure 4.7: Cumulative extracted energy of observer-based receding horizon control. The order of the radiation subsystem for observer design is *nrad*. $T_p = 8$ s and $H_s = 2$ m used; $G = B = 0.3 * (M + \mu_\infty)$.

either not applicable for some problems, the wave energy example being one, or do not preserve unbiasedness in the presence of uncertainties, we have proposed a new robust low-order filter design method. An LPV formulation of the bilinear system has been used to design a filter that is based on a lower order model of the plant, but with closed-loop robust disturbance attenuation guarantees around the full-order model. Although the optimal filter synthesis is a (non-convex) BMI problem, we have used an efficient LMI-based coordinate descent algorithm to find solutions locally. Example simulations in dynamic radiation force estimation of a wave energy converter have been used to show robustness of the $H_\infty$ filter under parametric and other model uncertainties by comparing the performance with an extended Kalman filter. In addition to to alleviating the need to know the spectrum of the noise covariances by the EKF, the $H_\infty$ filter is robust to model uncertainties.

We have also investigated the robustness of the filters when used within a receding horizon control of a heaving buoy. It was shown that performance, in terms of power extracted, is only marginally lower when using these observers compared to a full-knowledge MPC controller. We have, therefore, presented a viable bilinear observer-based control scheme for a heaving WEC.

# Chapter 5

# Matrix Exponential Methods for Integration of Bilinear Control Systems

In Chapter 3, the use of an indirect method for the computation of an optimal controller was proposed. The main computational cost of the projected gradient method in Algorithm 1 is the forward integration of the system dynamics and the integration of the adjoint dynamics backwards. For this reason, the computational efficiency of relevant ODE solvers for bilinear systems was investigated.

Based on the variation of constants formula for linear systems, we show that bilinear control systems with zero-order hold on the input can be integrated to a high prescribed precision by computing the action of the matrix exponential on a vector. We make use of advanced new results in the literature for efficient computation of the latter with some a priori guarantees on the error tolerance. In Section 5.3.3, we derive new a priori bounds on the computational complexity of integrating bilinear systems with a given backward error as a function of the problem data. We also investigate the use of Krylov subspace methods for computing the action of a matrix exponential on a vector.

In Section 5.4, using our new bounds on computational complexity, we propose a direct exponential integrator to solve bilinear ODEs via the solution of a sparse linear system. We also analyse the sparsity and computational complexity of solving the resulting linear system. It is then compared with a similarly implemented sparse fourth-order explicit Runge-Kutta scheme. Numerical experiments are also used to assess the advantages of the exponential integrators compared with the classical Runge-Kutta method. In addition to the wave energy system, we use a PDE heat transfer model for the controlled cooling of a metal slab.

In Section 5.6, the exponential integrator with our new bounds is used in a novel sparse direct collocation of bilinear systems for optimal control. Example academic quadratic problems with bilinear dynamics are used to demonstrate the feasibility of the proposed approach.

## 5.1 Exponential Integrators: Avoiding $\varphi$ Functions

The work in [112] reviews exponential integrators for some ODEs, namely methods of integration for solving initial value problems of the form

$$\dot{x}(t) = Lx(t) + g(t, x(t), p(t)),\ x(t_0) = x_0,\ t \geq t_0, \tag{5.1}$$

where $x(t) \in \mathbb{R}^n$, $L \in \mathbb{R}^{n \times n}$, and $g(\cdot)$ is a nonlinear function with possibly time varying parameters $p(t)$. The solution of (5.1) satisfies the variation of constants formula

$$x(t) = e^{(t-t_0)L}x_0 + \int_{t_0}^{t} e^{(t-\tau)L} g(\tau, x(\tau), p(\tau))d\tau,\ \forall t \geq t_0. \tag{5.2}$$

An equivalent representation for (5.2) is [112, Lem. 5.1]:

$$x(t) = e^{(t-t_0)L}x_0 + \sum_{l=1}^{\infty} (t-t_0)^l \varphi_l((t-t_0)L) g_{t_0}^{(l-1)},\ \forall t \geq t_0, \tag{5.3}$$

where $g_{t_0}^{(l-1)} = \frac{d^{l-1}}{dt^{l-1}} g(t, x(t), p(t))|_{t=t_0}$ and the so called $\varphi$-functions satisfy the recursive equations

$$\varphi_{l+1}(z) = \frac{\varphi_l(z) - \frac{1}{l!}}{z},\ l = 0, 1, \ldots,\ \varphi_0(z) = e^z. \tag{5.4}$$

The proof of [112, Lem. 5.1] is based on a Taylor series expansion of $g(\cdot)$ about $t_0$. In computer simulations, some finite $p^{\text{th}}$ order truncation of (5.3) is often sought to approximate the solution as

$$x(t) \approx \hat{x}(t) = e^{(t-t_0)L}x_0 + \sum_{l=1}^{p} (t-t_0)^l \varphi_l((t-t_0)L) g_{t_0}^{(l-1)},\ \forall t \geq t_0. \tag{5.5}$$

In practice, only orders of only up to $p = 4$ are considered [112].

A result of great interest to us is to compute (5.5) without explicitly computing the $\varphi$-functions [113, Thm. 2.1]; a method to compute the action of the $\varphi$-functions on a vector with only a marginal increase in computational cost is proposed in [113, Thm. 2.1]— an expression for the right hand side of (5.5) is derived involving only a single

matrix exponential. Let $L \in \mathbb{R}^{n\times n}$, the matrix exponential of $L$ is given by

$$e^L = I_n + L + \frac{L^2}{2!} + \frac{L^3}{3!} + \dots. \tag{5.6}$$

where $I_n$ is the identity matrix of size $n$.

Now, let $W = [w_p,\, w_{p-1},\, \dots, w_1] \in \mathbb{R}^{n\times p}$ where the stacked vectors $w_k$ are derivatives of the nonlinear term; i.e. $g_{t_0}^{(k-1)} =: w_k \in \mathbb{R}^n$, $k = 1, \dots, p$. Next, we define the augmented matrix

$$\tilde{L} := \begin{bmatrix} L & W \\ 0 & J \end{bmatrix} \in \mathbb{R}^{(n+p)\times(n+p)}, \text{ where } J := \begin{bmatrix} 0 & I_{p-1} \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{p\times p}. \tag{5.7}$$

Then, the integral in (5.5) can be computed via the matrix exponential of $\tilde{L}$. It can be shown that

$$\varphi_0((t-t_0)\tilde{L}) = e^{(t-t_0)\tilde{L}} = \begin{bmatrix} e^{(t-t_0)L} & K \\ 0 & e^{(t-t_0)J} \end{bmatrix}, \tag{5.8}$$

where the last column of $K$ is

$$K(1:n,p) = \sum_{l=1}^{p} (t-t_0)^l \varphi_l((t-t_0)L) g_{t_0}^{(l-1)}.$$

Therefore, the right hand side of (5.5) can be written as

$$\hat{x}(t) = \begin{bmatrix} I_n & 0 \end{bmatrix} e^{(t-t_0)\tilde{L}} \begin{bmatrix} x_0 \\ e_p \end{bmatrix}, \tag{5.9}$$

where $e_p := \begin{bmatrix} 0_{p-1} \\ 1 \end{bmatrix}$, the last column of a $p$-dimensional identity matrix.

In an explicit time-stepping method, also called exponential time differencing in the literature [112], the evaluation of the solution of the semilinear ODE (5.1) can be done in a semi-analytical fashion by computing the action of a matrix exponential on a vector. By discretizing the time span of interest, the ODE can be integrated as accurately as we can calculate the matrix exponential or its action on a vector. This is one of the reasons that motivates the need for a computationally inexpensive method to compute expressions of the form $e^L y$, where $y$ is a vector [113, Sec. 3]. The computational saving made in not calculating $\varphi$-functions is most pronounced in applications where $n \gg p$; semidiscretized PDEs often result in such approximations, where $n$ is a number of orders bigger than $p$. Although not relevant to the bilinear systems we consider here, it is also important to mention that the nonlinearity in (5.2) can be approximated using various quadrature rules to derive exponential variants of the classical linear multistep, Runge-Kutta and general linear methods of implicit and explicit kinds [112]. See [114]

for a survey of existing exponential methods for general semi-linear systems, their error analysis, and Butcher tableau and some historical perspective going back decades.

## 5.2 Exponential Integrators for Bilinear Systems

Here, we consider bilinear systems of the form:

$$\dot{x}(t) \quad = Ax(t) + \sum_{i=1}^{r} N_i x(t) u_i(t) + Bu(t), \quad x(t_0) = x_0 \tag{5.10}$$

where $u \in \mathbb{U}$ and $\mathbb{U} := [u \in \mathbb{R}^m | \ |u_i| \leq 1\}$, $r \leq m$.

As a result of the nonlinear term, there is no simple computable variation-of-constants formula, unlike for linear systems [115, 116]. The exact solution involves an infinite Volterra series and is not suitable for direct computations. Here we consider the case of bilinear control systems with piece-wise constant control. To find an exact expression for the integral of such systems, we use the following result.

**Lemma 5.1.** *The solution of the non-autonomous initial value problem*

$$\dot{x}(t) = Lx(t) + \sum_{j=0}^{p-1} \frac{t^j}{j!} b_j, \quad x(0) = x_0, \, \forall t \geq 0, \tag{5.11}$$

*is given by*

$$x(\tau) = \varphi_0(\tau L) x_0 + \sum_{j=0}^{p-1} \tau^{j+1} \varphi_{j+1}(\tau L) b_j, \ \ \forall \tau, \, 0 < \tau < h \tag{5.12}$$

*where the functions $\varphi_i(\cdot)$ are as defined in* (5.4).

*Proof.* See Appendix B. □

Lemma 5.1 allows the solution of linear control systems with a polynomial-order hold (i.e. with piecewise polynomial in time control inputs) to be accurately computed using new results in the field of exponential integrators; in this thesis, we consider inputs sampled with zero-order hold .

### 5.2.1 Exact solution of bilinear control systems with ZOH

We are interested in algorithms that solve the ODE in (5.10) exactly to a sufficiently high accuracy for control applications. As mostly is the case in digital control systems,

we consider the case where the control input is generated by a zero-order hold sample (ZOH) on some signal. The input being constant in each sampling interval allows us to exactly evaluate the integral at the collocation points. Then, in some control interval $k$, the bilinear system can be considered a time-invariant linear system and (5.10) can be re-written as the linear system

$$\dot{x}(t) \quad = [A + \sum_{i=1}^{r} N_i u_i(t_k)]x(t) + Bu(t_k), \tag{5.13}$$

where $t \in [t_k, t_{k+1})$ and $u(t) = u(t_k)$, $\forall t \in [t_k, t_{k+1})$, $t_{k+1} := t_k + h$, $t_0 = 0$, $k = 0, 1, \ldots, N$ where $h$ is the (constant) ZOH or control interval. Then, the exact solution of (5.13) at time $t_{k+1}$ is given by the variation-of-constants formula

$$x(t_{k+1}) = e^{h\hat{A}_k} x(t_k) + \int_0^h e^{(h-\tau)\hat{A}_k} Bu(t_k) d\tau, \tag{5.14}$$

where $\hat{A}_k = [A + \sum_{i=1}^{r} N_i u_i(t_k)]$ and we assume $|u_i(t)|$ is bounded for all $i$ and time $t$.

It appears from (5.14) that the system can be integrated exactly using exponential integrators. Looking back at the formulation in Section 5.1 and Lemma 5.1, we can re-write (5.14) as

$$x(t_{k+1}) = \quad e^{h\hat{A}_k} x(t_k) + h\varphi_1(h\hat{A}_k) Bu(t_k), \tag{5.15}$$

$$= \qquad \begin{bmatrix} I_n & 0 \end{bmatrix} e^{h\tilde{A}_k} \begin{bmatrix} x_k \\ 1 \end{bmatrix}, \tag{5.16}$$

where $x_k = x(t_k)$ and $\tilde{A}_k = \begin{bmatrix} \hat{A}_k & Bu_k \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}$.

Often, we want to integrate the bilinear system (5.13) over a range $[t_0,\ t_f]$ with $N\ (= \frac{t_f - t_0}{h})$ intervals. Since the $\hat{A}_k$ could be different in each of the $N$ ZOH intervals, we are potentially computing the action of up to $N$ matrix exponentials on a vector. Therefore, the results of [113] and related research are relevant — we need to compute these actions cheaply (with respect to $n$ and $N$).

Unlike classical time-stepping integration methods, the integration (5.14) does not require a Jacobian linearization around some nominal input and state trajectory. Based on the variation of constants formula, (5.14) evaluates the bilinear system exactly where a ZOH is applied on the input — a ZOH input is the case for many discrete control applications and almost all digital controllers. Moreover, the matrix exponential $e^{h\hat{A}_k}$ and the integrals in (5.14) can be computed offline and stored if the input is 'quantized' and takes on a relatively small number of values (i.e. computationally not too many

for available memory resources). This would allow for an offline computation of the exponents and real-time integration would be very efficient; each integration step will cost a single matrix vector multiplication. In the next section, we discuss methods of computing the action of the matrix exponential and its action on a vector.

## 5.3 Exponential Integrators with only Matrix-Vector Multiplication

We have shown that bilinear systems can be integrated by computing the action of a matrix exponential on a vector, $e^X y$, where $X \in \mathbb{R}^{n\times n}$ and $y \in \mathbb{R}^n$. A naive approach for computing $e^X y$ with very high accuracy would be to compute $e^X$ sufficiently accurately using one of the many methods for computing the matrix exponential and then multiply with the vector $y$. However, despite huge advances in making matrix exponential computations more efficient, the process is expensive ($\approx O(n^3)$) and such a naive approach would not take advantage of the information in the vector $y$. Especially in the case where $X$ is sparse, $e^X$ becomes denser, hence destroying sparsity for the next matrix-vector multiplication. On the other hand, Krylov subspace based methods project the exponential operator onto a much smaller subspace, taking advantage of the relationship between $X$ and $y$, and then use minimal computations for the smaller matrix exponential. The use of Krylov subspace methods for integration and their error analysis goes back at least thirty years [117] and they have been applied in exponential integrators of large scale systems arising in various disciplines ranging from quantum physics to mathematical finance; see [114] for a thorough list of applications.

An exhaustive list of methods for computing the matrix exponential is reviewed in [118]. Of the "19 dubious methods" reviewed in [118], the "scaling and squaring method" by Higham [119, 120] is the most popular due to its numerical efficiency and advanced error analysis. As such, it is even used to compute the smaller size Hessenberg matrix exponential within Krylov subspace exponential integrators. Below we first review the squaring and scaling method, and then discuss two alternative sparse implementations for exponential integrators.

### 5.3.1 The Scaling and Squaring method for computing $e^X$

The Padé and Taylor series approximations for the matrix exponential $e^X$ have computational cost and roundoff errors that increase with the norm $\|X\|$ and the spread of its eigenvalues [118, 120]. However, for small norm $\|X\|$ or small spectral radius $\rho(X)$, which is the largest absolute value of the eigenvalues of $X$, these are two of the most

effective approximations both in accuracy and computational cost. Based on the identity $e^X = (e^{X/\tau})^\tau$, a matrix is first scaled so that its norm or spectral radius is small and its approximation is then raised to the same power. The scaling and squaring method of [119, 120] uses a scale $\tau = 2^s$, with some positive integer $s$, so that the Padé approximation of $e^{X/2^s}$ can be raised to the same power via $s$ successive squarings.

The parameter $s$ is chosen so that $2^{-s}X$ has a norm of order 1. Then, an appropriate order diagonal Padé approximation (i.e. $R_l(2^{-s}X) = D_l(2^{-s}X)^{-1}N_l(2^{-s}X)$) is chosen to guarantee specified backward error bounds on the exponential; see Appendix C for explicit expressions of $D_l(\cdot)$ and $N_l(\cdot)$. For an appropriately scaled matrix $X$, it is shown [120, Thm. 2.1] that, with $R_l(2^{-s}X)^{2^s} = e^{X+\Delta_X}$, the backward error $\Delta_X$ in the scaling and squaring computation can be bounded by

$$\frac{\|\Delta_X\|}{\|X\|} \leq \frac{-\log(1-f(\theta))}{\theta}, \tag{5.17}$$

where $\theta = \|2^{-s}A\|$, $f(\theta) := \sum_{k=2l+1}^{\infty} |c_i|\theta^i$ and $c_i$ are known exactly from the Padé expansion and the scaling is such that the power series converges [119, (2.2) and (2.6)]. The backward error for algorithms is defined as follows [121, Sec. 1.5]. Assume an algorithm computes the approximation $\hat{y}$ for $y = f(x)$, where $f(\cdot)$ is some function. Then the backward error is defined as

$$\min\{\|\Delta x\| | \hat{y} = f(x+\Delta x)\}.$$

In other words, it is the smallest perturbation in the data $x$ whose exact solution is the same as the computed solution $\hat{y}$. The forward error is the error in the computation $y-\hat{y}$. Therefore, if we require a given accuracy $\frac{\|\Delta_X\|}{\|X\|} \leq tol$ (for example, double precision level accuracy $tol = 2^{-53}$), upper bounds on $\theta$ for a range of orders $l$ can be computed offline and stored as $\theta_l$ in algorithm design.

Generating $\theta_l$ for arbitrary tolerances would require symbolic computation to get reliable values. The authors of [113] generated them using symbolic computations with a zero-finder for the polynomial $\sum_{k=l+1}^{150} |c_k|\theta^{k-1} = tol$, a 250 decimal place arithmetic was used with $tol$ equal to the unit roundoff level (or eps) of IEEE half, single, double and quadruple precision floating point arithmetic. Since symbolic computations are very expensive, we have found it impractical to generate these parameters for an arbitrary tolerances. We have therefore opted to use the values generated by the authors of [113].

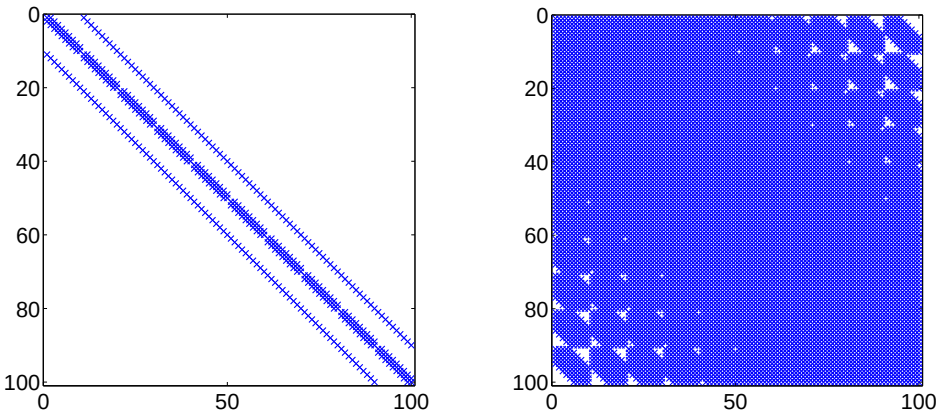
In an optimal algorithm implementation, an important element in choosing $s$ and $l$ for a given error bound $tol$ is to use a combination that minimizes the cost of computation.

Let $C(l,s)$ represent the cost in number of matrix-matrix multiplications. Then, it can be shown that $C(l,s) = \pi_l + s$, where $\pi_l (\approx l/2)$ represents the number of matrix-matrix multiplications required to evaluate $D_l(X)$ and $N_l(X)$ such that $D_l(2^{-s}X)R_l(2^{-s}X) = N_l(2^{-s}X)$. The number of squarings needed is shown to be $s = \max\{\lceil \log_2 \|X\|/\theta_l \rceil, 0\}$, where $\theta_l$ are maximal values that $\|2^{-s}X\|$ can take such that the backward error bound $\frac{\|\Delta_X\|}{\|X\|} < tol = 2^{-53}$, is still guaranteed.

It is also shown in [120] that $l = 13$ is the optimal degree that achieves the least number of multiplications $s$ for a backward error bounded by the IEEE double precision unit roundoff — $l = 7$ and $l = 17$ are also shown to be optimal for the IEEE single ($tol = 2^{-24} \approx 6.0 \times 10^{-8}$) and quadruple precision arithmetic eps ($tol = 2^{-105} \approx 2.5 \times 10^{-32}$), respectively. The version of the scaling and squaring method with this latest error analysis is implemented as the function `expm` in Matlab versions 2010a and newer. Although any matrix $p-$norm, $p \in \{1,2,\infty\}$, will work in the above analysis, the 1-norm is used for its relative computational ease ($\|X\|_1 := \max\limits_{1\le j\le n} \sum\limits_{i=1}^{n} |X_{i,j}|$, $\forall X \in \mathbb{R}^{n\times n}$ (i.e. maximum column sum) [122, (2.3.8)]).

