

AMERICAN UNIVERSITY OF BEIRUT

MARINE LOCOMOTION: A TETHERED
UAV–BUOY WITH AN INTEGRATED
CONTROL SYSTEM

by

AHMAD HASSAN KOURANI

A dissertation
submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
to the Department of Mechanical Engineering
of the Maroun Semaan Faculty of Engineering and Architecture
at the American University of Beirut

Beirut, Lebanon
January 2022

AMERICAN UNIVERSITY OF BEIRUT

MARINE LOCOMOTION: A TETHERED
UAV–BUOY WITH AN INTEGRATED
CONTROL SYSTEM

by
AHMAD HASSAN KOURANI

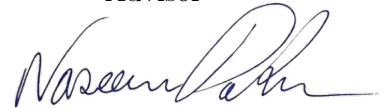
Approved by:

Dr. Marwan Darwish, Professor
Mechanical Engineering, AUB

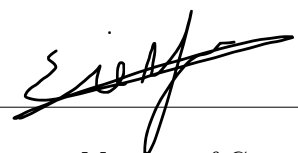


Committee Chair


Dr. Naseem Daher, Assistant Professor
Electrical and Computer Engineering, AUB

Advisor


Dr. Elie Shammas, Associate Professor
Mechanical Engineering, AUB

Member of Committee


Dr. James Mynderse, Associate Professor
Mechanical Engineering, Lawrence Technological University

Member of Committee


Dr. Najib Metni, Associate Professor
Mechanical Engineering, Notre Dame University

Member of Committee


Date of thesis defense: January 20, 2022

ACKNOWLEDGEMENTS

In the Name of Allah.

I want to dedicate this work to *Eng.* Hussein Akhdar. This dissertation is the outcome of my special journey in control systems and robotics research. I feel happy to remember all of the challenges that I have encountered and the feelings that I have experienced, which lead me to where I am today, empowered with patience, wisdom, strength, belief, and courage. From the marine expertise of my previous advisor, Dr. Matthias Liermann, and the control systems and robotics expertise of my current advisor, Dr. Naseem Daher, came out the idea of this dissertation. I would especially thank Dr. Naseem for believing in me, for his guidance, for the uncounted hours that we discussed and learned together, and for his friendly support, and Dr. Matthias for teaching me patiently in the early days. I also want to thank the chair of my PhD committee, Dr. Marwan Darwish, and the committee members, Dr. Elie Shamma, Dr. James Mynderse, and Dr. Najib Metni, for serving on my committee and for their constructive feedback that helped improve the quality of this dissertation. I want to thank my friends and the supporting AUB community who contributed to the success of this work. My special thanks goes to my wife, Alaa, for her loving support and continuous patience during these times of crisis and along the journey of my PhD studies. Finally, I want to thank my parents whose unwavering support set me on the road to this PhD a long time ago, and my parents-in-law for their unconditional care.

ABSTRACT

OF THE DISSERTATION OF

Ahmad Hassan Kourani for Doctor of Philosophy
Major: Mechanical Engineering

Title: Marine Locomotion: A Tethered UAV–Buoy with an Integrated Control System

Unmanned aerial vehicles (UAVs) are reaching offshore. In this thesis, the novel problem of a marine locomotive quadrotor UAV, which manipulates the surge velocity of a floating buoy by means of a cable, is formulated. The proposed robotic system can have a variety of novel applications for UAVs where their high speed and maneuverability, as well as their ease of deployment and wide field of vision, give them a superior advantage. In addition, the major limitation of limited flight time of quadrotor UAVs is typically addressed through an umbilical power cable, which naturally integrates with the proposed system. A detailed high-fidelity dynamic model is presented for the buoy, UAV, and water environment, in a simplified two-dimensional planar case and in the full scale three-dimensional case. Furthermore, a Directional Surge Velocity Control System (DSVCS) is proposed to allow both the free movement of the UAV around the buoy when the cable is slack, and the manipulation of the buoy's surge velocity when the cable is taut. Using a spherical coordinates system centered at the buoy, the control system commands the UAV to apply forces on the buoy at specific azimuth and elevation angles via the tether, which yields a more appropriate realization of the control problem as compared to the Cartesian coordinates, where the traditional x -, y -, and z -coordinates do not intuitively describe the tether's tension and orientation. The proposed robotic system and controller offer a new method of interaction and collaboration between UAVs and marine systems from a locomotion perspective. The system is validated in virtual high-fidelity simulation environments (MATLAB/Simulink[®] and ROS-Gazebo), which were specifically developed for this work, while considering various settings, operating conditions, and wave scenarios.