Of course, there is also a fixed cost associated with solving the linear equation $D_l(\cdot)R_l(\cdot) = N_l(\cdot)$, where $D_l(\cdot)$, $N_l(\cdot)$ are evaluated as given in [120, Eq. 3.4-3.6]; $D_l$ is also shown to be non-singular under some mild conditions (e.g. in the necessary case of $\|2^{-s}X\| \le \theta_l$.)

However, exponential integrators require not the matrix exponential, but its action on a vector. Although the improved squaring and scaling method of [120] is presented as possibly the best choice to compute the matrix exponential, it does not solve the need to compute the action on a vector efficiently. Figure 5.1a shows the sparsity of the dynamics matrix from a model of the metal slab cooling bilinear control system in Section 5.5. Considering a Padé approximation of order $l = 13$, we can see from Figure 5.1b that the linear systems that the approximation needs to solve (i.e. $D_l(2^{-s}X)R_l(2^{-s}X) = N_l(2^{-s}X)$ are full. Moreover, the solution $R_l(2^{-s}X)$ ends up 100% full. In other words, even for the sparsest problems coming from finite difference models of PDEs, the matrix exponential ends up being full and its computational cost is $O(n^3)$. This is undesirable for large systems. Where the new modified scaling and squaring method has to be used (for example, to find the exponential of the Hessenberg matrix within Krylov subspace methods), we refer the reader to [123] for a discussion on efficient computation of the Padé approximations and the corresponding computational cost analysis. Below we consider alternative methods that exploit sparsity in the problem.




(a) $2^{-s}X$ has 4.6% non-zero elements (b) $D_l(2^{-s}X)$ has 97% non-zero elements

Figure 5.1: The sparsity pattern for the sparse dynamic matrix from a bilinear PDE and the sparsity of the matrix $D_l(2^{-s}X)$, $l = 13$ solved by the Padé approximation.

### 5.3.2 An efficient scheme by Al-Mohy and Higham [113] for computing $e^X y$

Motivated by possible computational savings for the case of large and sparse matrices, a new Taylor approximation based method that adapts the scaling and squaring scheme is presented by Al-Mohy and Higham in [113]. Adapting the backward error bounds of [120] and deriving the result shown in (5.9), a method for numerical solution for ODEs is presented. Exploiting the scaling identity as before, $e^X y$ is rewritten as

$$e^X y = (e^{\bar{s}^{-1}X})^{\bar{s}} y = \underbrace{e^{\bar{s}^{-1}X} e^{\bar{s}^{-1}X} \dots e^{\bar{s}^{-1}X}}_{\bar{s} \text{ times}} y \tag{5.18}$$

where the recurrence

$$b_{i+1} := T_l(\bar{s}^{-1}X) b_i,\ i = 0, 1, \dots, \bar{s}-1,\ b_0 := y \tag{5.19}$$

gives $b_{\bar{s}} \approx e^X y$. A truncated Taylor series of order $l$ is used to approximate the matrix exponential, $T_l(x) = \sum_{k=0}^{l} \frac{x^k}{k!}$.

Here the integer $\bar{s}$ is not necessarily a power of 2 and, as a result there is no squaring, but multiplication of an $n \times n$ matrix with a vector; the computational cost of (5.18) is $O(n^2)$ for a general matrix $X \in \mathbb{R}^{n \times n}$. The Taylor series approximation of the matrix exponential is used because, unlike the Padé approximation, it avoids the solution of dense linear systems, while allowing a similar backward error analysis as in the scaling and squaring method of [119]. With $\bar{s}$ chosen, such that $\bar{s}^{-1}X$ has a sufficiently small norm, we have

$$T_l(\bar{s}^{-1}X)^{\bar{s}} := e^{X+\Delta X}, \tag{5.20}$$

where $\Delta_X$ is the backward error associated with the truncation errors in the approximation $e^X \approx T_l(\bar{s}^{-1}X)^{\bar{s}}$. With an appropriate scaling, $\Delta_X$ has the power series expansion

$$\Delta_X = \bar{s}h_{l+1}(\bar{s}^{-1}X), \tag{5.21}$$

where $h_l(x) := \sum_{k=l}^{\infty} c_k x^k$, and the coefficients $c_k$ are known exactly [119, Sec. 2]. The aim of approximating $e^X$ by $T_l(\bar{s}^{-1}X)^{\bar{s}}$ with some a priori backward error tolerance $tol$ means choosing $\bar{s}$ and $l$ so that the tail of the series satisfies:

$$\frac{\|\Delta_X\|}{\|X\|} = \frac{\|h_{l+1}(\bar{s}^{-1}X)\|}{\|\bar{s}^{-1}X\|} \leq tol. \tag{5.22}$$

In the original scaling and squaring method of [119], it is shown that the backward error in (5.22) can be bounded using the following possibly conservative bound for the norm of the tail of the Taylor approximation sequence $h_l(\cdot)$:

$$\|h_l(A)\| \leq \sum_{k=l}^{\infty} |c_k| \|X\|^k. \tag{5.23}$$

When $\|X\|$ is large, a naive choice of a large scaling $\bar{s}$ can cause the accumulation of roundoff errors in the squaring phase of the scaled matrix. With the objective to avoid unnecessarily large choices for the scaling, a sharper bound was provided in [120, Thm. 1.1] by making use of the sequence $\{\|X^k\|^{1/k}\}$; we reproduce this result here. Let us define a new function $\tilde{h}_l(x) := \sum_{k=l}^{\infty} |c_k| x^k$. For a bounded and convergent sequence $h_l(\cdot)$, with the definition $\|X^{k^*}\|^{1/k^*} = \max\{\|X^k\|^{1/k} : k \geq l \text{ and } c_k \neq 0\}$, it follows [120, Thm. 1.1] that

$$\begin{aligned} \|h_l(X)\| &\leq \sum_{k=l}^{\infty} |c_k| \|X^k\| = \sum_{k=l}^{\infty} |c_k| (\|X^k\|^{1/k})^k, \\ &\leq \sum_{k=l}^{\infty} |c_k| (\|X^{k^*}\|^{1/k^*})^k = \tilde{h}_l(\|X^{k^*}\|^{1/k^*}). \end{aligned} \tag{5.24}$$

The bound $\tilde{h}_l(\|X^{k^*}\|^{1/k^*})$ is sharper than (5.23) because $\|X^k\|^{1/k} \leq \|X\|$, for $k = 1, 2, \ldots$. The sharpness is also more pronounced in cases where it has been empirically shown that $\|X^{k^*}\|^{1/k^*} << \|X\|$; example problems have been used to show this in [120]. Of course, for a general matrix we do not know $\|X^{k^*}\|^{1/k^*}$; only for a nilpotent matrix $X$ or possibly for simple small matrices could this maximum be computed. The possibility of computing $\|X^k\|^{1/k}$, $k = 1 : l_{max}$, where $l_{max}$ is some large number is not a computationally tractable or desirable task. A practical method for estimating $\|X^{k^*}\|^{1/k^*}$ is presented in [120, Lemma 4.1, Lemma 4.2]. We reproduce below some relevant results from these lemmas.

**Lemma 5.2.** *[120, Lemma 4.1] For any $k \geq 1$ such that $k = pm_1 + qm_2$ with $p,q \in \mathbb{N}$ and $m_1, m_2 \in \mathbb{N} \cup \{0\}$,*

$$\|X^k\|^{1/k} \leq \max\{\|X^p\|^{1/p}, \|X^q\|^{1/q}\}. \tag{5.25}$$

**Lemma 5.3.** *[120, Lemma 4.2] With $h_l(\cdot)$ and $\tilde{h}_l(\cdot)$ as defined above, assume the sequence $h_l(A)$ is convergent and $p \in \mathbb{N}$. Then*

$$\|h_l(X)\| \leq \tilde{h}_l(\max\{\|X^p\|^{1/p}, \|X^{p+1}\|^{1/(p+1)}\}), \text{ if } l \geq p(p-1). \tag{5.26}$$

This means, for an order $l$ Taylor approximation, the backward error can be bounded by:

$$\frac{\|\Delta_X\|}{\|X\|} = \frac{\|h_{l+1}(\bar{s}^{-1}X)\|}{\|\bar{s}^{-1}X\|} \leq \frac{\tilde{h}_{l+1}(\bar{s}^{-1}\alpha_p(X))}{\bar{s}^{-1}\alpha_p(X)}, \tag{5.27}$$

where

$$\alpha_p(X) := \max\{\delta_p, \delta_{p+1}\},\ \delta_p := \|X^p\|^{1/p}, \tag{5.28}$$

and $p$ satisfies $p(p-1) \leq l+1$. The effectiveness of this estimate is its ease of computation. For example, if $l = 100$, we compute $\|X^p\|_1$ only for up to $p = 11$.

In algorithm design, a set of $\theta_l = \max\{\theta : h_{l+1}(\theta)/\theta < tol\}$ can be computed and stored for various backward error tolerances and Taylor approximation order $l$. The computational cost of the iteration in (5.18) using parameters $\bar{s}$ and $l$ chosen using the 1-norm bound in (5.23) can be shown to be

$$C_l(X) := l\bar{s} = l\lceil \|X\|_1/\theta_l \rceil \tag{5.29}$$

*matrix-vector* multiplications. Note the emphasis here; there are no matrix-matrix multiplications. The parameter $l$ is chosen to minimize the above cost (5.29). However, where $\alpha_p(X) << \|X\|_1$, $l$ can be chosen to minimize the alternative computational cost:

$$\begin{aligned} C_l*(X) &= \min_l\{l\lceil \alpha_p(X)/\theta_l \rceil : 2 \leq p \leq p_{max},\ p(p-1) \leq l+1 \leq l_{max}\}, \\ &= l^*\bar{s}^* = l^*\lceil \alpha_p(X)/\theta_{l^*} \rceil, \end{aligned} \tag{5.30}$$

where $l_{max}$ is the maximum limit imposed on the Taylor approximation order and $l^*$ denotes the smallest such $l$ that attains the minimum in (5.30). The computational cost in evaluating $e^X y$ is then proportional to $l^*\bar{s}^* n^2$ for a general matrix $X$ and vector $y$ of size $n$.

Although it happens that $l/\theta_l$ is a monotonically decreasing function, the authors of [120] have imposed a maximum value $l_{max}$ for floating point arithmetic implementations on a computer, since a very large $l$ may result in roundoff error accumulation. Since the

computation of optimal orders for the Taylor approximations does not consider the problem data $y$ or its relationship with powers of $X$, the iterative implementation of the method in [113] uses this information. With $l^*$ and $\bar{s}^*$ chosen to minimise $C_l(X)$ ((5.29) or (5.30)), the pseudocode below shows a snippet of the iterative computation of (5.18) from [113, Algorithm. 3.2]:

**Algorithm 3** Compute $e^X y$: adapted snippet from [113, Algorithm. 3.2] with no preprocessing.

1: Choose best parameters $l^*$ and $\bar{s}^*$ for tol
2: $b = y,\ \ f = y,$
3: **for** $i = 1$ to $\bar{s}^*$ **do**
4: $\quad c_1 \leftarrow \|y\|_\infty$
5: $\quad$ **for** $j = 1$ to $l^*$ **do**
6: $\quad\quad b \leftarrow Xb/(sj),\ c_2 = \|b\|_\infty$
7: $\quad\quad f \leftarrow f + b$
8: $\quad\quad$ if $c_1 + c_2 \leq tol\|f\|_\infty$, break and end
9: $\quad\quad c_1 \leftarrow c_2$
10: $\quad$ **end for**
11: $\quad b \leftarrow f$
12: **end for**

From Line 8 of the pseudocode above, we can see that the method does not necessarily use an $l^*-$order Taylor approximation to compute $T_l(X)b_i$ in (5.19); if the norm of $\sum_{j=k+1}^{l} \frac{X^j}{(j!s)} b_i$, for some $k < l^*$, is less than the requested tolerance, the Taylor series is terminated prematurely to use a $k^{th}$ order approximation. The authors in [113] have empirically shown that the use of two terms in the tail is a reliable approximation, i.e. Line 8 translates to $\frac{\|X^{k-1}b_i\|}{(k-1)!} + \frac{\|X^k b_i\|}{k!} \leq tol\frac{\|X^k b_i\|}{k!}$. Since the algorithm spends most of its resources performing these matrix-vector multiplications ($\approx$88% of CPU time [113]), this early termination process can bring significant computational savings. This is in the spirit of what is at the heart of the iterative Krylov subspace methods we see in the next subsection. However, unlike the fully iterative Krylov subspace methods, this algorithm is "direct rather than iterative" [113] since we can tightly bound the number of operations needed to compute $e^X y$ a priori once we compute the norm of $X$ or $\alpha_p(X)$. Using a variety of large sparse problems, it is shown in [113] that this algorithm performs as well as the best Krylov subspace implementations.

### 5.3.3 A priori bounds on computational complexity for exponential integration of a bilinear system

We have shown in Section 5.2.1 that exponential integrators can be used for 'exact to a given accuracy' integration of bilinear systems with ZOH on the input. We have also discussed the use of a new efficient direct method from [113] for the computation of the action of a matrix exponential on a vector. Unlike iterative methods, this new scheme allows the a priori computation of the computational complexity for a given backward error bound on the solution. We now consider again the problem solved at each time step, $t_k$, $k = 1,\ldots,N$, in the bilinear integration (5.15):

$$x(t_{k+1}) = \begin{bmatrix} I_n & 0 \end{bmatrix} e^{h\tilde{A}_k} \begin{bmatrix} x_k \\ 1 \end{bmatrix}, \tag{5.31}$$

where $x_k = x(t_k)$ and $\tilde{A}_k = \begin{bmatrix} \hat{A}_k & Bu_k \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}$, $\hat{A}_k = [A + \sum_{i=1}^{r} N_i u_i(t_k)] \in \mathbb{R}^{n\times n}$.

We may want to precompute and store the best scaling $s$ and the Taylor expansion order $l$ that guarantee a given backward error tolerance rather than recomputing $s$ and $l$ at every sampling step for new values of $u_k$. This will be especially necessary in direct transcription of optimal control problems where the same $s$ and $l$ will have to be used for all intervals; see Section 5.4. If we use the 1-norm of the augmented matrix $\tilde{A}_k$ to bound the error as in (5.29), an upper bound on $\|\tilde{A}_k\|_1$ can be derived for all feasible input sequences. From the definition of the matrix 1-norm (i.e. the maximum 1-norm of the columns), we get:

$$\|\tilde{A}_k\|_1 \leq \max\left(\|\hat{A}_k\|_1, \|Bu_k\|_1\right), \tag{5.32}$$

$$\|\tilde{A}_k\|_1 \leq \max\left(\gamma, \beta\right), \tag{5.33}$$

where $\gamma$ and $\beta$ are the upper bounds

$$\gamma := \|A\|_1 + \sum_{i=1}^{r} |\max_t(u_i(t))| \|N_i\|_1 = \|A\|_1 + \sum_{i=1}^{r} \|N_i\|_1 \geq \|\hat{A}_k\|_1, \tag{5.34}$$

$$\beta := \|B\|_1 \|u\|_1 = m\|B\|_1 \geq \|Bu_k\|_1. \tag{5.35}$$

With $|\max_t(u_i(t))| = 1$ considered without loss of generality, the two inequalities (5.34) and (5.35) simply result from the triangle inequality and positive scalability properties

of a matrix norm.[1]

As we will demonstrate shortly, the bound (5.33) can be very conservative depending on the problem. The alternative is to use the possibly significantly less conservative measure $\alpha_p(\tilde{A}_k)$ in (5.27)-(5.28).

Simple matrix arithmetic reveals:

$$\tilde{A}_k^j = \begin{bmatrix} \hat{A}_k^j & \hat{A}_k^{j-1} B u_k \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}, \forall j \geq 2 \tag{5.36}$$

To compute $\alpha_p(\tilde{A}_k) = \max\{d_p, d_{p+1}\}$, $d_p := \|\tilde{A}_k^p\|^{1/p}$, it is clear from Lemmas 5.2–5.3 that we can use bounds on $d_p$. We propose the following bound:

$$\|\tilde{A}_k^p\|_1 \leq \max\|\bar{A}^p\|_1, \|\bar{A}^{p-1}\|_1 \beta, \tag{5.37}$$

where $\bar{A}(i,j) = |A(i,j)| + \sum_{l=1}^{r} |N_l(i,j)|$ and $\beta$ is the bound on $\|Bu\|_1$ in (5.35).

Since $\bar{A}$ is formed by summing element-wise absolute values of the matrices, the bound can in principle be conservative. Although simulations with random matrices indicate that the latter bound is not always better than the first, they suggest it generally does better. In any case, since these computations do not have to be done online, computing both and using the better bound will suffice. Table 5.1 compares the optimal $l$ and $s$ generated via the bounds on the 1-norm from (5.33) and on $\alpha_p$ (5.37), respectively, for the bilinear wave energy control system discussed in Chapter 3. It is clear that, under all the given sampling times the 1-norm based bound gives a bigger order of approximation than the one based on (5.37). Although not shown here, the $\alpha_p(\cdot)$ bound we have proposed is observed to be similarly less conservative for all example bilinear systems considered for direct collocation in Section 5.4.

---

[1]

$$\begin{aligned} \text{Definition: } \|X\|_p &:= \max_{\|v\|=1} \|Xv\|_p. \\ &\quad \text{For any } X \in \mathbb{R}^{m\times n}, v \in \mathbb{R}^n,\ \|Xv\|_p \leq \|X\|_p \|v\|_p \text{ follows from the definition.} \\ \Longrightarrow \|AB\|_p &= \max_{\|v\|=1} \|ABv\|_p \leq \max_{\|v\|=1} \|A\|_p \|Bv\|_p = \|A\|_p \|B\|_p \end{aligned}$$

Table 5.1: Optimal $l$ and $s$ computed using bounds on the 1-norm (5.33) and $\alpha_p(\cdot)$ (5.37). The bilinear wave energy system (3.7) with backward error tolerance equal to the IEEE single precision eps $tol = 2^{-24} \approx 6 \times 10^{-8}$ is used; the corresponding values for IEEE half precision eps $tol = 2^{-11} \approx 5 \times 10^{-4}$ are shown in brackets. The maximum value for $l$ was limited to $l_{max} = 100$ and $h$ is the integration stepsize. $C_{ratio}$ denotes the ratio of the computational complexities of the two bounds in number of matrix-vector multiplications; it is equal to to the quotient of the computational complexity (5.29) when using the $\|\tilde{A}_k\|_1$ and the computational complexity (5.30) when using the bound on $\alpha_p(\tilde{A}_k)$.

| | Using bound on $\|\tilde{A}_k\|_1$ | | Using bound on $\alpha_p(\tilde{A}_k)$ | | |
|---|---|---|---|---|---|
| $h$ | $l^*$ | $s^*$ | $l^*$ | $s^*$ | $C_{ratio}$ |
| 0.001 | 6 (3) | 1 (1) | 5 (4) | 1 (1) | 1.2 (0.75) |
| 0.01 | 15 (10) | 1 (1) | 7 (5) | 1 (1) | 2.1 (2.0 ) |
| 0.1 | 45 (74) | 2 (1) | 11 (8) | 1 (1) | 8.2 (9.3) |
| 1 | 60 (92) | 14 (8) | 24 (19) | 1 (1) | 35 (38.7) |

For the exponential integration method, we have discussed error bounds in terms of the backward error. However, in the numerical integration of ODEs we may want to give a guarantee on the forward relative error, which is a better measure to determine if the computed results are acceptable. Of course, the rule [124, Sec. 3.1]

$$\text{forward error} \;\leq\; \text{condition number} \;\times\; \text{backward error},$$

where the condition number of the matrix exponential,

$$\kappa_{exp}(X) := \lim_{\varepsilon \to 0} \sup_{\|\Delta_X\| \leq \varepsilon} \frac{\|e^{X+\Delta_X} - e^X\|}{\|e^X\|} \tag{5.38}$$

can be efficiently approximated using various algorithms [124, Ch. 2]. For a stable method, it can be expected that the error in computing $e^X y$ to be bounded by a "modest multiple" of $\kappa_{exp}(X)$eps, where eps is the machine accuracy [120]. The authors of [120] have demonstrated this using various benchmark problems; in our simulations we have used their algorithm implemented in the function `expm_cond` from the Matrix Function Toolbox [124] to check this. Therefore, in determining a useful forward error bound, one can make use of bounds on the condition number of the exponential with the given problem data.