On the control systems side, a practical guideline is proposed for designing and tuning adaptive backstepping control systems by leveraging the similarity with PID control laws for a class of second-order nonlinear systems. A complete set of mathematical formulations, visual aids, and a well-structured algorithm are provided to exploit the benefits of the established link. This aims at facilitating the adoption and dissemination of advanced nonlinear control laws, namely adaptive backstepping and its variants, in more real-life and industrial applications while benefiting from the legacy of PID tuning rules. Furthermore, the proposed guideline allows for upgrading primitive PID controllers to more advanced nonlinear control system, and assessing their stability margins using the provided algorithm without much added complexity. The adaptive backstepping control law is formulated as a two degrees-of-freedom control law that combines the sum of a feedback PID control component and a feedforward model compensation component. The relationship between backstepping and PID gains is provided in the form of a third-order polynomial, and a simplified second-order one, with practical tuning guidelines. The paper culminates with devising an algorithm to design and tune backstepping gains based on the established PID similarity. The proposed control law and tuning methodology are validated on a quadrotor unmanned aerial vehicle (UAV) system in both numerical simulations as well as experimentally on a physical quadrotor UAV platform.

ABBREVIATIONS

2D	Two-Dimensional
3D	Three-Dimensional
CBNC	Cartesian-Based Nominal Controller
CoM	Center of Mass
DOF	Degrees of Freedom
DSVCS	Directional Surge Velocity Control System
FSV	Forward-Surge Velocity
FSVC	Forward-Surge Velocity Controller
GPS	Global Positioning System
ICS	Integrated Control System
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
MoI	Moment of Inertia
PID	Proportional-Integral-Derivative
PSO	Particle Swarm Optimization
PWM	Pulse-Width Modulation
RLS	Recursive Least Squares
RTK	Real-Time Kinematic
SAR	Search and Rescue
SISO	single-input single-output
SVCS	Surge Velocity Control System
UAV	Unmanned Aerial Vehicles
UGV	Unmanned Ground Vehicle
USV	Unmanned Surface Vehicles
VTOL	Vertical Takeoff and Landing

TABLE OF CONTENTS

Acknowledgements	1
ABSTRACT	2
ILLUSTRATIONS	9
TABLES	12
1 INTRODUCTION	13
1.1 Offshore UAVs	13
1.2 Limitations of offshore UAVs	14
1.3 UAVs vs USVs	14
1.4 UAV-USV Teams	14
1.5 Tethered UAVs	15
1.6 Power Solutions for UAVs	16
1.6.1 Power over Tether	16
1.6.2 Hybrid-Electric Propulsion	16
1.7 Control of Complex Robotic Systems	17
1.8 Thesis Statement	17
1.9 Contributions	19
1.10 Thesis Structure	20
2 TWO DIMENSIONAL MARINE LOCOMOTIVE UAV	21
2.1 Introduction	21
2.2 System Modeling	21
2.2.1 Preliminaries	21
2.2.2 Problem Definition	22
2.2.3 Polar Coordinates	23
2.2.4 Water Medium Model	23
2.2.5 Buoy's Dynamic Model	25
2.2.6 UAV's Dynamic Model	27
2.2.7 System Constraints	28
2.2.8 The Tethered UAV-Buoy System Model	30