### 5.3.4 A Krylov subspace method for computing $e^X y$

Krylov subspace methods are applied to large sparse problems in a variety of fields from iterative linear solvers [125], to the model reduction of large scale systems [92] and almost universally in exponential integrators for PDEs [114]. Before discussing the

tenet of Krylov subspace methods or their application in exponential integrators, we will very briefly refer to the subject of "functions of matrices".

In [124], various ways are used define the function of a matrix. For a matrix $X \in \mathbb{C}^{n\times n}$, a compact definition is based on the Cauchy integral theorem [122, 124]:

$$f(X) = (f_{kj}), \text{ where } f_{kj} := \frac{1}{2\pi i}\oint_{\Gamma} f(z) e_k^T (zI - X)^{-1} e_j dz, \tag{5.39}$$

where $\Gamma$ is a closed contour surrounding the eigenvalues of $X$ (i.e. all $\lambda \in \Lambda(X)$) and the function $f(\cdot)$ is analytic on $\Gamma$ and its interior. Such a function $f(\cdot) \in \mathbb{C}^{n\times n}$ can also be defined on the spectrum of the matrix $X$ through the existence of its derivatives on each distinct eigenvalue of $A$ [124, Definition 1.1].

An important property of matrix functions allows the use of Krylov subspaces with the Arnoldi process for approximating $e^X y$. It says, for a given $X \in \mathbb{C}^{n\times n}$, any matrix function as defined above satisfies $f(X) = p_k(X)$, where $p_k(\cdot)$ is a polynomial function of the matrix $X$

$$p_k(X) = a_k X^k + a_{k-1} X^{k-1} + \ldots + a_1 X + a_0 I$$

with order $k \leq n-1$. One such polynomial for any $X$ is the Hermite interpolating polynomial on the spectrum of $X$; see [124, Def. 1.4].

On the other hand, the $k^{th}$ *Krylov subspace* of $X \in \mathbb{C}^{n\times n}$ and a non-zero vector $y \in \mathbb{C}^n$ is defined as:

$$\mathcal{K}_k(X, y) := \text{span}\{y, Xy, \ldots, X^{k-1}y\}.$$

or

$$\mathcal{K}_k(X, y) := \text{span}\{q(X)y \ : \ q(\cdot) \text{ is a polynomial of degree } \leq k-1\}.$$

Therefore, $f(X)y \in \mathcal{K}_d(X, y)$, where $d \leq n$ the smallest degree polynomial representation for $f(X)$. In other words, the size of a Krylov subspace necessary to capture $f(X)y$ depends on both $X$ and $y$. Often, for large $n$, a small order approximation $f(X)y \approx p_d(X)y$, $d << n$ may suffice. This is what we are interested in.

**Algorithm 4** The Arnoldi iteration with a normalised vector

1: Start $v_1 = y/\|y\|_2$
2: **for** $j = 1$ to $n$ **do**
3: $z = Xv_j$
4: **for** $i = 1$ to $j$ **do**
5: $h_{i,j} = v_i^T z$
6: $z = z - h_{i,j} v_i$
7: **end for**
8: $h_{j+1,j} = \|z\|_2$
9: **if** $h_{j+1,j} = 0$ (or $\leq tol\ \|X\|$), $l = k$ quit; **happy breakdown**
10: $v_{j+1} = z/h_{j+1,j}$
11: **end for**

The Arnoldi process in Algorithm 4 generates, after $l$ steps, a sequence of orthonormal basis $V_l = [v_1, v_2, \ldots, v_l]$ of the Krylov subspace $\mathcal{K}_l$ (spanned by $\{y, Xy, X^2y, \ldots, X^{l-1}y\}$) and what is the $l-$step partial Hessenberg reduction satisfies

$$XV_l = V_l H_l + h_{l+1,l} v_{l+1} e_l^T,$$

where $e_i$ is the $i^{\text{th}}$ column in the identity matrix $I_l$, $H_l \in \mathbb{C}^{l \times l}$ consisting of the coefficients $h_{i,j}$ computed by Line 5 is upper Hessenberg (i.e. almost upper triangular — has zero entries below the first subdiagonal) and $v_{l+1}$ is a unit vector that satisfies $V_l v_{l+1} = 0$. This implies the important property [117, Thm. 3.1]

$$q_k(X)y = \|y\|_2 V_l q_k(H_l) e_1$$

for all polynomials $q_k$ of degree $\leq l-1$. This property allows the approximation of matrix functions by a polynomial of degree $\leq l$

$$f(X)y \approx \|y\|_2 V_l f(H_l) e_1.$$

The additional property $V_l^T(\tau X)V_l = \tau H_l$, makes the Kryolv subspace invariant to the matrix scaling process. For general matrix functions $f(X)$, no useful bounds are known for the error of the approximation $\|f(X)y - \|y\|_2 V_l f(H_l) e_1\|$. However, in approximating the action of the exponential function using the above iteration for the exponential function $f(X) = e^X$, some conservative a priori error bounds and highly reliable a posteriori error estimates are known. The work in [117] analyses Krylov methods for computing the matrix exponential and derives some a priori and a posteriori forward error bounds. In [117, Thm. 4.3, 4.7], a priori forward error bounds on the estimate are

provided as a function of the spectrum of $A$

$$\|e^X y - bV_l e^{H_l} e_1\|_2 \leq 2b\frac{a^l e^a}{l!},\ \ a = \|X\|_2,\ b = \|y\|_2.$$

The corrected algorithm that uses one more iteration in the Arnoldi iteration has a bound $\|error\| \leq 2b\frac{a^{l+1}e^a}{(l+1)!}$. However, unless the norm of $A$ is very small, these bounds give errors that are many orders of magnitude bigger than actual errors. The lack of sharpness in these bounds and the absence of other useful a priori bounds meant reliable a posteriori estimate had to be sought.

The a posteriori estimates given in [117, Sec. 5] are sharp and have been proven good estimates over two decades of applications within exponential integrators. The estimate is given as:

$$\|e^X y - bV_l e^{H_l} e_1\|_2 \leq h_{l+1,l} b |e_l^T \varphi_1(H_l) e_1|,$$

where the function $\varphi_1(\cdot)$ is as given in (5.4) of the first section. As in Section 5.1, we do not directly compute these $\varphi-$ functions. By augmenting the Hessenberg matrix $H_l$ returned from the Arnoldi process,

$$\tilde{H}_l = \begin{bmatrix} H_l & e_1 \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{(l+1)\times(l+1)}, \tag{5.40}$$

we get

$$e^{\tilde{H}_l} = \begin{bmatrix} e^{H_l} & \varphi_1(H_l)e_1 \\ 0 & 1 \end{bmatrix}$$

In our integration scheme, the step (5.15)

$$x_{k+1} = \begin{bmatrix} I_n & 0 \end{bmatrix} e^{h\tilde{A}_k} [x_k^T\ 1]^T$$

is computed using the Krylov subspace method. In each sampling time, software implementations that compute $e^{h\tilde{A}_k}([x_k^T\ 1]^T)$ in [126, 127] iteratively find the order of the Arnoldi approximation $H_l$ until error tolerances are met. Therefore, when $l << n$, the main computational cost is the matrix-vector multiplication computed $l$ times on line 3 of the Arnoldi iteration. The cost of the computation $e^{\tilde{H}_l}$, which is $O((l+1)^3)$, would then be small in comparison to computing the matrix exponential first and then multiplying with a vector. For example, the 'most efficient implementation' in the literature `phipm` [126] uses the scaling and squaring method of [120] with IEEE double precision accuracy to compute $e^{\tilde{H}_l}$ and so has quantifiable a posteriori computational cost bound of $M(\tilde{H}_l)(l+1)^3$ matrix-matrix multiplications, where $M(\tilde{H}_l) := 44/3 + 2\lceil \log_2 \|A\|/5.37 \rceil$. Note that the Krylov subspace method is truly iterative, be-

cause both the computational cost and error bounds can only be computed after evaluating our estimate. Unlike for the method of [113], we cannot find a priori bounds — for example, no useful a priori bounds are known for the order of the Krylov subspace to guarantee some error tolerance.

When $\|h\tilde{A}_k\|$ is large, the Krylov method may end up using a large order subspace. To avoid this, efficient implementations in the literature [126, 127] are adaptive; in addition to the Hessenberg matrix order $l$, they scale $h\tilde{A}_k$ by $\tau$, dividing the sampling time $[t_k, t_{k+1})$ into further $\tau$ time steps. The choice of the Krylov subspace size and scaling steps $1/\tau$ are chosen adaptively online to satisfy error bounds while minimizing the cost of computation. This iterative adaptation does also add to the overhead computational cost in software implementations.

### 5.3.5 Explicit classical Runge-Kutta methods

The relationship between the classical Runge-Kutta approximation and the matrix exponential method has long been known in the integration of linear systems [128]; both rely on a Taylor series expansion. However, whereas the method of [113] is based on a rigourous backward error analysis with a priori bounds, the Runge-kutta method has only asymptotic "$h^p$-type" local truncation error bounds. Looking back at the bilinear dynamics with ZOH (5.13) and taking advantage of the knowledge of the function evaluations within each sampling interval, the action of the classical $4^{\text{th}}$-order RK integrator in a single time-step with stepsize $h$ can be recast as [62, pp. 99][128]:

$$x(t_{k+1}) = x(t_k) + \frac{1}{6}f_1 + \frac{1}{3}f_2 + \frac{1}{3}f_3 + \frac{1}{6}f_4, \tag{5.41}$$

where

$$\begin{aligned} f_1(k) &:= h(\hat{A}_k x(t_k) + Bu(t_k)) \\ f_2(k) &:= h(\hat{A}_k[x(t_k) + \frac{f_1(k)}{2}] + Bu(t_k)) \\ f_3(k) &:= h(\hat{A}_k[x(t_k) + \frac{f_2(k)}{2}] + Bu(t_k)) \\ f_4(k) &:= h(\hat{A}_k[x(t_k) + f_3(k)] + Bu(t_k)) \end{aligned} \tag{5.42}$$

where $\hat{A}_k = A + \sum_{i=1}^{r} A_i u_i(t_k)$. The state at the next step can further be simplified to

$$x_{k+1} = [I + h\hat{A}_k + \frac{1}{2!}(h\hat{A}_k)^2 + \frac{1}{3!}(h\hat{A}_k)^3 + \frac{1}{4!}(h\hat{A}_k)^4]x_k + \frac{1}{6}[I + 4h\hat{A}_k + (h\hat{A}_k)^2 + \frac{1}{4}(h\hat{A}_k)^3]hBu_k \tag{5.43}$$

As a method for general nonlinear ODEs, the above form shows that the Runge-Kutta scheme presented does not take advantage of the special linear structure of the system shown in (5.15). It approximates the integral action of the input as shown above and not exactly as in (5.15). Similarly to Algorithm 3.2 in [113], an efficient implementation of (5.43) would, of course, not compute the terms $(h\hat{A}_k)^i, i = 1,\ldots,4$; iterative matrix-vector multiplications can be used to compute each $f_i$ in (5.41) instead.

Motivated by the need to exploit sparsity and the possibility of using the direct exponential integrator within direct collocation of optimal control problems with bilinear dynamics, we propose a novel equivalent method for integrating bilinear control system ODEs with the direct exponential integrator. For a clearer exposition and comparison, we first consider the classical Runge-Kutta scheme.

## 5.4 Solving Bilinear ODEs via Sparse Linear Systems of Equations

The various implicit and explicit Runge-Kutta methods are popular in direct transcription or in solving ODEs in indirect schemes for optimal control problems; Euler, Trapezoidal and the classical fourth-order Runge-Kutta scheme are a few examples [62]. In the integration of nonstiff dynamical systems, as we have done in Chapter 3, the explicit fourth order scheme (RK from here on) is often preferred since it strikes a good compromise between truncation error requirements and computational cost per step [129].

The local truncation error of a solution from RK is known to be of order $O(h^5)$ [118]. In applications where the required step size of integration $h$ is too big to guarantee the forward relative error tolerance requested by the user, efficient variable step-size implementations like Matlab's `ode45` use adaptive schemes to decrease the steps until local error tolerances are met. In cases where the time step is sufficiently small, a coarser discretization is used with local interpolants. For example, the Dormand-Prince method (implemented as Matlab's `ode45`) uses six function evaluations per step with its interpolation formulae for local error estimates of fifth order. In Chapter 3, we have used the variable step integrator `ode45`. Here, motivated partly by a simpler hardware implementation with cost analysis and partly to make a fair comparison (by avoiding over-

head costs in Matlab's adaptive implementation) with the direct exponential integrator from [113], we consider the fixed step RK.

For the trapezoidal and Hermite-Simpson collocations, [62, Sec. 4.6] discusses sparsity considerations when discretizing ODEs within direct solvers. It is shown there that exploiting grid separability (i.e. separating/concatenating state variables by grid point) results in a sparse structure for Jacobians and Hessians of the resulting grid constraints. In addition, it is also shown that separating linear terms in the constraint functions allows for a more sparse implementation and derivation of the gradient structures for efficient numerical approximations. In a similar spirit, we consider the classical fourth order explicit Runge-Kutta method [62, pp. 133] and present a sparsity preserving collocation method for bilinear dynamics.

**Collocation with RK**

Within what is called a noncondensed collocation, we introduce the local variables of integration $f_i(k), i = 1,\ldots,4$ in (5.41) as additional unknowns rather than eliminate them. If these terms are eliminated, we get grid constraints of the form (5.43), whose right hand side evaluations square the dynamic matrices of the ODE destroying sparsity in the matrices. Therefore, with the explicit expressions:

$$\begin{aligned}
x_{k+1} &= x(t_k) + \frac{1}{6}(y_1(k) + 2y_2(k) + 2y_3(k) + y_4(k)),\\
y_1(k) &= h([A_0 + \sum_{i=1}^{r} A_i u_i(t_k)]x(t_k) + Bu(t_k)),\\
y_2(k) &= hf(x_k + \frac{y_1}{2}, t_k + \frac{h}{2}; \cdot),\\
&= h([A_0 + \sum_{i=1}^{r} A_i u_i(t_k)][x(t_k) + \frac{y_1(k)}{2}] + Bu(t_k)),\\
y_3(k) &= hf(x_k + \frac{y_2}{2}, t_k + \frac{h}{2}; \cdot),\\
&= h([A_0 + \sum_{i=1}^{r} A_i u_i(t_k)][x(t_k) + \frac{y_2(k)}{2}] + Bu(t_{k+1})),\\
y_4(k) &= hf(x_k + y_3, t_k + h; \cdot),\\
&= h([A_0 + \sum_{i=1}^{r} A_i u_i(t_k)][x(t_k) + y_3(k)] + Bu(t_k)),
\end{aligned} \tag{5.44}$$

where $f(\cdot)$ is the right hand side of the bilinear system ODE (5.13), the additional local variables $y_i(k)$ couple the state values at the grid points $t_k$ and $t_{k+1}$. Below, we use the notation $y_{k,i}$ for $y_i(k)$, $i = 1,2,3,4$. For the whole integration horizon, we define a new variable $\boldsymbol{x}$ containing the state variables $x_k$ at the integration nodes and augmented with

the variables $y_{k,i}$. Let

$$\boldsymbol{x} := [x_0^T\ y_{0,1}^T\ y_{0,2}^T\ y_{0,3}^T\ y_{0,4}^T\ x_1^T\ y_{1,1}^T\ y_{1,2}^T \dots, y_{N-1,3}^T\ y_{N-1,4}^T\ x_N^T]^T.$$

Then, the collocation conditions (5.44) over all integration nodes can be represented with the concise notation:

$$\boldsymbol{A}(\boldsymbol{u})\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} - \boldsymbol{b} = 0, \tag{5.45}$$

where $\boldsymbol{u} := [u_0^T\ u_1^T \dots u_{N-1}^T]^T$, $\boldsymbol{b} := [x_0^T\ 0_n^T \dots 0_n^T]^T$, and $\boldsymbol{A}(\boldsymbol{u})$ is linear in the inputs with

$$\boldsymbol{A}(\cdot) := \left[\begin{array}{c|c|c|c|c|c|c|c|c|c|c}
I & 0 & & & & & & & & & \\ \hline
A(u_0) & -I & 0 & & & & & & & & \\ \hline
A(u_0) & \frac{A(u_0)}{2} & -I & 0 & & & & & & & \\ \hline
A(u_0) & 0 & \frac{A(u_0)}{2} & -I & 0 & & & & & & \\ \hline
A(u_0) & 0 & 0 & A(u_0) & -I & 0 & 0 & & & & \\ \hline
I & \frac{I}{6} & \frac{I}{3} & \frac{I}{3} & \frac{I}{6} & -I & 0 & & & & \\ \hline
0 & 0 & 0 & 0 & 0 & A(u_1) & -I & & & & \\ \hline
0 & 0 & 0 & 0 & 0 & A(u_1) & \frac{A(u_1)}{2} & -I & 0 & & \\ \hline
0 & 0 & 0 & 0 & \ddots & \ddots & \ddots & & & & \\ \hline
0 & 0 & 0 & 0 & \dots & A(u_{N-1}) & 0 & \frac{A(u_{N-1})}{2} & -I & 0 & \\ \hline
0 & 0 & 0 & 0 & \dots & A(u_{N-1}) & 0 & 0 & A(u_{N-1}) & -I & 0 \\ \hline
0 & 0 & 0 & 0 & \dots & I & \frac{I}{6} & \frac{I}{3} & \frac{I}{3} & \frac{I}{6} & -I
\end{array}\right] \tag{5.46}$$

$\boldsymbol{B} := \begin{bmatrix} I_N \otimes \bar{B} \\ 0 \end{bmatrix}$, $\bar{B} := \begin{bmatrix} 0 \\ 1_4 \end{bmatrix} \otimes hB$, $A(u_k) := h(A + \sum_{i=1}^{r} N_i u_i(t_k))$, $h$ is the integration step, $\otimes$ denotes the Kronecker product and $1_n$ stands for a column vector of ones of length $n$.

Assuming the number of control actions to be the same as the number of integration steps, we have $\boldsymbol{u} \in \mathbb{R}^{mN}$ and the augmented variable $\boldsymbol{x} \in \mathbb{R}^{(5N+1)n}$. In reality, this is not always true. The length of the ZOH can often be limited by control actuation considerations and can be many times the integration step; to include such cases, we represent the number of control actions by $M$, i.e. $\boldsymbol{u} \in \mathbb{R}^{n_u}, n_u = mM$, $\boldsymbol{x} \in \mathbb{R}^{n_x}, n_x = (1+5N)n$ . For equispaced integration nodes, we represent by a positive integer $M_N$, the ratio of the number of integration steps $N$ to the number of ZOH intervals (or control actions) $M$ is given by $M_N := N/M$; see Figure 5.2 for an illustration.

Although the matrices of the linear equation in (5.45), $\boldsymbol{A} \in \mathbb{R}^{n_x \times n_x}$ and $\boldsymbol{B} \in \mathbb{R}^{n_x \times n_u}$, can potentially grow to very large sizes for $N \gg n, m$, their elements are mostly zero . All the sparsity in $A(u_k)$ and $B$ are preserved in this noncondensed formulation because there is no squaring of matrices.

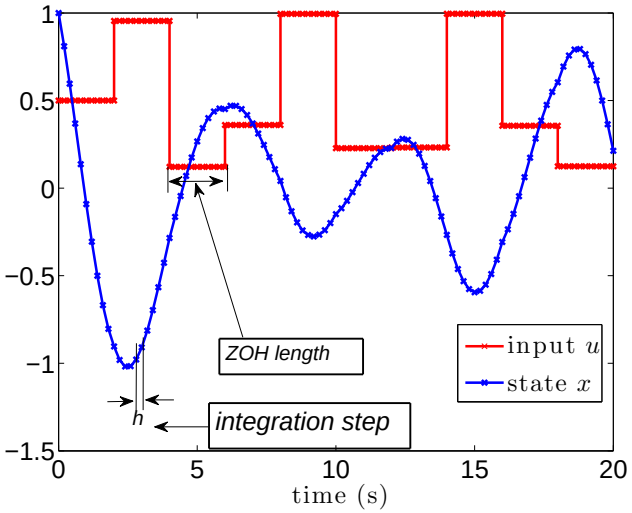



Figure 5.2: The ZOH length can be many multiples of the integration step length. A snippet from a WEC simulation showing the velocity ($x$ here) against bilinear controller ($u$ here): $M_N = 100/10 = 10$.