2.3	Control System Design	32
2.3.1	Attainable Setpoints	32
2.3.2	Operational Modes and State Machine	34
2.3.3	Controller Design	36
2.3.4	Parameter Estimation	42
2.4	Simulations	43
2.4.1	Simulation Settings	43
2.4.2	Velocity Bounds	44
2.4.3	Simulation Scenarios	46
2.4.4	Simulation Results and Discussion	47
2.4.5	Extended Simulations	51
2.5	Practical Considerations	52
2.5.1	States Estimation	52
2.5.2	Power Considerations	54
2.5.3	Platform	55
2.5.4	Extension to 3D Space	55
2.6	Conclusion	56
3	THREE DIMENSIONAL MARINE LOCOMOTIVE UAV	57
3.1	Introduction	57
3.2	System Modeling	58
3.2.1	Preliminaries	58
3.2.2	Problem Definition	58
3.2.3	Spherical Coordinates	60
3.2.4	Water Medium Model	61
3.2.5	Buoy's Dynamic Model	63
3.2.6	UAV's Dynamic Model	65
3.2.7	System Constraints	66
3.2.8	The Tethered UAV–Buoy System Model	68
3.3	Control System Design	70
3.3.1	Operational Modes and State Machine	71
3.3.2	Controller Design	71
3.3.3	Parameter Estimation	79
3.4	Simulations	79
3.4.1	Simulation Settings	80
3.4.2	Simulation Scenarios	81
3.4.3	Simulation Results and Discussion	82
3.4.4	Extended Simulations	83
3.5	Conclusion	86

4	A PRACTICAL GUIDELINE FOR DESIGNING AND TUNING ADAPTIVE BACKSTEPPING CONTROLLERS FOR A CLASS OF SECOND-ORDER SYSTEMS BASED ON PID SIMILARITY	87
4.1	INTRODUCTION	87
4.1.1	Backstepping-PID Similarity	87
4.1.2	Motivation	88
4.1.3	Contribution	88
4.1.4	Structure	89
4.2	Problem Formulation	89
4.2.1	Preliminaries	89
4.2.2	Problem Formulation	90
4.3	Adaptive Backstepping Control System Design	91
4.3.1	Parameter Estimation Law	91
4.3.2	Controller Design	91
4.4	From Backstepping to PID Form	94
4.4.1	Expressing the Backstepping Control Law in PID Form	94
4.4.2	Comparison to Other Formulations	97
4.4.3	Implementation of the Backstepping-PID Gains Relationship	98
4.4.4	Simplified Backstepping-PID Relationships	101
4.4.5	From Backstepping to Pole Placement	105
4.4.6	Tuning Procedure	108
4.5	Simulation and Experimental Validation	108
4.5.1	Dynamic Model	109
4.5.2	Problem Formulation	110
4.5.3	Control Inputs and Reference Signals	111
4.5.4	Simulation Model Elements	112
4.5.5	Testing Scenario	112
4.5.6	Tuning Process	113
4.5.7	Simulation Results	115
4.5.8	Experimental Validation	116
4.6	Conclusion	119
5	CONCLUSIONS AND RECOMMENDATIONS	120
5.1	Research Implications	121
5.2	Recommendations for Future Work	121
A	Buoy Inertia and Damping Matrices	124
B	UAV Inertia Matrix	126
C	Spherical Coordinates Details	127

D Euler-Lagrange Formulation	129
E ROS-Gazebo Simulations	132
E.1 Terminology	132
E.2 Robot Model	132
E.3 Simulation Environment	134
E.4 Implementation	136
E.5 Simulation Scenario	138
E.6 Simulation Results	139
E.7 Future Work	141
F Repositories	142
Bibliography	143