As a result, the size of the nonzero elements grows only linearly with the number of steps $N$. Let the notation $\mathrm{nnz}(X)$ stand for the number of nonzero elements of matrix $X$, then we have:

$$\mathrm{nnz}(\boldsymbol{A}) \leq (7n_A + 10n)N + n, \tag{5.47}$$

$$\mathrm{nnz}(\boldsymbol{B}) = 4N\,\mathrm{nnz}(B), \tag{5.48}$$

where $n_A = \max_u \{\mathrm{nnz}(A(u)) : u \in \mathbb{U}\}$.

This sparsity preservation is important in model predictive control of systems with fast processes where fast sampling rates mean $N = (t_f - t_0)/h$ is very large. Another area is where the system matrices $A, N_i$ come from the discretization of a PDE and often are very large and sparse. In both cases, it is also vital to note that memory requirements in hardware implementation do not grow as we only need to store the smaller matrices $A, N_i \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times m}$ and the input sequence $\boldsymbol{u}$. Moreover, as can be seen in Figure 5.3, the linear system matrix $\boldsymbol{A}$ is block diagonal and lower triangular. This example plot is from a direct RK collocation of the wave energy system where the system dynamic matrix $A$ relatively full (has $\approx 78\%$ nonzero elements); even in this case $\boldsymbol{A}$ shows a sparsity of $2.7\% \approx 1765/(255^2)$.

### Collocation with the direct exponential integrator

In the last subsection, we have derived a noncondensed direct transcription scheme for optimal control problems with bilinear dynamics using the classical Runge-Kutta collocation scheme. By appending the states at consecutive grid points by using auxiliary local variables, we have derived a larger sparse problem with sparse gradient structures. We show here that a similar sparse matrix representation can be formulated for the dynamic constraints using the direct ex-

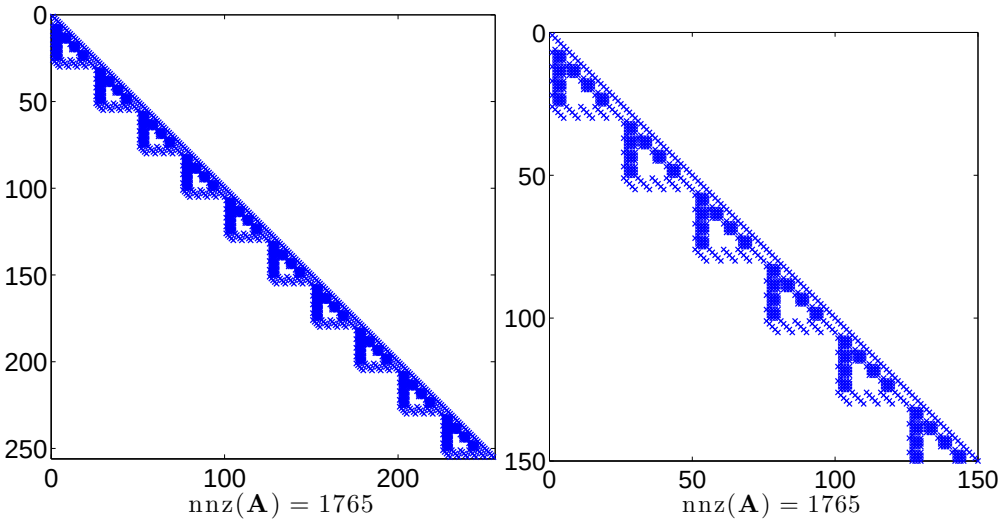



Figure 5.3: The sparsity structure of $\boldsymbol{A}(\cdot)$ in (5.45): wave energy system model simulated with parameters $n = 5$, $m = 2$, $\text{nnz}(A) = 18$; $N = 10$, $M = 2$

ponential integrator of Algorithm 3.

In solving the bilinear dynamics using the exact integration formula (5.15), the direct exponential scheme is used to compute the exact evolution between the integration nodes $t_k$ and $t_{k+1}$

$$x(t_{k+1}) = \begin{bmatrix} I_n & 0 \end{bmatrix} e^{h\tilde{A}(u_k)} \begin{bmatrix} x_k \\ 1 \end{bmatrix}, \tag{5.49}$$

where $x_k = x(t_k)$ and $\tilde{A}(u_k) = \begin{bmatrix} \hat{A}_k & Bu_k \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}$, $\hat{A}_k = [A + \sum_{i=1}^{r} N_i u_i(t_k)]$.

Although the optimal Taylor expansion order $l$ and scaling $s$ required to integrate the dynamics with some required error bound guarantees can be computed for each interval and corresponding control input, we opt to use the best upper bound computed a priori (see Section 5.3.3 for reference) for all intervals; as before we assume constant integration steps here. With this in mind, we formulate a non-iterative matrix representation for the direct exponential integration method. This way, integration of a bilinear system is reformulated to solving a linear equation. The resulting linear system can be solved using an appropriate linear solver.

Looking back at the iterations of the direct method for finding the action of the matrix exponential on a vector (5.18), we can re-write the process for computing $e^X y$ as

$$b_{i+1} = T_l(s^{-1}X)b_i, \tag{5.50}$$

$$= \sum_{j=1}^{l} \frac{(s^{-1}X)^j}{j!} b_i, \tag{5.51}$$

for $i = 0, \ldots, s-1$, with $b_0 = y,\ b_s \approx e^X y$.

Now, let's define the new auxiliary variables based on the intermediate variables generated in the iterations of Algorithm 3

$$g_{i,j} \quad := \quad \frac{X}{sj} g_{i,j-1},\ g_{i,0} = b_i,\ j = 1, \ldots, l,\ \ i = 0, \ldots, s-1, . \tag{5.52}$$

We then have from (5.51) and (5.52),

$$b_{i+1} \quad = g_{i,0} + g_{i,1} + \ldots + g_{i,l} \tag{5.53}$$

By introducing additional auxiliary variables for $b_i$ transitions, we can preserve block diagonality in the matrix representation of the resulting collocation contraints and subsequently the sparsity patterns of gradients of these constraints. Let

$$f_{i,j} := f_{i,j-1} + g_{i,j},\ j = 1, \ldots, l,\ i = 0, \ldots, s-1, \text{ with } f_{i,0} := g_{i,0} := b_i. \tag{5.54}$$

Then,

$$b_{i+1} = f_{i,l}, \text{ and } b_s = f_{s,l} \approx e^X y. \tag{5.55}$$

The iterations can be then be represented as a linear equation $\hat{X}\hat{y} = 0$, where $\hat{X} \in \mathbb{R}^{2sln \times (2sl+1)n}$ is

$$\left[\begin{array}{c||c|c|c|c|c|c|c|c|c|c|c|c}
X/s & -I & & & & & & & & & & & \\ \hline
I & I & -I & & & & & & & & & & \\ \hline
0 & X/(s2) & 0 & -I & & & & & & & & & \\ \hline
0 & 0 & I & I & -I & & & & & & & & \\ \hline
0 & 0 & 0 & X/(s3) & 0 & -I & & & & & & & \\ \hline
0 & 0 & 0 & 0 & I & I & -I & & & & & & \\ \hline
0 & 0 & 0 & 0 & 0 & \ddots & & & & & & & \\ \hline
0 & 0 & 0 & 0 & 0 & \ldots & X/(sl) & 0 & -I & & & & \\ \hline
0 & 0 & 0 & 0 & 0 & \ldots & 0 & I & I & -I & & & \\ \hline\hline
0 & 0 & 0 & 0 & 0 & \ldots & 0 & 0 & 0 & X/s & -I & & \\ \hline
0 & 0 & 0 & 0 & 0 & \ldots & 0 & 0 & 0 & \ddots & \ddots & & \\ \hline
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \ldots & 0 & I & I & -I
\end{array}\right]_{\hat{X}} \tag{5.56}$$

We can approximate $e^X y$ by solving $\bar{X}\bar{y} = \bar{b}$, where $\bar{X} = \hat{X}(:, n+1 : 2sln+n) \in \mathbb{R}^{2sln \times 2sln}$, $\bar{y} \quad = \quad \left[\begin{array}{cccccccccccc} g_{0,1}^T & f_{0,1}^T & g_{0,2}^T & \cdots & f_{0,l-1}^T & g_{0,l}^T & f_{0,l}^T & g_{1,0}^T & \cdots & g_{s,l}^T & f_{s,l}^T \end{array}\right]^T$ and $\bar{b} = -\hat{X}(:, 1:n)y \in \mathbb{R}^{2sln}$.

In collocating a bilinear system over $N$ intervals, $X$ in (5.56) is replaced by $\tilde{A}(u_k) \in \mathbb{R}^{(n+1)\times(n+1)}$ in (5.49). Similarly to what we did for the RK method, by concatenating the local variables $\bar{y}$ and therefore the linear equation blocks $\bar{X}$ for consecutive integration intervals, we end up with a bigger linear system of the form

$$\boldsymbol{A}(\boldsymbol{u})\boldsymbol{x} + \boldsymbol{b}(x_0) = 0, \tag{5.57}$$

where $\boldsymbol{x} \in \mathbb{R}^{n_x}$, $n_x = (n+1)[N(2sl+1)-1] \approx N(2sl+1)(n+1)$, is the concatenation of the local integration variables over the whole integration interval, $l$ and $s$ are the bounds on the Taylor approximation order and scaling for a given error tolerance. Similarly to the Runge-Kutta collocation, $\boldsymbol{u} := [u_0^T\ u_1^T \ldots, u_{M-1}^T]^T \in \mathbb{R}^{n_u}$, $n_u := mM$.

In comparing the size of the resulting linear problems in the RK and exponential methods, the size of $\boldsymbol{u}$ is the same. The difference is in $n_x$; the relative terms to compare are $5n$ and $(n+1)(2sl+1)$. Since $l, s \geq 1$, the exponential collocation scheme can result in a smaller linear system by at most a factor of $5/3$. The vector $\boldsymbol{b}(x_0)$ is mostly zero with $\text{nnz}(\boldsymbol{b}(x_0)) = \text{nnz}(x_0) + (N-1)$. It can be shown here that the number of nonzero elements in it can be bounded by

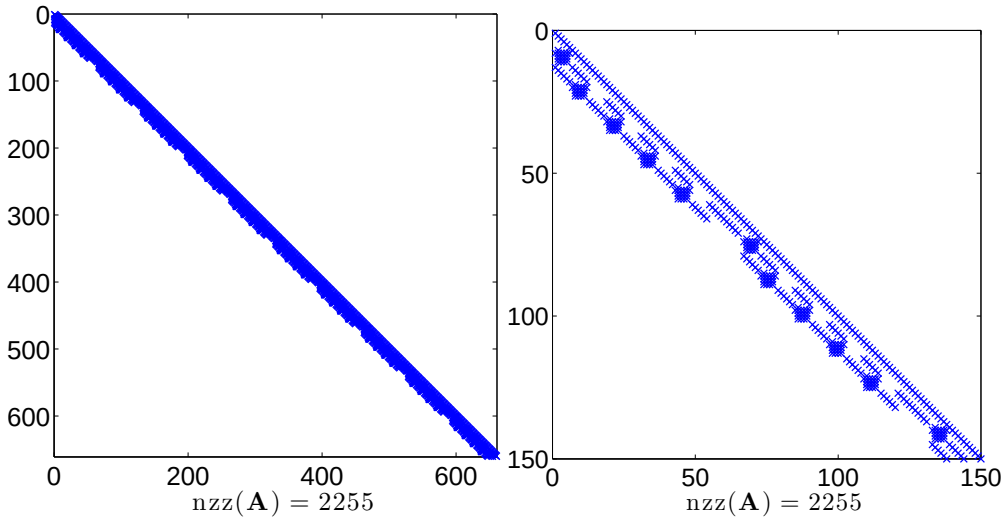
$$\text{nnz}(\boldsymbol{A}) \quad \leq \quad (sln_A + (4sl+1)(n+1))N, \tag{5.58}$$

where $n_A := \max_u \{\text{nnz}(A(u)) : u \in \mathbb{U}\}$, $A(u) := [A + \sum_{i=1}^{r} N_i u_i]$.

Therefore, comparing (5.58) with the corresponding bound for the Runge-Kutta method (5.47), the sparsity for the two methods can be bounded (approximately) by $\frac{(7n_A+10n)N+5n}{n^2(1+5N)^2}$ and $\frac{(sln_A+(4sl+1)n)}{N(n+1)^2(2sl+1)^2}$, respectively. Therefore, in settings when $2sl+1$ is much larger than 5, we are guaranteed to get a much more sparser problem. Moreover, for a sufficiently large $N$ (i.e. sufficiently small integration step $h$), the direct exponential collocation can even be both sparser and smaller in size.

In Figure 5.4 we depict the sparsity structure of the linear system matrix $\boldsymbol{A}(\cdot)$ using the same wave energy example shown in Figure 5.3 and with the same settings; all these examples have an integration step of $h = 0.1$ seconds. In this simple setting, the exponential discretization results in a larger sized linear equation (compare 660 with 255, i.e. roughly 2.5 times bigger). However, it is much sparser — has sparsity of $2255/660^2 (\approx 0.5\%)$ compared to 2.7% sparsity for RK.

A very important aspect of both collocations is that they result in sparse lower triangular systems with unity on the diagonal. These sparse nonsingular matrices are the easiest/cheapest to solve [89, Sec. C.2]. In fact, looking at both collocation schemes, we can bound the number of flops ( floating point operations) required in solving them using the number of integration steps and the sparsity of the original bilinear system matrices. For a given input sequence $\boldsymbol{u}$, it can be shown that the total cost in flops of solving the linear equation for the Runge-Kutta collocation




(a) (b) zoom on (a)

Figure 5.4: The sparsity structure of $\boldsymbol{A}(\cdot)$ in (5.57): wave energy system model simulated with parameters $n = 5$, $m = 2$, $\text{nnz}(A) = 18$; $N = 10$, $M = 2$

is (see Appendix D):

$$\text{cost}_{RK} \quad \leq \quad (\frac{14n_A}{5} + 3n)(1 + 5N). \tag{5.59}$$

Similarly, for the exponential collocation, the bound on the cost can be shown to be

$$\text{cost}_{EXP} \quad \leq \quad (n_A + 4n + 3)(2sl + 1)N \tag{5.60}$$

flops.

**Algorithm parallelization considerations**

In a serial implementation, the bounds on the CPU time (or time complexity) to solve the two linear systems will be proportional to the respective bounds on the number of flops. In this case, time complexity is measured by the total number of flops required by the algorithm. However, with the recent ease in the availability of hardware for parallel computations, time complexity can be reduced significantly. The study of architectures for parallel computing on (multi-core) graphics processors (GPUs), field-programmable gate arrays (FPGAs) and multi-core CPUs is an active research area [130]. Although we are not going to thoroughly discuss parallelization, we will point an obvious way that time-parallelism can be integrated to solve the bilinear system ODEs we have considered in even less time.

Consider the forward substitution used in solving the lower triangular matrices in (5.45) and (5.57). As usual, we assume that all arithmetic operations take identical time and regard communication between blocks as instant [130]. Then, if all arithmetic operations per each stage of the forward substitution can be parallelized (i.e. we have sufficient parallel resources), the time complexity in

terms of flop operations is equal to $2n_x - 1$ [131], where $n_x$ is the size of the matrices $\boldsymbol{A}(\cdot)$. Therefore, for our methods of integration via the solution of large linear systems, the time complexity can be reduced significantly (roughly by a factor of $n_A/n$); it becomes directly proportional to the state size $n$ and the number of integration steps $N$.

## 5.5 Example Simulations with Bilinear Systems

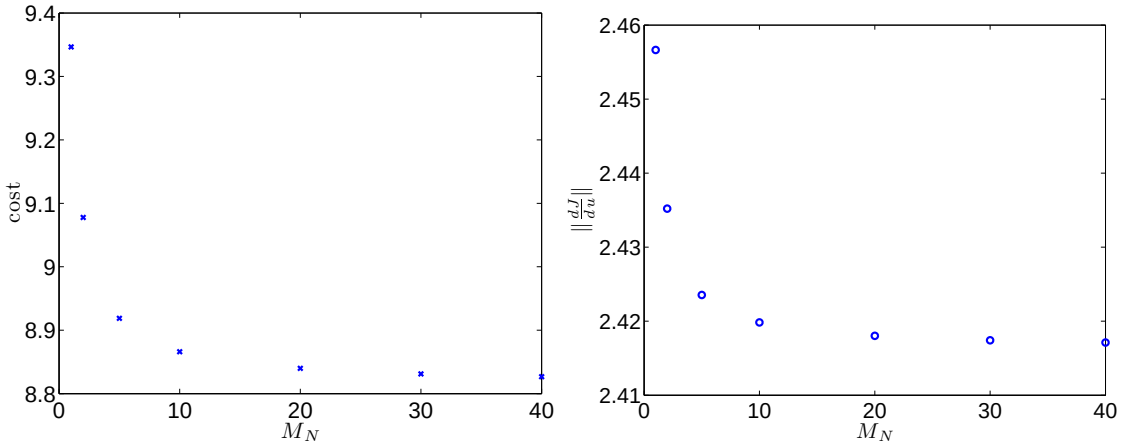
**Example 1: Optimal Control of a Wave Energy Converter**

The fist example we consider here is the optimal control of the wave energy converter discussed in Chapter 3. The indirect optimal control scheme expends its main computational cost in the projected gradient method in Algorithm 1, where the bilinear system dynamics with piece-wise constant inputs is integrated forward. The other computation involves the integration of the adjoint bilinear dynamics backwards in time. From here, all computations were performed on a 2.4 GHz Intel Core 2 Duo CPU machine in MATLAB version 2011a.

For this problem, we consider appropriate integration step sizes $h$ for a given ZOH in the input discretization. We can then empirically determine, in each case, how much finer the state has to be resolved to approximate the objective function and its gradient sufficiently well in the projected gradient scheme. From the experiments reported in Figure 3.8, it is known that extracted power flattens around a prediction horizon of two typical wave periods. For the typical periods ($T_p$) considered, we look at prediction horizon of about 12 seconds for the system reported in that same figure. With a ZOH length of 0.1 seconds as an example, Figures 5.5b and 5.5a show the convergence of the cost function (3.8) and the norm of the gradient of the cost (3.22), respectively, of a typical iteration as a function of the $M_N$— the ratio of the input sampling time to the integration step $h$. From these figures and as was observed in many simulations, it can then safely be taken that resolving the state about 5 to 10 times more finely is then good enough.

The local truncation error for the fourth-order method can be shown to be of order $O(h^5)$ [118]. Therefore, the RK method can outperform the matrix exponential based integrators when the time-steps required for the given forward error tolerance are of the same order or bigger than the sampling time for the numerical solution required by the user. However, in some applications the required stepsize of integration $h$ may be too big to guarantee the forward relative error tolerance requested by the user. In that case, highly efficient implementations like Matlab's `ode45` use adaptive schemes to decrease the steps until local error tolerances are met. This may mean the use of many more steps by the RK method than an exponential integration scheme. In such cases, the exponential integrators will be computationally superior. However, depending on the problem, actual errors should be compared a posteriori as well.

Figure 5.6 shows a typical plot of errors in the norm of the solution against computational cost in `CPU time`. As the 'latest efficient variation' of the Krylov subspace method, we use the Matlab

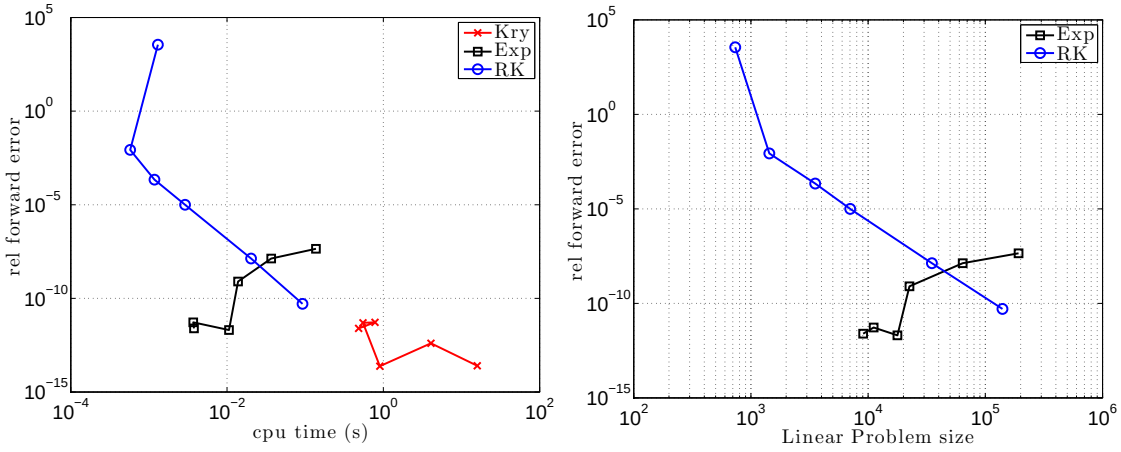



(a) (b)

Figure 5.5: Convergence of cost and gradient with integration step: (a) cost function (b) norm of gradient of cost with input sampling time of 0.1 and $T_h = 12$.

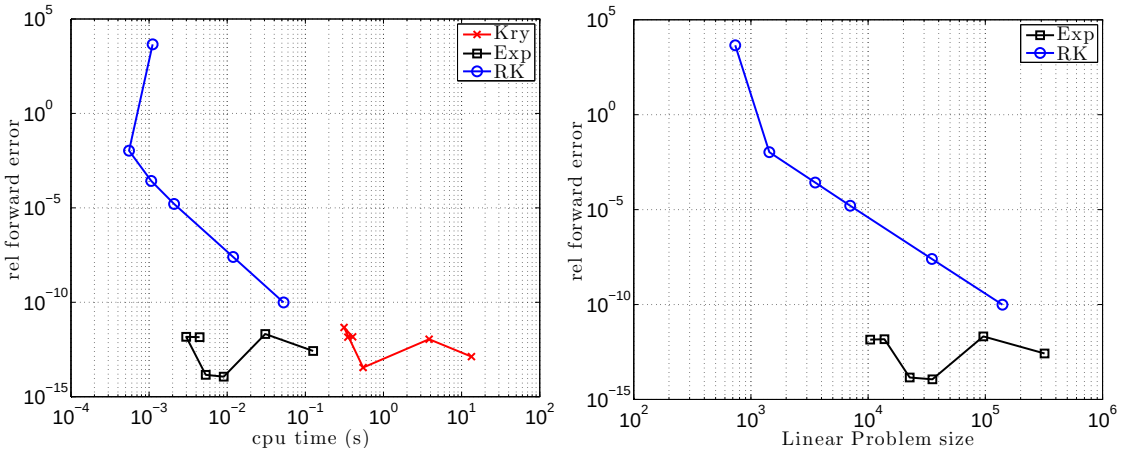
function `expmvp` from [126] to compute the integration in each sampling time. It was used with an error tolerance set to $tol = 2^{-14} \approx 6.1 \times 10^{-5}$ and the maximum possible Krylov subspace dimension in the adaptive scheme was set to 10. We also use the same tolerance for the method adapted from [113] and used with our $\alpha_p(\cdot)$ bound in (5.37). Since we do not have an analytic solution for the bilinear system, the global forward error of numerical solutions are computed using a solution computed using Matlab's `expm` with a very high accuracy; we set the tolerance to $2^{-53}$. This can safely be taken as closest to the real solution since numerical solutions from all methods converge to this in the limit; what we have labelled `rel forward error` on the following plots refers to relative forward error to this solution. A similar performance profile is seen for the backward integration.

From Figure 5.6a and Figure 5.7a, it is apparent that both exponential integration schemes (Kry and EXP) give highly accurate solutions. The data points, from left to right, represent integration of the dynamics over a 20 second prediction horizon with decreasing sampling intervals, $h = [1, 0.5, 0.2, 0.1, 0.02, 0.005]$. The ZOH length is set to 5 times the sampling time for the state. The Runge-Kutta scheme exhibits an order 4 global error convergence with sampling time as expected. The exponential integrators, on the other hand, have errors clustered around well below the tolerances set for the solution. Since CPU time may depend on implementation, we also show in Figures 5.6b and 5.7b the size of the linear problems solved to compute the integrals (this depends linearly on the number of matrix-vector multiplications for the direct EXP and RK methods). The results demonstrate that the RK method requires much finer integration steps to satisfy a given error tolerance. However, the exponential integrators satisfy the required tolerances at all given step sizes. Comparing Figures 5.6a and 5.7a, the EXP method shows a significant improvement in accuracy with a marginal cost in computations as the tolerance is decreased to IEEE single precision roundoff. It can be concluded here that depending on the ac-




(a) (b)

Figure 5.6: Logarithmic plots of the forward error of the solution against computational cost of three numerical methods in solving the bilinear system. The keys 'KRY', 'EXP', and 'RK' stand for the Krylov integrator (using the algorithm `expmvp` from [126]), adapting the method of [113] and the $4^{\text{th}}$ order Runge-Kutta method, respectively: (a) the $x-$axis shows total CPU time of integration (b) the size $n_x$ of the linear systems solved by the direct EXP and RK methods are shown on the $x-$axis.




(a) (b)

Figure 5.7: A repeat of the experiment in Figure 5.6 with the tolerance for 'EXP' set to IEEE single precision eps ($2^{-24}$) and all other parameters the same.

curacy required, the method EXP can be computationally much cheaper than RK. Alternatively, for a given computational resource, if the sampling steps are not too small, the direct exponential method can give far better accuracies compared to RK.

Finally, all the integration schemes were applied to the wave energy problem within a receding horizon controller. With a setting of prediction horizon $T_h = 12$s, integration step $h = 0.01$s (i.e. $N = 1200$), $M_N = 10$ (ZOH interval of 0.1s), the device model was simulated in closed-loop

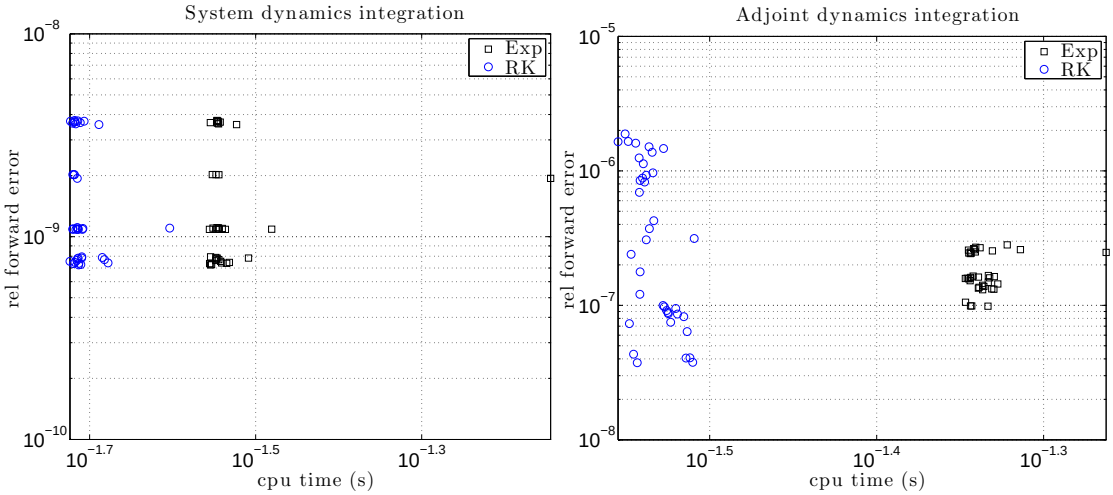



Figure 5.8: Computational cost of integration schemes in a closed-loop simulation.

control. The exponential collocation was used with an error tolerance equal to IEEE half precision. As would be expected from the results in Figure 5.6b, the exponential scheme resulted in a bigger problem size (8000 NLP variables compared to 2705 for the Runge-Kutta collocation). Figure 5.8, shows a scatter diagram of cpu time per gradient projection scheme iteration for the RK and EXP methods implemented within a receding horizon control. Under these settings, it can be noted of course that EXP is roughly twice as expensive in solving both forward and adjoint dynamics for roughly a similar forward error. From the simulation results, we may conclude that for the wave problem and at this sampling frequency, it may be beneficial to use the RK scheme.

**Example 2: A Heat Transfer Model for Cooling a Metal Slab**

The second example we consider is a bilinear control system that arises from the discretization of a controlled PDE — a model for heat transfer in a metal slab whose different surfaces are selectively cooled in a rolling mill [132, 133]. The first reference [132] is concerned with model reduction of large scale bilinear systems, while the latter [133] investigates LQR control of the linear system approximation. Assuming a sufficiently long (or infinitely long) slab relative to its width and height, the heat distribution along the length axis is considered stationary. Considering the state as the temperature $\theta$ at each point $\zeta$ of the 2-dimentional cross sections of the slab $\Omega \in \mathbb{R}^2$, the evolution of the heat distribution over time $t$ is modelled as

$$
\begin{aligned}
c\rho\theta_t(\zeta,t) &= \lambda\nabla^2\theta(\zeta,t),\ \forall\zeta\in\Omega,\ \forall t\in[0,T], && (5.61)\\
\lambda\theta_\nu(\zeta,t) &= \kappa_j(\theta(\zeta,t)-\theta_{ext,j}),\ \forall\zeta\in\Gamma_j, j=1,2,3,4\ \forall t\in[0,T], && (5.62)\\
\theta(0,\zeta) &= \theta_0(\zeta),\ \zeta\in\Omega, && (5.63)
\end{aligned}
$$

where the material parameters $\rho$ (specific density of the metal), $\lambda$ (heat conductivity of the

metal), $c$ (heat capacity of the metal) are assumed to be constant in $\Omega$ at temperatures of over 700°C. At the boundaries, the heat transfer coefficient $\kappa_j$ in the Robin boundary conditions determine the heat flow normal to the corresponding boundaries $\Gamma_j$; $\theta_{ext,j}$ is the external temperature at boundary $j$.

In practice, cooling at the surface is achieved by spraying cooling fluid. Considering quenching in gas, for example, the intensity of the spraying nozzles can be taken as the control input; $u_j = u_{rate,j}\kappa_j$. Considering a steel slab with square cross-sections and appropriate parameter values and scaling as in [133], we use the case where two of the boundaries are insulated (no spraying, $\kappa_j = 0$) and the other two boundaries are cooled using two independent controls. On the unit square $\Omega = [0,1] \times [0,1]$, the resulting bilinear controlled system can be written as [132]

$$\begin{aligned} x_t(\zeta,t) &= a\nabla^2 x(\zeta,t),\ \forall \zeta \in \Omega,\ \forall t \in [0,1], && (5.64)\\ \nu \cdot \nabla x(\zeta,t) &= b_1 u_1(t)(x(\zeta,t)-1),\ \forall \zeta \in \Gamma_1 := 0 \times (0,1), && (5.65)\\ \nu \cdot \nabla x(\zeta,t) &= b_2 u_2(t)(x(\zeta,t)-1),\ \forall \zeta \in \Gamma_2 := (0,1) \times 0, && (5.66)\\ x(\cdot,t) &= 0 \ \text{ on } \Gamma_3 := 1 \times [0,1] \text{ and } \Gamma_4 := [0,1] \times 1, && (5.67) \end{aligned}$$

where $a$, $b_1$ and $b_2$ are coefficients resulting from the scaling the problem; $\nu$ is the vector normal to the boudarties and $u_1,\ u_2 \in [0,1]$. Discretizing this PDE using finite differences and a $k \times k$-mesh, we redefine the state of the finite dimensional system as the vector of temperature at the nodes $(i,j)$, $x = \text{vec}(x_{i,j})$. Then, including the Robin and Dirichlet boundary conditions we get the bilinear system

$$\dot{x}(t) = \alpha A x(t) + \beta_1 u_1(t) N_1 x(t) + \beta_1 u_2 N_2 x(t) + Bu(t), \tag{5.68}$$

where $\alpha$, $\beta_1$, and $\beta_1$ are functions of the scaling process using the parameters of the metal shown in [133].

$$\begin{aligned} A &:= \frac{1}{h_\zeta}(I \otimes F_k + F_k \otimes I + E_1 \otimes I + I \otimes E_k),\ E_i := e_i e_i^T \in \mathbb{R}^{k^2 \times k^2},\\ N_1 &:= \frac{1}{h_\zeta} E_1 \otimes I,\ N_2 := \frac{1}{h_\zeta} I \otimes E_k,\\ B &:= [b_1\ b_2],\ b_1 := \frac{1}{h_\zeta} e_1 \otimes e,\ b_2 := \frac{1}{h_\zeta} e \otimes e_k,\ e := [1,\ldots,1]^T \in \mathbb{R}^k, \end{aligned}$$

where $\otimes$ denotes the Kronecker product, $e_i \in \mathbb{R}^k$ is the $i^{th}$ column of the identity matrix and

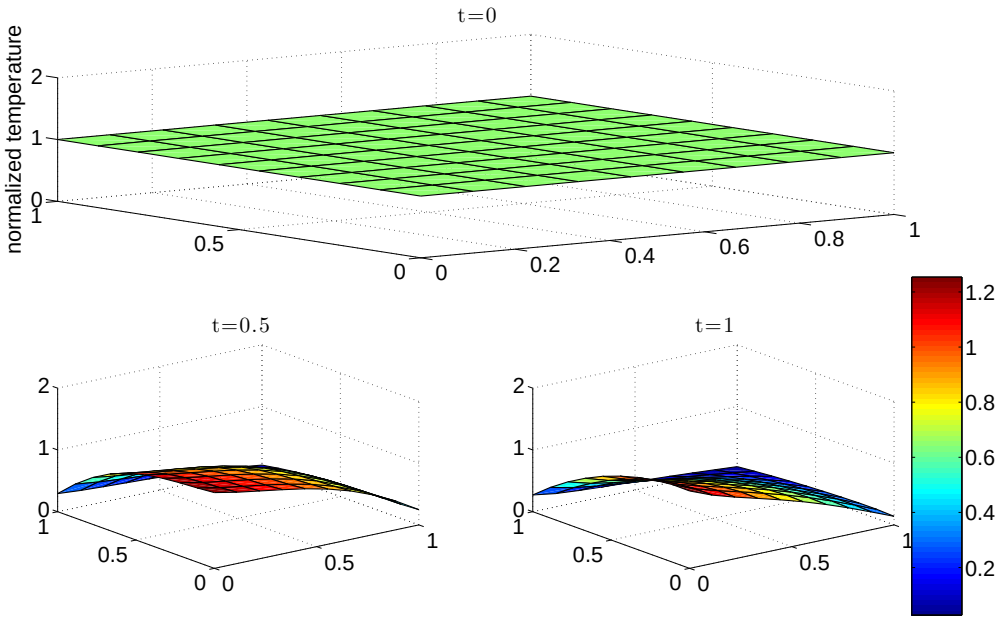
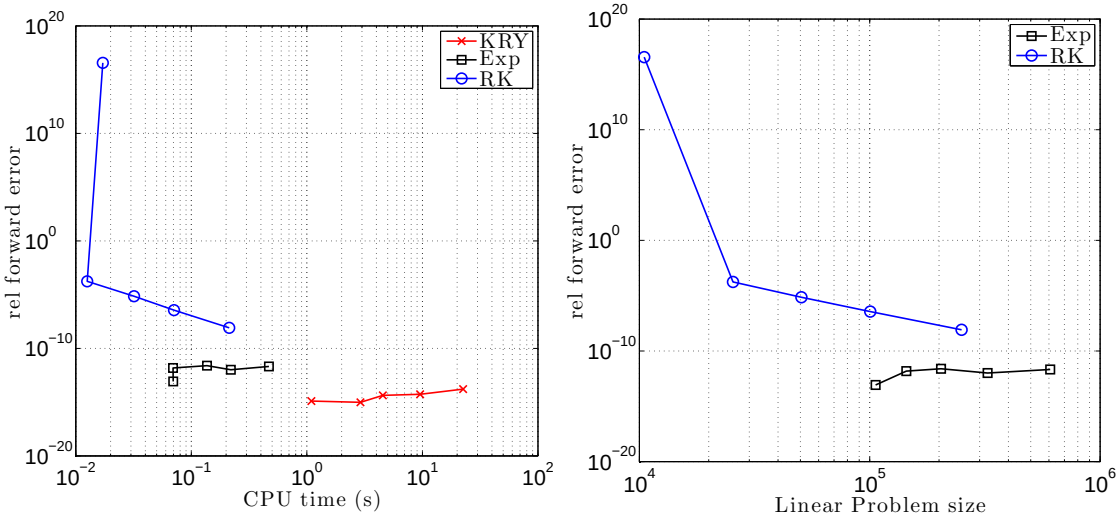



Figure 5.9: Simulation profiles of a metal slab with two sides being cooled by two different control inputs.

$F_k \in \mathbb{R}^{k\times k}$ is the finite differencing matrix

$$F_k := \begin{bmatrix} -2 & 1 & & & & \\ 1 & -2 & 1 & \ddots & & \\ & & & \ddots & \ddots & 1 \\ & & & & 1 & -2 \end{bmatrix}.$$

With an initial uniform temperature of 800°C and external room temperature of 20°C, the scaled system was simulated using the three integration schemes discussed. A $10\times 10$ mesh was used; i.e. $n = k^2 = 100$, with sinusoidal inputs $u_1(t) = (1+\cos(2\pi t))/2$ and $u_2(t) = (1+\sin(2\pi t))/2$. Figure 5.9 shows the evolution of the temperature profile at three points for $t \in [0,1]$. Using tolerance $tol = 2^{-24}$, Figure 5.10 shows the error plot against computational cost for various integration steps; $M_N = 5$, and $h = 1/[20,\ 50,\ 100,\ 200,\ 500]$ from left to right. It can be noted that the exponential integrators are stable for all integration steps whereas the RK scheme is unstable when the time steps are too coarse. This is because EXP scales the problem automatically by using a larger scaling $\bar{s}$ whenever the time steps are too coarse for the required error tolerance. Similarly, the Krylov subspace method implemented in `expmvp` from [126] is adaptive; and scales $h\tilde{A}_k$ by dividing each integration step further till the a posteriori error estimates are good enough. Again, EXP can give better error guarantees at a competitive cost if the required error tolerances are sufficiently small for a given sampling time. However, for very small sampling times, the RK scheme can give similar accuracies as EXP, albeit at a higher cost.




(a) (b)

Figure 5.10: Forward error of solution against computational cost of three numerical methods in solving the controlled metal cooling bilinear system. (a) the x-axis shows total CPU time of integration (b) the size of the linear systems solved by the direct EXP and RK methods are shown on the x-axis.

We have shown that the matrix exponential based integrator from Al-Mohy-Higham [113] can be used for computationally efficient and precise integration of bilinear systems. Similarly to the method of [113], the Krylov subspace method can be used to compute the action of a matrix exponential within an exponential integrator with similar accuracy. However, the Krylov solver is a truly iterative method and can only give a posteriori error estimates. The method of Al-Mohy-Higham, in addition to giving a priori error bounds, is a direct method.

In this section, we have also proposed and analysed the use of this direct exponential integrator and the classical Runge-Kutta methods for solving initial and final value problems for a bilinear control system. We have shown that these integrations can be performed by solving a single sparse lower triangular system that is also nonsingular and block diagonal. The methods were then used for the solution of forward and adjoint dynamics within the indirect control algorithm for the wave energy system. Direct collocation with an interior point method was observed to perform computationally relatively worse for the nonconvex optimal control problem in Chapter 3. In the next section, we present a novel direct transcription scheme using the direct exponential integrator for problems where direct transcription may be effective. We will present example quadratic control problems with bilinear dynamics.

## 5.6 Exponential Collocation for Direct Transcription of Quadratic Optimal Control Problems with Bilinear Dynamics

Within a receding horizon scheme for optimal control of linear systems, one of two different direct collocation approaches can be used; the condensed and noncondensed approaches [48, 134]. For example, if we consider the dense approach, the state can be eliminated resulting in a small dense optimization problem (with the input sequence as the only optimization variable and) with linear constraints as functions of the input only. This, however, is not possible for a bilinear system as we will demonstrate below. For example, let us consider the system of (5.10) with a scalar input $u(t) \in \mathbb{R}$ and the $N$–length ZOH input sequence $\{u_0, u_1, \ldots, u_{N-1}\}$ in some finite horizon $N$. In the simplest case, we assume we use an Euler discretization for the system dynamics:

$$\dot{x}(t) = Ax(t) + u(t)Nx(t) + Bu(t), \tag{5.69}$$

$$x_{k+1} = (I + hA + hNu_k)x_k + hBu_k, \tag{5.70}$$

where $h$ is the stepsize of integration and $u_k \in [-1,1]$, $k = 0, \ldots, N-1$.