ILLUSTRATIONS

2.1	Planar model of a quadrotor UAV pulling a floating buoy through a tether.	22
2.2	Depiction of the UAV–buoy system in violation of three constraints: (a) slack cable, (b) hanging buoy, and (c) ‘fly-over’ phenomenon.	28
2.3	UAV–buoy system operational states in the locomotion task: (a) free UAV motion around the buoy within the cable limit, used in initializing the system, (b) ready to pull forward (or backward), the UAV is in the right position to generate tension in the cable when asked to do so, (c) switching UAV’s positioning between front and rear, while following the trajectory marked in dashed blue to avoid cable entanglement, and (d) coupled and pulling forward (or backward) to manipulate the buoy surge velocity.	35
2.4	Demonstrative diagram showing the modes’ transition behavior and the buoy’s velocity tracking performance during a theoretical scenario. The dotted boundary lines govern the actions of the state machine.	36
2.5	Architecture of the Surge Velocity Control System (SVCS) for the tethered UAV–buoy system.	38
2.6	Buoy’s heave dynamic amplification, $\Delta h_{\text{amp},n}$, under excitation of different fully-developed single wave components $\{A_n, T_n\}$ of the sets $A = \{0.27, 0.61, 1.2, 1.5, 3.3\}/2\text{m}$, and $T = \{3, 4, 5, 5.7, 8\}\text{s}$. The mean buoy’s immersed height, \bar{h}_{im} , draws the boundary dome for the ‘fly-over’-free region.	45
2.7	Sample screenshots from animations of two simulation scenarios (C1 and C2) in true scale. Animations are generated via a custom-built simulator that is specifically developed to serve as a physics engine for tethered UAV–buoy locomotives.	47

2.8	States and control inputs for the simulation scenarios C1 and C2, with both the state machine-supervised surge velocity control system (SVCS) and a standard Cartesian-based nominal UAV controller (CBNC). The region in red marks when the mode is not ‘pulling’, and the region in green marks when the mode is ‘repositioning’.	48
2.9	Simulation scenarios C3 (high-frequency small-amplitude waves) and C4 (high-amplitude low-frequency waves) to illustrate the SVCS performance against extreme sea conditions. The yellow strips mark the first buoy–water separation (‘fly-over’) in each case.	50
2.10	Simulation scenario C5 (buoy’s motion damping against strong currents) to compare the SVCS and CBNC performance against extreme current conditions. The blue strip marks the crash of the CBNC-controlled UAV.	51
2.11	States and control inputs for the medium-sized system described in S3 and S4, with SVCS, under the wave conditions described in E3.	53
3.1	Three-dimensional model of a tethered quadrotor UAV–buoy system with tensile force interaction in a marine environment.	59
3.2	Unit vectors in spherical coordinates system: $\{r, \alpha, \varphi\}$. The system is left-handed, with $r \geq 0$, $\alpha \in (-\frac{\pi}{2}, \frac{\pi}{2}]$, and $\varphi \in (-\pi, \pi]$.	61
3.3	Water medium model elements and description, including: wave characteristics, wave velocity, floating object orientation, immersed height, Stokes-drift current, and lumped water surface current.	62
3.4	Architecture of the proposed Directional Surge Velocity Control System (DSVCS) for the tethered UAV–buoy system.	72
3.5	Detailed control map of the DSVCS for the tethered UAV–buoy system showing the two outer-loop (position/velocity) controllers and the inner-loop (attitude) controller.	79
3.6	States and control inputs of the directional surge velocity control system (DSVCS) for simulation scenarios C1 (a) and C2 (b).	82
3.7	The buoy’s location (top), velocity (middle), and side-slip angle (bottom) during the locomotion task in C1 (a) and C2 (b).	84
3.8	Trajectory and main states tracking for variants of the tethered UAV–buoy system in different environments: (a) E1, (b) E2 and E3, (c) E4 and E5, (d) E6 and E7. In all case, $l = 10$ m except in E6 and E7 where $l = 15$ m.	86
4.1	Coupled backstepping gains, k_1 , k_2 , and γ , for specific k_P , k_D , and k_I values: (a) at constant k_D and (b) at constant k_P . Three solution subsets exist: Subset #1 - solid lines, Subset #2 - dashed lines, and Subset #3 - dotted lines.	96