From (5.70), we can form the iterations

$$\begin{aligned}
x_1 &= (I + hA + hNu_0)x_0 + hBu_0 \\
x_2 &= (I + hA + hNu_1)x_1 + hBu_1 \\
x_2 &= (I + hA + hNu_1)[(I + hA + hNu_0)x_0 + hBu_0] + hBu_1 \\
x_2 &= r_0(x_0,h,A,N,B)u_0 + r_1(x_0,h,A,N,B)u_0u_1 \\
&\quad + r_2(x_0,h,A,N,B)u_1 + r_3(x_0,h,A,N,B) \\
&\vdots \\
x_N &= r_N(x_0,h,A,N,B)\prod_{i=0}^{N-1} u_i + \ldots,
\end{aligned} \tag{5.71}$$

where $r_i(\cdot)$ are functions of the parameters $x_0, h, A, N$ and $B$.

Unlike for linear systems, in eliminating the states we do not end up with a linear equation (or linear equality constraint) in the optimization variables $\{u_0, u_1, \ldots, u_{N-1}\}$. The parameter condensation results in a representation of the final state as an $Nth$ order polynomial of the optimization variables $\{u_0, u_1, \ldots, u_{N-1}\}$. This does not allow for a subsequent convenient condensed optimization problem; the equality constraint of the resulting NLP becomes a very high order polynomial whose order scales with the sampling frequency and so the constraint and its gradients are not tractable for numerical solution. Therefore, the viable option for bilinear systems is to use a noncondensed representation where both the state and input values are taken as optimization parameters. Here we present a noncondensed direct transcription of bilinear systems using the classical Runge-Kutta scheme and the direct exponential scheme we have

developed. We also formulate the transcription so as to exploit any sparsity in the differential equations of the optimal control problem.

As discussed in Section 3.3.5, in solving an NLP constructed by the discretization of an optimal control problem, first and second derivatives of the constraint and the objective functions are required by the solver. Within the optimization software, these are often numerically approximated using finite differences or are analytically computed. For the trapezoidal and Hermite-Simpson discretizations, [62, Sec. 4.6] discusses sparsity considerations. It is shown there that exploiting grid separability (i.e. separating/concatenating state variables by grid point) results in a sparse structure for Jacobians and Hessians of the constraints. In addition, it is also shown that separating linear terms in the constraint functions allows for a more sparse implementation.

For the RK scheme, consider the collocation in (5.45) again; we had kept the local variables $y_i(t_k)$ as extra unknowns to get a sparse linear system. In a direct transcription, we can keep them as NLP variables resulting in the equality constraint

$$g(\boldsymbol{z}) := \boldsymbol{A}(\boldsymbol{u})\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} - \boldsymbol{b} = 0, \tag{5.72}$$

where the augmented NLP variable is now $\boldsymbol{z} := \begin{bmatrix} \boldsymbol{u}^T & \boldsymbol{x}^T \end{bmatrix}^T$, with size $n_z := mM + (1+5N)n$ and all variables are as defined in (5.45).

Although the number of equality constraints $n_x$ and the matrices $\boldsymbol{A} \in \mathbb{R}^{n_x \times n_x}$ of the equality constraint (5.72) can potentially grow to very large sizes with the number of integration steps $N \gg n, m$, we have shown that they stay sparse and that computational complexity grows only linearly with $N$; see (5.59).

We have seen in Section 3.3.5 that an IPM requires the Jacobian and Hessian of the constraints in solving the nonlinear KKT optimality conditions (3.32). For the sparse RK transcription, with the NLP variable $\boldsymbol{z} := \begin{bmatrix} \boldsymbol{u}^T & \boldsymbol{x}^T \end{bmatrix}^T$ as defined above, we investigate the structure of the gradients of the constraint $g(\boldsymbol{z}) = \boldsymbol{A}(\boldsymbol{u})\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} - \boldsymbol{b} = 0$. Lets define the cost function as $F(\boldsymbol{z})$ and consider equality constraints arising from the collocation of the dynamics. Now, let the Lagrangian of the constrained optimization problem be $\mathcal{L}(z, \nu) := F(x) - \nu^T g(x)$. The Hessian of the Lagrangian is then

$$\mathcal{L}_{zz} = \nabla^2_{zz} F(\boldsymbol{z}) + \sum_{i=1}^{n_c} \nu_i \nabla^2_{zz} g_i(\boldsymbol{z}), \tag{5.73}$$

where $n_c$ is the number of constraints; here $n_c = n_x$. The Jacobian of the constraints is then defined as:

$$\begin{aligned}\nabla_z g(z) &:= \begin{bmatrix} \frac{\partial g_1(z)}{\partial z} \\ \frac{\partial g_2(z)}{\partial z} \\ \vdots \\ \frac{\partial g_{nc}(z)}{\partial z} \end{bmatrix}, \\ &= \begin{bmatrix} G(\boldsymbol{x}) & \boldsymbol{A}(u) \end{bmatrix}^T, \end{aligned} \tag{5.74}$$

where $G(\boldsymbol{x}) = \left[\frac{\partial(\boldsymbol{A}(\boldsymbol{u})\boldsymbol{x})}{\partial \boldsymbol{u}} + \boldsymbol{B}\right]$. The Hessian of the $i^{th}$ constraint can be written as

$$\nabla_{\boldsymbol{z}}^2 g_i(\boldsymbol{z}) = \nabla_{\boldsymbol{z}}(\nabla_{\boldsymbol{z}} g_i(\boldsymbol{z}))' = \nabla_{\boldsymbol{z}}[(\nabla_{\boldsymbol{u}} g_i(\boldsymbol{z}))' (\nabla_{\boldsymbol{x}} g_i(\boldsymbol{z}))'] \tag{5.75}$$

$$= \nabla_{\boldsymbol{z}} \begin{bmatrix} g_{i,:}(\boldsymbol{x}) & \boldsymbol{A}_{i,:}(u) \end{bmatrix}' \tag{5.76}$$

$$\nabla_{\boldsymbol{z}}^2 g_i(\boldsymbol{z}) = \begin{bmatrix} \nabla_{\boldsymbol{u}} g_{i,:}(\boldsymbol{x})^T & \nabla_{\boldsymbol{x}} g_{i,:}(\boldsymbol{x})^T \\ \nabla_{\boldsymbol{u}} \boldsymbol{A}_{i,:}(\boldsymbol{u})^T & \nabla_{\boldsymbol{x}} \boldsymbol{A}_{i,:}(\boldsymbol{u})^T \end{bmatrix}^T \tag{5.77}$$

$$\nabla_{\boldsymbol{z}}^2 g_i(\boldsymbol{z}) = \begin{bmatrix} 0_{n_u,n_u} & \Delta_i^T \\ \Delta_i & 0_{n_x,n_x} \end{bmatrix}, \tag{5.78}$$

where $\Delta_i \in \mathbb{R}^{n_{\boldsymbol{x}} \times n_{\boldsymbol{u}}}$ and $\Delta_i = \nabla_{\boldsymbol{u}} \boldsymbol{A}(\boldsymbol{u})_{i,:}^T$.

Now let $\boldsymbol{A}(\boldsymbol{u}) =: \begin{bmatrix} a_1(\boldsymbol{u})^T \\ a_2(\boldsymbol{u})^T \\ \vdots \\ a_{n_z}(\boldsymbol{u})^T \end{bmatrix}$ and $\tilde{\Delta} := \sum_{i=1}^{n_c} \nu_i \Delta_i$. Then,

$$\tilde{\Delta} = \sum_{i=1}^{n_c} \nu_i \partial_{\boldsymbol{u}} a_i(\boldsymbol{u}) \tag{5.79}$$

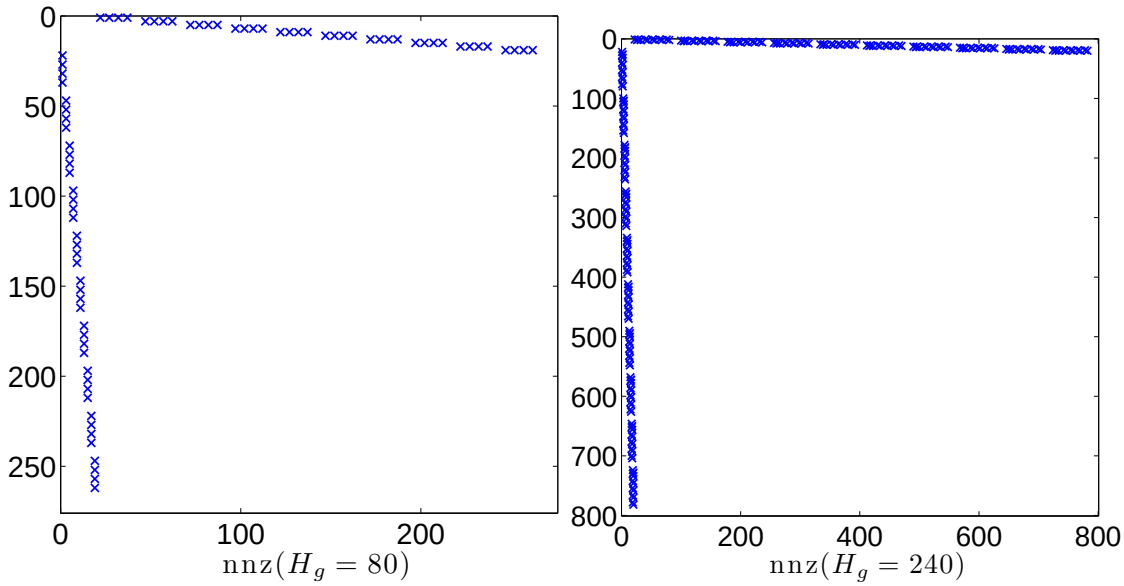
$$= \partial_{\boldsymbol{u}} \sum_{i=1}^{n_c} \nu_i a_i(\boldsymbol{u}) \tag{5.80}$$

$$= \partial_{\boldsymbol{u}} (\nu^T \boldsymbol{A}(\boldsymbol{u}))^T. \tag{5.81}$$

The Hessian for the constraints is implemented by simply noting that

$$H_g = \begin{bmatrix} 0_{n_u,n_u} & \tilde{\Delta}^T \\ \tilde{\Delta} & 0_{n_x,n_x} \end{bmatrix}. \tag{5.82}$$

From (5.74) and (5.82), the Jacobian has at most a similar sparsity level as $\boldsymbol{A}$, while the Hessian is much sparser; see Figure 5.11a. For the same example shown in Figure 5.3, the Hessian has a sparsity of $80/(250^2) \approx 0.13\%$, i.e. an order smaller. Of course, all the sparsity in $A(u_k)$ and $B$




(a) RK (b) EXP

Figure 5.11: The sparsity structure of the Hessian when RK and EXP collocation schemes are used.

are preserved in this noncondensed formulation, because there is no squaring of matrices.

Similarly, we can also use the direct exponential integrator of (5.57) to get the slightly different constraint functions

$$g(\boldsymbol{z}) = \boldsymbol{A}(\boldsymbol{u})\boldsymbol{x} + \boldsymbol{b}(x_0) = 0, \tag{5.83}$$

where the augmented NLP variable is now $\boldsymbol{z} := \begin{bmatrix} \boldsymbol{u}^T & \boldsymbol{x}^T \end{bmatrix}^T$, with size is $n_z = mM + (n+1)[N(2sl+1)-1]$. All variables are as defined in (5.45). As can be seen from Figure 5.11b and our sparsity analysis in Section 5.4, the exponential collocation can result in a larger number of constraints whose gradients are sparser. From the example in Figure 5.11, we see that the Hessian is about three time bigger but (compare 780 compared to 250) also three times sparser ($240/780^2 \approx 0.04\%$ compared to $0.13\%$).

Here we consider the direct transcription for use within an IPM to solve optimal control problems with bilinear dynamics and cost function of the form

$$J(x(\cdot),u(\cdot);x(t_0)) = \int_{t_0}^{t_f} L(x(t),u(t);p,t)dt + x(t_f)^T P x(t_f), \tag{5.84}$$

where $L(\cdot) := x^T Q x + u^T R u$. As usual, $Q \in \mathbb{R}^{n\times n}$ is assumed positive semidefinite and $R \in \mathbb{R}^{m\times m}$ positive definite. In many problems, the matrices $Q$ and $R$ are diagonal (or block diagonal) and therefore highly sparse. Therefore, in the example applications we consider below, the gradients of the Lagrangian in the direct transcription of the optimal control problem retain all the sparsity we discussed. As a result, although the problem size can grow in a noncondensed Runge-Kutta

or exponential transcription, the resulting NLP remains sparse and therefore possibly computationally as tractable as a denser smaller problem.

To validate our implementation, we first consider benchmark scalable bilinear optimal control problems from [135]. The optimal control considered there are of 'minimum fuel' problem type with the form

$$\min_{u(\cdot)} \int_0^{t_f} u^2(t)dt + x(t_f)^T P x(t_f), \tag{5.85}$$

subject to the bilinear dynamics

$$\dot{x}(t) = Ax(t) + u(t)N_1 x(t),\ x \in \mathbb{R}^n,\ u \in \mathbb{R},\ x(0) = x_0 \text{ given }.$$

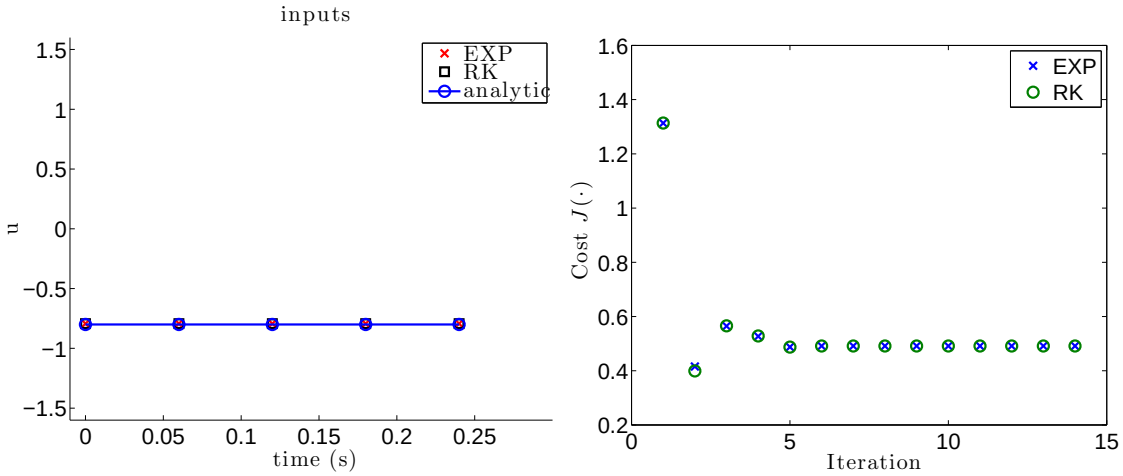
In [135, Example. 3.2], the problem with $P = I_n$, $A$ is the left-shift operator and $N_1$ is the identity operator is considered; i.e.

$$A = \begin{pmatrix} 0 & 1 & & & \\ & 0 & 1 & & \mathbf{0} \\ & & 0 & 1 & \\ & \mathbf{0} & & \ddots & \ddots \end{pmatrix},\ N_1 = I_n.$$

It is then theoretically shown, for any $n$, that the optimal control is a constant $u(t) = c, \forall t \in [t_0, t_f]$. Using our noncondensed Runge-Kutta and Exponential direct transcriptions and solving with an interior point solver confirms this. Figure 5.12a shows the solution when $n = 10$, $t_f = 0.3$, $x_0 = [1,\ 1/4, \ldots, 1/n^2]^T$, the number of integration steps is $N = 51$, and the number of control actions is set to $M = 5$. The solutions from both discretizations agree with the analytic solution. Similar results were confirmed for various $n$, different initial conditions and integration steps. Depending on initial conditions we get convergence by the interior point solver within 15 iterations; see Figure 5.12b. Since this example has a constant input and only the final value of the state $x(t_f)$ contributes to the cost, the problem was also solved with only a single integration step; even in this case the IPM with both transcriptions shows a similar convergence to the same analytic solution.

In [135, Example. 3.2], a similar optimal control problem for the system with matrices

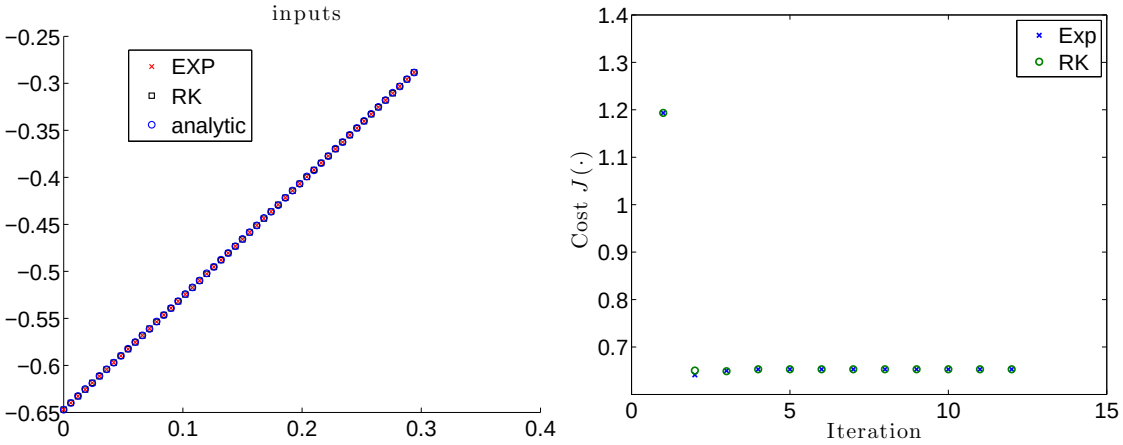
$$A = \begin{pmatrix} 0 & 1 & & & \\ & 0 & 2 & & \mathbf{0} \\ & & 0 & 3 & \\ & & & 0 & 4 \\ & \mathbf{0} & & & \ddots & \ddots \end{pmatrix},\ N_1 = \begin{pmatrix} 0 & 0 & & 0 & \\ 1 & 0 & \ldots & 0 & \\ 0 & 1 & 0 & & \\ & 0 & 1 & 0 & \\ & & \mathbf{0} & \ddots & \ddots \end{pmatrix},$$




(a) Optimal input from solving a direct transcription of (5.85)

(b) Convergence of the IPM

Figure 5.12: Comparison of the IPM solutions with the analytic solution for [135, Example. 3.2].




(a) Optimal input from solving a direct transcription of (5.85)

(b) Convergence of the IPM

Figure 5.13: Comparison of the IPM solutions with the analytic solution for [135, Example. 3.3].

is considered. With the same cost function, the optimal input function here is shown to be of the form $u(t) = c_1 + c_2 t$. Figures 5.13a and 5.13b show the result of a simulation with $x_0 = 1_n$, $t_f = 0.3$, . Since the input is linearly time varying, we have used $M = N = 51$ so that the control problem objective is approximated well. Both direct transcription schemes show convergence a corresponding analytic local solution. As before, the IPM solution tolerance was set to $10^{-7}$, which is attained within a few iterations. The analytic solution was computed by an a posteriori a brute force sweep of the control parameter spaces $c_1$ and $c_2$.

## 5.7 Conclusion

In this chapter, we have described a framework for using exponential integrators for semilinear systems. Making use of the variation of constants formula for such systems, we showed bilinear control systems with zero-order hold on the control input can be integrated exactly by exponential methods. Since exponential integrators rely on the computation of the action of a matrix exponential on a vector, a Krylov subspace method and a new direct method by Al-Mohy-Higham [113] for efficiently computing the action of a matrix exponential on a vector were considered. Although both can be used for exact to high precision integration, the latter was preferred because it has computable a priori error bounds as a function of the computational resources used. Both methods were compared with the classical explicit fourth order Runge-Kutta method.

The method of [113] was then adapted for integration of bilinear control systems with ZOH and new a priori error bounds on the computational complexity were derived as a function of the problem data. It was also shown using simulations and analysis that the direct exponential integrator with our new bounds can be, for given error tolerances, computationally superior to the Runge-Kutta method. We have also shown that the action of the matrix exponential on a vector using Al-Mohy-Higham [113] can be computed by solving an equivalent sparse linear system of equations. The resulting linear system of equations were shown to be nonsingular lower triangular. Bounds on the sparsity of these linear systems and their computational cost have been derived.