4.2	Coupled backstepping gains, k_1 and k_2 , as a function of k_P and k_D at constant k_I values ($k_I=0,1,2,4$ corresponding to four rows). The left column in (a) shows the backstepping gains at constant k_D and k_I , and the right column in (b) shows the backstepping gains at constant k_P and k_I	99
4.3	Coupled backstepping gains, k_1 and k_2 , in the simplified form, for specific k_P , k_I , and k_D values at constant k_D in (a) on the left, and at constant k_P in (b) on the right. The colored regions in red and green represent the sets of all feasible k_1 and k_2 combinations, respectively. The blue line marks the boundary between the two regions, and the grey region represents the set of unfeasible solutions. The colored stars mark the boundaries of the solution set.	103
4.4	Coupled pole locations, p_1 and p_2 , for specific k_P and k_D values at constant k_D in (a) on the left, and at constant $k'_P = k_P - \gamma$ in (b) on the right. The stars and asterisks mark the same set of the corresponding gains in Fig. 4.3.	107
4.5	Control system architecture for a quadrotor UAV.	109
4.6	Quadrotor system setup with an indoor motion capture system.	110
4.7	Desired trajectory tracking of the x and y position states (left) and their corresponding tracking errors (right) of the ICS-controlled quadrotor in each step of the tuning process executed via numerical simulations.	114
4.8	Trajectory (top) and tracking states (bottom) of the quadrotor in numerical simulations after fine-tuning the ICS per the hereby proposed Algorithm 2.	116
4.9	Demonstration of the quadrotor platform (Quanser QBall-2) lifting a payload inside the UAV testing laboratory.	117
4.10	3D trajectory tracking of the ICS-controlled quadrotor in the payload-drop experiment. The experiment is depicted at the top, and the x_u , y_u , and z_u positions and tracking errors are shown at the bottom.	118
E.1	Sample structure of the links and joint architecture in ROS [1].	133
E.2	Sample robot model for the tethered UAV–buoy system.	133
E.3	Gazebo scene with independently spawned buoy (orange) and UAV robots in a water environment with a landing pad (black).	135
E.4	Gazebo scene for the tethered UAV–buoy system at different moments during the locomotion task.	139
E.5	Gazebo simulation tracking results.	140

TABLES

2.1	Tethered UAV–buoy model parameters	43
2.2	Tracking Error and Consumed Energy	49
2.3	Velocity tracking error (cm/s) of the extended simulations. ‘F’ refers to failed cases.	54
3.1	Three-dimensional tethered UAV–buoy system model parameters	80
4.1	Backstepping-PID relationships in the literature	98
4.2	PD Gains Tuning Intervals and Effect on Backstepping Gains . .	104
4.3	Effects of Tuning PID Gains [2] and their Influence on the Back- stepping Gains	105
4.4	Tuning steps for the x and y subsystems of the quadrotor’s adap- tive backstepping controller starting from the selected PID gains.	115
E.1	Links and Joints tree of the tethered UAV–buoy robot	134

CHAPTER 1

INTRODUCTION

Aerial drones are finding their way into different sectors of the industry, including construction [3,4], agriculture [5], package delivery [6], inspection and maintenance [7], to name a few, in which drones not only independently fly in the air, but also physically interact with the environment. In terms of activities, unmanned aerial vehicles (UAVs) can move slung payloads in solo [8] or cooperatively [9] for transportation tasks, they can interact and collaborate with unmanned ground vehicles (UGVs), and they can be equipped with robotic manipulators to achieve different geometric configurations [10] or to cooperatively manipulate other objects [3, 11, 12].

A common medium for UAVs to interact with their environment is through a tether [13,14], as it can have a variety of interesting applications including the transmission of power, forces, and data. Tethered UAVs were studied for stability and control while maintaining positive cable tension in [15]. The numerous applications of UAV raise the challenge to customize the control algorithm to best fit each application, and to upgrade them with more features to withstand the conditions imposed by these applications.

1.1 Offshore UAVs

Aerial drones are well-suited for applications that meet the 4D criteria: dull, dirty, distant, and dangerous [16]. As such, the offshore oil industry and offshore wind-farms are excellent candidates for their adoption, given the potential of drones to become the go-to technology in assets inspection and infrastructure maintenance [16]. For example, traditional offshore solutions such as inspecting offshore wind-farms entail moving a vessel, which is expensive and requires a human crew, unlike the deployment of drones that can significantly save cost and time [7]. UAVs can also perform sensing jobs, place sensors, and perform on-site repairs and maintenance [7]. Furthermore, offshore applications of UAVs make it more likely for aviation authorities to permit their utilization, since they are

deployed faraway from human populations [17].

1.2 Limitations of offshore UAVs

Due to their limited power capacity and flight time, the interaction of UAVs with the marine environment is still in its early stages. Current uses are mainly limited to information gathering such as transmitting visual feedback to human operators, targeting the locations of floating objects for their retrieval [18], and generating and transmitting full area maps and path-planning for other agents to perform rescue missions [19]. Physical interaction is limited to low-power applications such as landing assistance on a rocking ship [20], power-feeding the UAV through a cable [21, 22], and sensing jobs [23].

1.3 UAVs vs USVs

Although unmanned surface vehicles (USVs) are naturally-suited robots in marine environments, UAVs can outperform them in certain aspects that make it more practical to adopt UAV-based marine solutions and applications. First, UAVs are advantageous over USVs in terms of their field of vision (bird's-eye view), ease of deployment, and maneuverability, all of which give UAVs the advantage while performing tasks in unstructured and hard-to-reach areas, and tasks that require precision and quick deployment. In addition, UAVs are especially advantageous in rivers since they can follow shorter paths above land and avoid in-water obstacles and waterfalls. Furthermore, it is more challenging to deploy and retrieve USVs since they require direct access to the water surface, whereas multirotor UAV's benefit from their vertical takeoff and landing (VTOL) capabilities to be independently deployed from anywhere. This fact highlights the advantage that UAVs have in addressing the issue of the limited and expensive free-space on offshore structures, vehicles, and coastal strips that stand to benefit from deploying aerial robotic system solutions.

1.4 UAV-USV Teams

Teams of unmanned aerial vehicles (UAVs) and unmanned surface vehicles (USVs) harness the advantages of each vehicle to form a superior robotic system. To face challenges brought by flood disasters [24], marine oil spill events [25, 26], search and rescue [27, 28], monitoring and patrol [29], and water surface cleanup [30], amongst others, heterogeneous UAV-USV systems have been employed.

The advantages of combining UAVs and USVs have been explored in literature [24, 31] based on their complementary abilities. UAVs are advantageous given

their wide field of vision and bird’s-eye view, higher maneuverability, flexibility, and ease of deployment; on the other hand, USVs are advantageous in their durability, extended running time, and large load-carrying capacity. For instance, to deal with flooding disasters, the advantage of a UAV’s wide-angle view of the environment, including the visual horizon, was combined with the long cruising ability of a USV to overcome their individual shortcomings, thus allowing the creation of a system with higher efficiency and lower risk in Search and Rescue (SAR) operations [24].

UAVs and USVs can form a dexterous robotic team in the marine environment, as evidenced by the various applications considered in the literature. A heterogeneous robotic swarm was considered in [28], where UAVs are used to generate and transmit path-planning information in addition to full area maps for USV agents to perform rescue missions. Further cooperation between the two vehicle systems can be achieved via additional sensing and control schemes, among which is the aerial visual tracking of the USV. One control solution for robust visual tracking was proposed in [29], and going further, even visual pose estimation of the USV is made possible using the UAV’s on-board camera as proposed in [27]. If they get within close proximity, UAVs can perform coordinated trajectory tracking of USVs as in [32], and they can target-track them to prioritize the camera tilt over the UAV’s motion while maintaining continuous monitoring of the USV [33]. Physical interaction is another aspect of the heterogeneous UAV-USV system. For instance, the USV can serve as a landing platform for the UAV [34], and power transmission from a USV to a UAV can be achieved through an umbilical power cable [21].