Based of these observations, a novel exponential integrator based direct transcription scheme for optimal control problems with bilinear dynamics was presented. A similar sparse collocation was formulated and analysed for the classical fourth order Runge-Kutta scheme for comparison. For both transcriptions, the structure and sparsity of the dynamic constraints in the resulting NLP were discussed. We leave some interesting aspects of the above results as future work.

# Chapter 6

# Conclusions and Future Work

## 6.1 Contributions

The focus of the first part of this thesis has been on model predictive control of a heaving wave energy converter with the aim to maximize extracted energy. This problem has also motivated the synthesis of lower-order observers for bilinear dynamical systems with bounds on the input. Another related problem we have tackled is the analysis of sparse ODE solvers for bilinear systems with zero-order hold (ZOH) on the control inputs. Here we summarise the main contributions of this thesis.

- In Chapters 2 and 3, we reviewed existing methods for active and passive control of wave energy converters for extracted energy maximization. We subsequently posed an optimal control problem for a generic heaving WEC with both active and passive power take-off control elements. Using an indirect method, the nature of the solution of the optimal control problem was analysed and shown to be of a bang-bang type when input constraints only are applied.
- Since the system model is bilinear and the cost function is not convex quadratic, the resulting optimization problem was shown not to be a quadratic program. Having discussed the difficulty in finding a global solution for this nonconvex optimal control problem, we have proposed a computationally efficient scheme to find local solutions. A variation of the projected gradient method (PGM) was exploited and shown to converge in a small number of iterations in various irregular ocean wave conditions. The performance of the PGM was also compared with a direct transcription of the optimal control problem solved using an interior point method. The computational cost of the PGM was shown to be much smaller in comparison to the direct transcription method. This is because the PGM was observed to converge within fewer iterations and also has lower computational complexity per iteration; the PGM requires only a single state and costate evaluation at each iteration, whereas the interior point solver performs gradient and Hessian computations

of the Lagrangian, as well as a number of line searches by solving a KKT system shown in (3.32).

- In Chapter 3, simulations were also used to compare the performance of the optimal controller we proposed within a receding horizon control scheme. In closed-loop, the model predictive methods we had developed were shown to give the WEC a wider bandwidth in both directions around its natural frequency. Simulations have also indicated that power output can be increased significantly using these methods compared to latching control and or an uncontrolled system. By considering a WEC whose damping element of the PTO is always engaged, we have demonstrated that device PTO parameter optimization is inherently dependent on how and whether the PTO is controlled. The performance of the predictive controllers on prediction horizon was also assessed.

- In the work of Chapter 3, complete knowledge of radiation forces was assumed in model predictive control. Since radiation forces cannot be directly measured in practice, the synthesis of (lower-order) model based observers for radiation force was investigated in Chapter 4. A low order $H_\infty$ filter design for bilinear systems with constraints on the control inputs was defined. The problem aims to design a filter that is based on a lower-order model of the plant, but with closed-loop robust disturbance attenuation guarantees around the full-order model. A survey of literature on the design of filters for such bilinear systems and related Lipschitz nonlinear systems was presented.

- By treating the bilinearizing inputs as linearly varying parameters, the bilinear system was posed as an equivalent polytopic differential inclusion. A discussion of linear parameter varying systems and quadratic stability and performance for polytopic linear differential inclusions was presented. The LPV formulation was then used to pose the filter design problem as an $L_2$ gain minimization for a polytopic system; the finite $L_2$ gain from disturbances to the filter estimation error was minimized. Since the resulting optimization problem is a (non-convex) BMI problem, we proposed an LMI-based coordinate descent algorithm to find solutions locally from a feasible initial condition. If a solution exists, a method for finding a feasible initial observer was given for the LMI method. Since the LMI-based algorithm is a local method for a nonconvex BMI optimization problem, the optimality gap to a global solution is not known. We have proposed an upper bound on the optimality gap through the solution of a full-order convex LMI problem; when small, this bound verifies the local solution is close in performance to a global solution.

- In Section 4.7, the proposed filter design scheme was successfully demonstrated using the wave energy example with varying-order models for the radiation subsystem. As the most widely used alternative for the design of filters for nonlinear systems, the Extended Kalman filter was also designed based on the same lower-order models for comparison. In addition to to alleviating the need to know the spectrum of the noise covariances by the EKF, the $H_\infty$ filter is robust to model uncertainties; this was demonstrated by showing

its better estimation error attenuation compared to an EKF filter of the same order under model parameter uncertainty. It was also noted that the real time implementation of an EKF filter requires the integration of the noise covariance dynamics in addition to the filter model. As a result, the $H_\infty$ filter has a smaller real time computational cost.

- In Chapter 4, we also investigated the performance of the filters in closed-loop simulations of a receding horizon control of the heaving buoy. The model predictive control algorithm (i.e. Algorithm 1 of Chapter 3) was implemented using estimates of the system dynamics and radiation forces from noisy measurements of the WEC velocity and position. Using simulations, the performance degradations from using the lower-order $H_\infty$ and EKF filters rather than assuming full state knowledge were shown to be marginal. The performance of increasing orders of models were compared. Similar marginal degradations in closed-loop performance were also seen under parametric uncertainty in the models used for filter design.

- Algorithm 1 of Chapter 3 motivated the need to efficiently compute the integration of bilinear system ODEs. Therefore, the computational efficiency of relevant ODE solvers for bilinear systems was investigated in Chapter 5. The classical Runge-Kutta method and advanced new results from matrix exponential based integration methods were surveyed. By using the variation of constants formula for semilinear systems, it was showed that bilinear control systems with ZOH on the control input can be integrated exactly to high accuracy by exponential methods.

- Since exponential integrators are based on the computation of the action of a matrix exponential on a vector, the iterative Krylov subspace method and a new direct method by [113] were considered for computing the action of a matrix exponential on a vector; literature on these sparse methods and the dense matrix exponential alternative were surveyed. Although both sparse schemes can be used for exact to high precision integration, the direct method of [113] was preferred because it has computable a priori error bounds as a function of the computational resources used.

- The method of [113] was then adapted for integration of bilinear control systems with ZOH. In Section 5.3.3, we derived new a priori error bounds on the computational complexity of employing the method of [113] for solving bilinear systems with ZOH. In Section 5.4, using these new bounds on computational complexity, we proposed a direct exponential integrator to solve bilinear ODEs via the solution of sparse linear systems. We also derived bounds for the sparsity and computational complexity of solving the resulting linear systems. This direct exponential method was then compared with a similarly implemented sparse fourth-order explicit Runge-Kutta scheme. The sparsity and computational complexity of the Runge-Kutta scheme was also presented.

- For both the Runge-Kutta and direct exponential method, the resulting linear systems in solving the bilinear ODEs were shown to be sparse block diagonal and unit lower trian-

gular. As such, they can be solved by forward substitution. Bounds were derived on the computational complexity of solving these linear systems. By considering parallelizing the forward substitution algorithm in solving these linear systems in hardware implementations, it was shown that the time complexity can be reduced significantly.

- Numerical experiments were used to assess the advantages of the exponential integrators compared with the classical Runge-Kutta method. In addition to the the wave energy system, a PDE heat transfer model for the controlled cooling of a metal slab was also used as an example. It was shown using simulations and analysis that the direct exponential integrator with our new bounds can be, for given error tolerances, computationally superior to the Runge-Kutta method. Based of these observations, a novel exponential integrator based direct transcription scheme for optimal control problems with bilinear dynamics was proposed. A similar sparse collocation was also formulated and analysed for the classical Runge-Kutta scheme for comparison. For both transcriptions, the structure and sparsity of the dynamic constraints in the resulting nonlinear program of the direct transcription were discussed. To demonstrate the feasibility of the proposed approach two example quadratic optimal control problems with bilinear dynamics were used.

## 6.2 Future Work

- In using the projected gradient algorithm of Chapter 3, offline simulations were employed to choose an appropriate constant step size. This was because the Lipschitz constant $L$ for the gradient of the objective function is not known; see Theorem (3.1). It will be interesting to investigate if the Lipschitz constant can be analytically determined as a function of the problem data or if a useful bound can be derived for it. In such a case, the convergence of the projected gradient method can, perhaps, be analytically investigated for the optimal control problem of (3.8) similar to the methods employed for the fast gradient schemes in [54].
- In the closed-loop receding horizon implementation of the proposed model predictive control in Chapter 3, stability was demonstrated only via simulation results. It will be interesting to further investigate closed-loop stability and under what conditions it can be guaranteed for the heaving WEC. Full knowledge of the excitation forces was also assumed in the simulations. In future work, various prediction methods should be assessed. The effect of prediction errors on closed loop performance could be investigated.
- Because of the unavailability of experimental resources to the work of this thesis, the optimal control methods proposed in Chapter 3 have only been assessed via simulations using approximate models. It will be interesting to investigate feasibility on a scaled physical model in a wave tank.

- In Chapter 4, an LMI-based coordinate descent algorithm was proposed to solve a BMI optimization problem, which works well in practice. By further investigating the structure of the problem, future work can investigate the use of convexification relaxations with potentially tighter bounds on the optimality gap.
- The low-order $H_\infty$ synthesis proposed in Chapter 4 is based on lower-order models for the bilinear system. These models can come from model reduction of a full-order bilinear model or 'coarser' approximations for bilinear PDE models. In future, the question "in what sense should the lower-order models be close or similar to the full-order model in order to give filters of performance close to the full-order model based filters?" can be investigated. In other words, we may want to find measures of how good a model is for estimator design.
- In Section 5.4, we had used forward substitution for solving the sparse lower triangular linear systems arising from the exponential and Runge-Kutta collocations. It will be interesting to study the numerical conditioning of these linear systems with sampling rate. As a novel collocation scheme for direct transcription of optimal control problems, it may then be interesting to study the Jacobian and Hessian of the constraints as well as the KKT matrix arising from the use of the exponential method within direct transcription of quadratic problems with bilinear dynamics. Tailor made solvers for the resulting KKT systems may also be considered since we have shown its structure.
- Figure 5.7 implies that if the required integration error tolerance was of the order $10^{-12}$, the computational complexity of the exponential method would be an order less than that of the Runge-Kutta in delivering a similarly accurate approximation of the solution. It would also be interesting to investigate applications where integration errors need to be much smaller than the requirements in the wave energy example. In such cases, the computational gains from using the direct exponential scheme compared to the classical Runge-Kutta method would be substantial.
- The sparse linear matrices for the Runge-Kutta and direct exponential collocations have different bandwidth and sparsity when their size is similar; this may mean it could be cheaper to solve the latter since the computational complexity in solving a sparse lower triangular block diagonal linear system grows linearly with the sparsity. It would be interesting to investigate various iterative and direct methods for solving these linear systems of equations.

# Appendices

## A Adjoint system dynamics (3.12) for the OCP in (3.8)

$$\dot{\lambda}_1(t) = \frac{C\lambda_2}{M+\mu_\infty}, \tag{A.1}$$

$$\dot{\lambda}_2(t) = 2B_{pto}u_2(t)x_2(t) - \lambda_1(t) + \frac{B_{pto}}{M+\mu_\infty}\lambda_2 u_2 - Gu_1(t) - B_r^T\lambda_{3:n}(t), \tag{A.2}$$

$$\dot{\lambda}_{3:n}(t) = -A_r^T\lambda_{3:n}(t) + C_r^T\lambda_2(t), \tag{A.3}$$

where all the variables are as given in (3.7). Note that this adjoint dynamics is also a bilinear system. Overall, the system and adjoint equations constitute a bilinear boundary value problem.

## B Exact Solution of Semilinear Initial Value Problems with polynomial in time nonlinearity

Here we show the proof for Lemma 5.1. We derive expressions for the solution of semilinear initial value problems with a polynomial in time nonlinearity.

*Proof.* This proof is entirely inspired by the proof of Lemma 1 in [123]. Separating the integrating factor $e^{(\tau-s)L}$, the solution of the non-autonomous initial value problem (5.11) can be written

as:

$$x(\tau) = e^{\tau L}x_0 + e^{\tau L}\int_0^\tau e^{-sL}\sum_{j=0}^{p-1}\frac{s^j}{j!}b_j ds,\ \forall\tau,\ 0<\tau<h. \tag{B.1}$$

$$\text{Let } s = \theta\tau, \text{ and rearrange sum and integral. Then,}$$

$$x(\tau) = e^{\tau L}x_0 + e^{\tau L}\sum_{j=0}^{p-1}\int_0^1 e^{-\theta\tau L}\frac{\theta^j\tau^j}{j!}b_j\tau d\theta, \tag{B.2}$$

$$x(\tau) = e^{\tau L}x_0 + e^{\tau L}e^{-\tau L}\sum_{j=0}^{p-1}\tau^{j+1}\int_0^1 e^{(1-\theta)\tau L}\frac{\theta^j}{j!}d\theta b_j, \tag{B.3}$$

$$x(\tau) = e^{\tau L}x_0 + \sum_{j=0}^{p-1}\tau^{j+1}\varphi_{j+1}(\tau L)b_j, \tag{B.4}$$

because $\varphi_l(z) = \int_0^1 e^{(1-\theta)z}\frac{\theta^{l-1}}{(l-1)!}d\theta$ [113, (2.3)].

□

# C Padé Approximation for $e^X$

The $(k,m)$ Padé approximation to $e^X$ is given as:

$$r_{k,m} = D_{k,m}^{-1}N_{k,m},$$

where

$$N_{k,m} = \sum_{j=1}^{k}\frac{(k+m-j)!(k)!}{(k+m)!(k-j)!}\frac{X^j}{j!} \tag{C.1}$$

and

$$D_{k,m} = \sum_{j=1}^{k}\frac{(k+m-j)!(m)!}{(k+m)!(m-j)!}\frac{X^j}{j!} \tag{C.2}$$

The diagonal Padé approximants $r_m(X)$ (i.e. $k=m$) are preffered because of their superior accuracy [119]. For these, one can show the unreliability of $r_m(X)$ for matrices $X$ with widely spread eigenvalues; i.e. $\text{cond}(D_m^{-1}) \approx \text{cond}(-X/2) \geq e^{\alpha_1-\alpha_n}$, $\alpha_1 \geq \ldots, \geq \alpha_n$ being the real parts of the eigenvalues of $X$. So, the Padé approximant is only used when $\rho(X)$ or $\|X\|$ is small. See [119, Eqn. 2.1-2.6] for backward error analysis using the Padé approximation.

# D Bounds on Computational Cost of Runge-Kutta and Exponential Integrators

The linear systems for both the Runge-Kutta and Exponential collocation schemes result in a unit lower linear system of equations. The term 'unit' in 'unit triangular' refers to all the diagonal elements having a value of 1; the matrices are therefore nonsingular and forward substitution can be applied without any need for a factorisation step. As in the standard numerical algebra literature, we use flops (or floating point operations) as a measure of computational complexity; a flop refers to one addition, subtraction, multiplication or division between two floating point numbers.

In a forward substitution process [89, App. C.2.1], if a row of a nonsingular lower triangular matrix has $k$ nonzero entries, then the number of flops required in the substitution for that row is $2(k-1)+1 = 2k-1$. Now consider the rows of $\boldsymbol{A}(\boldsymbol{u})$ in (5.45). Analysing the $N$ times repeating blocks of size $5n \times 5n$, we can see $1/5$ of the rows have six elements. Since the number of elements in the rows of the $A(u_k)$ is not necessarily constant, we can use a bound on them. Let $n_A$ stand for the bound on the number nonzero elements of $A(u_k)$, $n_A = \max_u\{nnz(A(u)) : u \text{ is admissible}\}$. The average number of elements per row of $\boldsymbol{A}(\boldsymbol{u})$, let's call it $k_{RK}$, can then be bounded by

$$k_{RK} \leq \frac{((n_A/n)+1)+3((2n_A/n)+1)+6}{5} = \frac{7n_A}{5n}+2. \tag{D.1}$$

Therefore, the total cost of solving (5.45) is then bounded by

$$\begin{aligned} \text{cost}_{RK} &\leq (2k_{RK}-1)n_x,\ n_x = (1+5N)n && \text{(D.2)}\\ &\leq (2(\frac{7n_A}{5n}+2)-1)(1+5N)n, && \text{(D.3)}\\ \text{cost}_{RK} &\leq (\frac{14n_A}{5}+3n)(1+5N), && \text{(D.4)} \end{aligned}$$

where all the variables are as defined in the main text.

In a similar analysis for the exponential collocation, the structure of (5.56) reveals that its exact average number of rows can be bounded by (D.5). Note here that the $X$ blocks in (5.56) would be replaced by matrices $\tilde{A}(u_k) = \begin{bmatrix} A(u_k) & Bu_k \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}$.

$$k_{RK} \leq \frac{(1+\frac{n_A+n}{n+1})+3}{2} = \frac{n_A+n}{2(n+1)}+2. \tag{D.5}$$

The total cost of solving the linear system for the exponential collocation is then bounded by

$$\begin{aligned} \text{cost}_{EXP} &\leq (2k_{EXP}-1)n_x,\ n_x=(n+1)(2sl+1)N && \text{(D.6)} \\ &\leq (\frac{n_A+n}{n+1}+3)(n+1)(2sl+1)N, && \text{(D.7)} \\ &\leq (n_A+4n+3)(2sl+1)N, && \text{(D.8)} \end{aligned}$$

where all the variables are as defined in the main text.

# References


[1] Paris: REN21 Secretariat. *Renewables 2012: Global status report*. Worldwatch Institute Washington, DC, 2012.

[2] J. Cruz. *Ocean Wave Energy Current Status and Future Perspectives*. Springer, 2008.

[3] J. Falnes. A review of wave-energy extraction. *Marine Structures*, 20(4):185–201, 2007.

[4] M. Folley and T. J. T. Whittaker. Analysis of the nearshore wave energy resource. *Renewable Energy*, 34(7):1709–1715, 2009.

[5] A. Babarit and A. H. Clément. Optimal latching control of a wave energy device in regular and irregular waves. *Applied Ocean Research*, 28(2):77–91, 2006.

[6] A. Clément, P. McCullen, A. F. de O. Falcao, , et al. Wave energy in europe: current status and perspectives. *Renewable and Sustainable Energy Reviews*, 6(5):405–431, 2002.

[7] B. Drew, A. R. Plummer, and M. N. Sahinkaya. A review of wave energy converter technology. *Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy*, 223(8):887–902, 2009.

[8] M. Robinson. Renewable energy technologies for use on the outer continental shelf. National Renewable Energy Lab Presentation, June 2006.

[9] C. Charcosset. A review of membrane processes and renewable energies for desalination. *Desalination*, 245(1-3):214–231, 2009.

[10] P. A. Davies. Wave-powered desalination: resource assessment and review of technology. *Desalination*, 186(1-3):97–109, 2005.

[11] J. Falnes. *Ocean waves and oscillating systems: linear interaction including wave-energy extraction*. Cambridge University Press, 2002.

[12] R. S. Johnson. *A modern introduction to the mathematical theory of water waves*, volume 19. Cambridge University Press, 1997.

[13] D. J. Acheson. *Elementary fluid dynamics*. Oxford University Press, USA, 1990.

[14] J. Falnes and J. Hals. Heaving buoys, point absorbers and arrays. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 370 (1959):246–277, 2012.

[15] Z. Yu and J. Falnes. State-space modelling of a vertical cylinder in heave. *Applied Ocean Research*, 17(5):265–275, 1995.

[16] E. Kristiansen, Å. Hjulstad, and O. Egeland. State-space representation of radiation forces in time-domain vessel models. *Ocean Engineering*, 32(17-18):2195–2216, 2005.

[17] R. Taghipour, T. Perez, and T. Moan. Hybrid frequency-time domain models for dynamic response analysis of marine structures. *Ocean Engineering*, 35(7):685–705, 2008.

[18] T. Bjarte-Larsson, J. Falnes, and T. Moan. Comparison of results from time-domain simulations and model tests of a water-pumping wave-power unit. In *The Proceedings of The Sixteenth (2006) International Offshore and Polar Engineering Conference, May*, pages 416–422, 2006.

[19] A. F. O. Falcão, P. A. P. Justino, J. C. C. Henriques, and J. M. C. S. Andrẽ. Reactive versus latching phase control of a two-body heaving wave energy converter. In *Proceedings of the European Control Conference*, pages 3731–3736, 2009.

[20] K. Unneland, T. I. Fossen, P. Van Dooren, and O. Egeland. Low order potential damping models for surface vessels. In *Proc. of the 7th IFAC Conference on Manoeuvering and Control of Marine Craft*, 2006.

[21] A. A. E. Price. *New Perspectives on Wave Energy Converter Control*. PhD thesis, The University of Edinburgh, Edinburgh, UK, 2009.

[22] W. E. Cummins. The impulse response function and ship motions. Technical Report Report number 1661, Hydrodynamics Laboratory Research and Development Report, 1962.

[23] S. Y. Kung. A new identification and model reduction algorithm via singular value decomposition. In *Proc. 12th Asilomar Conf. Circuits, Syst. Comput., Pacific Grove, CA*, pages 705–714, 1978.

[24] G. Balas, R. Chiang, A. Packard, and M. Safonov. Robust control toolbox. *For Use with Matlab. User's Guide, Version*, 3, 2005.

[25] J. Falnes. On non-causal impulse response functions related to propagating water waves. *Applied Ocean Research*, 17(6):379–389, 1995.

[26] F. Fusco and J. V. Ringwood. Short-term wave forecasting for real-time control of wave energy converters. *Sustainable Energy, IEEE Transactions on*, 1(2):99–106, 2010.

[27] J. Falnes. Optimum control of oscillation of wave-energy systems. wave energy converters: Generic technical evaluation study. Technical Report Annex Report B1: Device fundamentals/Hydrodynamics, Hydrodynamics Laboratory Research and Development Report, August 1993.

[28] J. Falnes. Optimum control of oscillation of wave-energy converters. *International Journal of Offshore and Polar Engineering*, 12(2):147–154, 2002.

[29] H. K. Khalil. *Nonlinear systems*, volume 3. Prentice Hall, 2002.

[30] K. Budal, J. Falnes, L. C. Iversen, P. M. Lillebekken, G. Oltedal, T. Hals, T. Onshus, and A. .S. Høy. The norwegian wave-power buoy project. In *The Second International Symposium on Wave Energy Utillization, Trondheim*, 1982.

[31] R. E. Hoskin and N. K. Nichols. Latching control of a point absorber. In *Proc. 3rd International Symposium on Wave, Tidal, OTEC, and Small Scale Hydro Energy*, pages 317–330, 1986.

[32] A. Babarit, G. Duclos, and A. H. Clément. Comparison of latching control strategies for a heaving wave energy device in random sea. *Applied Ocean Research*, 26(5):227–238, 2004.

[33] A. F. O. Falcão. Phase control through load control of oscillating-body wave energy converters with hydraulic PTO system. *Ocean Engineering*, 35(3-4):358–366, 2008.

[34] U. A. Korde. Phase control of floating bodies from an on-board reference. *Applied Ocean research*, 23(5):251–262, 2001.

[35] J. J. Cândido and P. A. P. S. Justino. Modelling, control and pontryagin maximum principle for a two-body wave energy device. *Renewable Energy*, 2010.

[36] U. A. Korde. Latching control of deep water wave energy devices using an active reference. *Ocean Engineering*, 29(11):1343–1355, 2002.

[37] J. A. M. Cretel, G. Lightbody, G. P. Thomas, and A. W. Lewis. Maximisation of energy capture by a wave-energy point absorber using model predictive control. In *Proc. IFAC World Congress on Automatic Control, Milano, Italy*, 2011.

[38] J. Falnes. Radiation impedance matrix and optimum power absorption for interacting oscillators in surface waves. *Applied Ocean Research*, 2(2):75–80, 1980.

[39] G. P. Thomas and D. V. Evans. Arrays of three-dimensional wave-energy absorbers. *Journal of Fluid Mechanics*, 108(1):67–88, 1981.

[40] A. Babarit, M. Guglielmi, and A. H. Clément. Declutching control of a wave energy converter. *Ocean Engineering*, 36(12):1015–1024, 2009.

[41] G. S. Payne, U. B. P. Stein, M. Ehsan, N. J. Caldwell, and W. H. S. Rampen. Potential of digital displacement hydraulics for wave energy conversion. In *Proc. of the Sixth European Wave and Tidal Energy Conference, Glasgow*, 2005.

[42] R. Henderson. Design, simulation, and testing of a novel hydraulic power take-off system for the pelamis wave energy converter. *Renewable Energy*, 31(2):271–283, 2006.

[43] M. A. Mueller, H. Polinder, and N. Baker. Current and novel electrical generator technology for wave energy converters. In *IEEE International Electric Machines & Drives Conference*, volume 2, pages 1401–1406, 2007.

[44] J. Hals, J. Falnes, and T. Moan. Constrained optimal control of a heaving buoy wave-energy converter. *Journal of Offshore Mechanics and Arctic Engineering*, 133:011401, 2011.

[45] A. A. E. Price and A. R. Wallace. Non-linear methods for next wave estimation. In *Proceedings of the 7th European Wave and Tidal Energy Conference, Porto*, 2007.

[46] F. Fusco and J. V. Ringwood. A study of the prediction requirements in real-time control of wave energy converters. *Sustainable Energy, IEEE Transactions on*, 3(1):176–184, 2012.

[47] J. T. Scruggs, S. M. Lattanzio, A. A. Taflanidis, and I. L. Cassidy. Optimal causal control of a wave energy converter in a random sea. *Applied Ocean Research*, 42:1–15, 2013.

[48] J. M. Maciejowski. *Predictive control: with constraints*. Pearson Education, 2002.

[49] A. E. Bryson and Y. C. Ho. *Applied optimal control: optimization, estimation, and control*. Hemisphere Publications, 1975.

[50] J. L. Speyer and D. H. Jacobson. *Primer on optimal control theory*. Society for Industrial and Applied Mathematics (SIAM), 2010.

[51] R. F. Stengel. *Optimal control and estimation*. Dover Publications, 1994.

[52] A. J. Krener. The high order maximal principle and its application to singular extremals. *SIAM Journal on Control and Optimization*, 15(2):256–293, 1977.

[53] D. P. Bertsekas. *Nonlinear programming*. Athena Scientific Belmont, Massachusetts, 1999.

[54] S. Richter, C. N. Jones, and M. Morari. Computational complexity certification for real-time mpc with input constraints based on the fast gradient method. *Automatic Control, IEEE Transactions on*, 57(6):1391–1403, 2012.

[55] Y. Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer, 2004.

[56] Z. Q. Luo and P. Tseng. On the linear convergence of descent methods for convex essentially smooth minimization. *SIAM J. Control Optim*, 30(2):408–425, 1992.

[57] D. P. Bertsekas. *Dynamic Programming and Optimal Control*. Athena Scientific, Belmont, Massachusetts, USA, 2005.

[58] G. Li, G. Weiss, M. Mueller, S. Townley, and M. Belmont. Wave energy converter control by wave prediction and dynamic programming. *Renewable Energy*, 48:392–403, 2012.

[59] A. A. Agrachev, G. Stefani, and P. L. Zezza. Strong optimality for a bang-bang trajectory. *SIAM journal on control and optimization*, 41(4):991–1014, 2003.

[60] H. Maurer, C. Büskens, J. H. R. Kim, and C. Y. Kaya. Optimization methods for the verification of second order sufficient conditions for bang-bang controls. *Optimal Control Applications and Methods*, 26(3):129–156, 2005.

[61] H. Maurer and G. Vossen. Sufficient conditions and sensitivity analysis for optimal bang-bang control problems with state constraints. *System Modeling and Optimization*, pages 82–99, 2009.

[62] J. T. Betts. *Practical methods for optimal control and estimation using nonlinear programming*. Cambridge University Press, 2010.

[63] J. Nocedal and S. J. Wright. *Numerical optimization*. Springer Verlag, 2006.

[64] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. .OM. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, 2000.

[65] M. Diehl, R. Amrit, and J. B. Rawlings. A lyapunov function for economic optimizing model predictive control. *Automatic Control, IEEE Transactions on*, 56(3):703–707, 2011.

[66] D. Angeli, R. Amrit, and J. B. Rawlings. On average performance and stability of economic model predictive control. *Automatic Control, IEEE Transactions on*, 57(7):1615–1626, 2012.

[67] A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1): 25–57, 2006.

[68] G. Duclos, A. Babarit, and A. H. Clément. Optimizing the power take off of a wave energy converter with regard to the wave climate. *Journal of Offshore Mechanics and Arctic Engineering*, 128:56, 2006.

[69] B. Teillant, J. C. Gilloteaux, and J. V. Ringwood. Optimal damping profile for a heaving buoy wave energy converter. In *8th IFAC Conference on Control Applications in Marine Systems*, Universitat Rostock, Germany, September 2010.

[70] P. M. Pardalos and V. Yatsenko, editors. *Optimization and Control of Bilinear Systems: Theory, Algorithms, and Applications*, volume 13. Springer Berlin, 2008.

[71] E. Abraham and E. C. Kerrigan. Optimal active control and optimization of a wave energy converter. *IEEE Transactions on Sustainable Energy*, 4(2):324–332, 2013.

[72] L. El Ghaoui and S. I. Niculescu. *Advances in linear matrix inequality methods in control*, volume 2. Society for Industrial Mathematics, 2000.

[73] H. Li and M. Fu. A linear matrix inequality approach to robust $\mathcal{H}_\infty$ filtering. *IEEE Transactions on Signal Processing*, 45(9):2338–2350, 1997.

[74] C. E. de Souza, U. Shaked, and M. Fu. Robust $\mathscr{H}_\infty$ filtering for continuous time varying uncertain systems with deterministic input signals. *IEEE Transactions on Signal Processing*, 43(3):709–719, 1995.

[75] Z. Wang, Y. Liu, and X. Liu. $\mathscr{H}_\infty$ filtering for uncertain stochastic time-delay systems with sector-bounded nonlinearities. *Automatica*, 44(5):1268–1277, 2008.

[76] R. Rajamani. Observers for Lipschitz nonlinear systems. *IEEE Transactions on Automatic Control*, 43(3):397–401, 1998.

[77] A. Alessandri. Observer design for nonlinear systems by using input-to-state stability. In *Decision and Control (CDC), 2004. 43rd IEEE Conference on*, volume 4, pages 3892–3897. IEEE, 2004.

[78] M. Abbaszadeh and H. J. Marquez. Robust $h_\infty$ observer design for a class of nonlinear uncertain systems via convex optimization. In *American Control Conference (ACC), 2007*, pages 1699–1704. IEEE, 2007.

[79] B. Song and J. K. Hedrick. Nonlinear observer design for Lipschitz nonlinear systems. In *American Control Conference (ACC), 2011*, pages 2578–2583. IEEE, 2011.

[80] B. Gérard, H. Souley Ali, M. Zasadzinski, and M. Darouach. $H_\infty$-filter for bilinear systems using LPV approach. *IEEE Transactions on Automatic Control*, 55(7):1668–1674, 2010.

[81] H. Souley Ali, M. Zasadzinski, H. Rafaralahy, and M. Darouach. Robust $H_\infty$ reduced order filtering for uncertain bilinear systems. *Automatica*, 42(3):405 – 415, 2006.

[82] B. L. Jones and E. C. Kerrigan. When is the discretization of a spatially distributed system good enough for control? *Automatica*, 46(9):1462–1468, 2010.

[83] M. Zasadzinski, H. Souley Ali, and M. Darouach. Robust reduced order unbiased filtering for uncertain systems. *International Journal of Control*, 79(2):93–106, 2006.

[84] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear matrix inequalities in system and control theory*, volume 15. Society for Industrial Mathematics, 1994.

[85] R. Shorten, F. Wirth, O. Mason, K. Wulff, and C. King. Stability criteria for switched and hybrid systems. *SIAM review*, 49(4):545–592, 2007.

[86] J. S. Shamma. *An Overview of LPV Systems*, pages 3–26. Springer, 2012.

[87] C. Scherer and S. Weiland. Linear matrix inequalities in control. DISC Course Lecture Notes, 2000.

[88] M. Henk, J. Richter-Gebert, and G. M. Ziegler. 16 basic properties of convex polytopes. *Handbook of discrete and computational geometry*, pages 255–382, 2004.

[89] S. P. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge Univ Pr, 2004.

[90] P. Gahinet, P. Apkarian, and M. Chilali. Affine parameter-dependent lyapunov functions and real parametric uncertainty. *IEEE Transactions on Automatic Control*, 41(3):436–442, 1996.

[91] I. J. Couchman, E. C. Kerrigan, and C. Böhm. Model reduction of homogeneous-in-the-state bilinear systems with input constraints. *Automatica*, 47(4):761–768, 2011.

[92] A. C. Antoulas. *Approximation of large-scale dynamical systems*, volume 6. Society for Industrial Mathematics, 2005.

[93] K. M. Grigoriadis and J. T. Watson Jr. Reduced-order $H_\infty$ and $L_2 - L_\infty$ filtering via linear matrix inequalities. *IEEE Transactions on Aerospace and Electronic Systems*, 33(4): 1326–1338, 1997.

[94] V. Blondel and J. N. Tsitsiklis. Np-hardness of some linear control design problems. *SIAM Journal on Control and Optimization*, 35(6):2118–2127, 1997.

[95] T. D. Quoc, S. Gumussoy, W. Michiels, and M. Diehl. Combining convex–concave decompositions and linearization approaches for solving BMIs, with application to static output feedback. *IEEE Transactions on Automatic Control*, 57(6):1377–1390, 2012.

[96] K-C. Goh, M. G. Safonov, and G. P. Papavassilopoulos. Global optimization for the biaffine matrix inequality problem. *Journal of global optimization*, 7(4):365–380, 1995.

[97] C. W. J. Hols and C. W. Scherer. A sum-of-squares approach to fixed-order $H_\infty$-synthesis. *Positive Polynomials in Control*, pages 580–580, 2005.

[98] D. Henrion and J-B. Lasserre. Convergent relaxations of polynomial matrix inequalities and static output feedback. *IEEE Transactions on Automatic Control*, 51(2):192–202, 2006.

[99] S. Kanev, C. Scherer, M. Verhaegen, and B. De Schutter. Robust output-feedback controller design via local bmi optimization. *Automatica*, 40(7):1115–1127, 2004.

[100] E. Simon, P. Rodriguez-Ayerbe, C. Stoica, D. Dumur, and V. Wertz. LMI-based coordinate descent method for solving BMIs in control design: A short survey & two improvements. In *Proceedings of the 18th IFAC World Congress*, 2011.

[101] M. Kocvara and M. Stingl. *PENBMI User's Guide (Version 2.1)*. PENOPT GbR, www.penopt.com, February 2012.

[102] M. Kočvara and M. Stingl. PENNON: A code for convex nonlinear and semidefinite programming. *Optimization methods and software*, 18(3):317–333, 2003.

[103] M. Kočvara F. Leibfritz, M. Stingl, and D. Henrion. A nonlinear sdp algorithm for static output feedback problems in compleib. In *Proceedings of the 16th IFAC World Congress on Automatic Control, Prague*, 2005.

[104] F. Leibfritz. Compleib: Constrained matrix optimization problem library, 2003.

[105] C. Scherer, P. Gahinet, and M. Chilali. Multiobjective output-feedback control via lmi optimization. *IEEE Transactions on Automatic Control*, 42(7):896–911, 1997.

[106] T. Iwasaki. The dual iteration for fixed-order control. *IEEE Transactions on Automatic Control*, 44(4):783–788, 1999.

[107] J. Lofberg. YALMIP: A toolbox for modeling and optimization in Matab. In *Computer Aided Control Systems Design, 2004 IEEE International Symposium on*, pages 284–289. IEEE, 2004.

[108] E. Simon and V. Wertz. Improvement of the LMI change of variables for multi-objective control design problems. *Optimal Control Applications and Methods*, 2012. ISSN 1099-1514. doi: 10.1002/oca.2040. URL `http://dx.doi.org/10.1002/oca.2040`.

[109] D. Simon. *Optimal state estimation: Kalman, $H_\infty$, and nonlinear approaches*. Wiley-Interscience, 2006.

[110] K. Reif, F. Sonnemann, and R. Unbehauen. An EKF-based nonlinear observer with a prescribed degree of stability. *Automatica*, 34(9):1119–1123, 1998.

[111] K. C. Toh, M. J. Todd, and R. H. Tütüncü. SDPT3—a Matab software package for semidefinite programming, version 1.3. *Optimization Methods and Software*, 11(1-4): 545–581, 1999.

[112] B. V. Minchev and W. M. Wright. A review of exponential integrators for first order semi-linear problems. Technical Report 2/05, Norwegian University of Science and Technology, Trondheim, Norway, 2005.

[113] A. H. Al-Mohy and N. J. Higham. Computing the action of the matrix exponential, with an application to exponential integrators. *SIAM Journal on Scientific Computing*, 33(2): 488–511, 2011.

[114] M. Hochbruck and A. Ostermann. Exponential integrators. *Acta Numerica*, 19(1):209–286, 2010.

[115] C. Lesiak and A. Krener. The existence and uniqueness of volterra series for nonlinear systems. *IEEE Transactions on Automatic Control*, 23(6):1090–1095, 1978.

[116] P. d'Alessandro, A. Isidori, and A. Ruberti. Realization and structure theory of bilinear dynamical systems. *SIAM Journal on Control*, 12:517, 1974.

[117] Y. Saad. Analysis of some krylov subspace approximations to the matrix exponential operator. *SIAM Journal on Numerical Analysis*, pages 209–228, 1992.

[118] C. Moler and C. Van Loan. Nineteen dubious methods for computing the matrix exponential, twenty-five years later. *Siam Review*, 45(1):3–49, 2003.

[119] N. J. Higham. The scaling and squaring method for the matrix exponential revisited. *SIAM Journal on Matrix Analysis and Applications*, 26(4):1179–1193, 2005.

[120] A. H. Al-Mohy and N. J. Higham. A new scaling and squaring algorithm for the matrix exponential. *SIAM Journal on Matrix Analysis and Applications*, 31(3):970–989, 2009.

[121] N. J. Higham. *Accuracy and stability of numerical algorithms*. Siam, 1996.

[122] G. H. Golub and C. F. Van Loan. *Matrix computations*. Johns Hopkins University Press (Baltimore), 1983.

[123] B. Skaflestad and W. M. Wright. The scaling and modified squaring method for matrix functions related to the exponential. *Applied Numerical Mathematics*, 59(3-4):783–799, 2009.

[124] N. J. Higham. *Functions of matrices: theory and computation*. Society for Industrial Mathematics, 2008.

[125] A. Greenbaum. *Iterative methods for solving linear systems*, volume 17. Society for Industrial Mathematics, 1997.

[126] J. Niesen and W. M. Wright. Algorithm 919: A krylov subspace algorithm for evaluating the $\varphi$–functions appearing in exponential integrators. *ACM Transactions on Mathematical Software (TOMS)*, 38(3):22, 2012.

[127] R. B. Sidje. Expokit: a software package for computing matrix exponentials. *ACM Transactions on Mathematical Software (TOMS)*, 24(1):130–156, 1998.

[128] E. J. Mastascusa. A relation between liou's method and the fourth-order runge-kutta method for evaluating transient response. *Proceedings of the IEEE*, 57(5):803–804, 1969.

[129] L. F. Shampine and M. W. Reichelt. The Matlab ode suite. *SIAM journal on scientific computing*, 18(1):1–22, 1997.

[130] G. A. Constantinides. Tutorial paper: Parallel architectures for model predictive control. In *Proceedings of the European Control Conference, Budapest*, pages 138–143, 2009.

[131] A. Hasan, E. C. Kerrigan, and G. A. Constantinides. Solving a positive definite system of linear equations via the matrix exponential. In *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pages 2299–2304. IEEE, 2011.

[132] P. Benner and T. Damm. Lyapunov equations, energy functionals, and model order reduction. *SIAM journal on control and optimization*, 49(2):686–710, 2011.

[133] P. Benner and J. Saak. *Linear-quadratic regulator design for optimal cooling of steel profiles*. Number Preprint SFB393/05-05. Citeseer, 2005.

[134] A. Shahzad, E. C. Kerrigan, and G. A. Constantinides. A stable and efficient method for solving a convex quadratic program with application to optimal control. *SIAM Journal on Optimization*, 22(4):1369–1393, 2012.

[135] S. P. Banks and M. K. Yew. On the optimal control of bilinear systems and its relation to lie algebras. *International Journal of Control*, 43(3):891–900, 1986.