1.5 Tethered UAVs

In addition to using robotic manipulators, a UAV can interact with its environment via a tether [13, 14], as it is flexible, extendable, light-weight, and can transmit tensile forces. Recently, there have been several advances and studies on the tethered UAV problem. A tethered power system for UAVs was proposed in [22] and is now commercially available [35]. Such a system can be optimized to suit the special case of a continuously oscillating marine power station as studied in [21]. A tethered UAV is also useful in information transmission. For instance, it plays a key role in the emergency marine communication network proposed in [36], where it takes off from the communication support ship to secure a wide network coverage, while being fed with bandwidth and power to sustain its location. To address the challenge presented by wind gusts in the open marine environment, which can affect the stability of tethered systems, few works can be found on the effect of wind disturbances on the tether and the UAV’s stability [13], and on the tether vibration [37]. To simplify the analysis of the tether dynamics, the system was modeled as a multi-element body in [38].

Beyond the use of a tether for power or information transmission, it can transmit tensile forces whereby a UAV can control the magnitude and direction of the link/tether force, which was proven to be a set of differentially flat outputs of tethered UAV systems in [39]. A practical example of force transmission is found in [40], where the transportation of a payload is cooperatively performed by a team of UAVs, which was made possible through calculation of the required wrench set. The clever use of a tether even allowed a single UAV to lift a heavy object after hooking itself at high ground then activating its on-board winch [41].

1.6 Power Solutions for UAVs

1.6.1 Power over Tether

From the above literature survey and discussion, it is evident that having an integrated system that incorporates an umbilical power cable can open the door in front of a whole new level of UAV marine applications [35]. An analogous system was investigated in [42], in which an unmanned ground vehicle carries the power source while following a tethered UAV. Furthermore, an optimal length and tension design of a cable that links a UAV and USV was provided in [21]. The optimization problem minimizes both fouling (cable entanglement or jamming) and excessive downforce on the UAV during dynamic heaves, which boosts the power capacity of the UAV and simultaneously optimizes the dynamic performance of the coupled system. In addition, the employment of USVs as a landing platform has been studied in the literature; for instance, a coupled UAV–USV system was proposed in [43], where the USV is equipped with an expendable landing deck for additional safety, and the system serves as a foundation for further collaborative tasks.

1.6.2 Hybrid-Electric Propulsion

Another option to address the power limitation of multirotor UAVs is hybrid-electric propulsion, where UAVs can benefit from extended flight time due the high power density of fuel compared to lithium-ion batteries [44]. For instance, a fuel cell weighs 3.5 times less than a lithium-ion battery of the same energy capacity, which results in extended flight time in real-life testing [45]. The conceptual design of a hybrid-electric propulsion system for small UAVs was proposed in [46], which has found its way into several applications such as the GAIA 160MP heavy-lift drone that successfully completed a 100 km flight across sea [47,48], the PERIMETER 8 [49], and the Quaternium [50].

1.7 Control of Complex Robotic Systems

It is desired to keep a controller simple, yet effective, even when dealing with complex robotic systems. The effectiveness and simple architecture of proportional-integral-derivative (PID) control systems has led to their wide adoption in real-world industrial applications [51]. Furthermore, PID controllers can be tuned intuitively based on basic concepts that relate system performance characteristics to each one of its three gains [52], or systematically following certain equivalence-based gain selection and tuning techniques [53,54], amongst other methods. However, PID controllers suffer from their inability to address the needs of increased precision and complexity of advanced nonlinear systems due to their oversimplification amongst other issues [55], which lead to the continuous evolution of other control system designs.

The stability and tuning of PID controllers are critical aspects of their deployment. To automate the tuning process of control systems, several self-tuning methods have been proposed by relying on Lyapunov's stability theory, such as designing self-tuning PID controllers for a class of linear and nonlinear PID control systems [51,56]. A set of PID gains that guarantee stability was determined for single-input single-output (SISO) systems in [57], by considering the system's minimum-phase property and Lyapunov stability theory. Furthermore, the tracking problem for second-order nonlinear time-invariant systems was studied in [58] to arrive at a PID controller design based on dynamic linearization models, where stability was again proven based on Lyapunov's stability theory.

Still, PID controllers cannot meet the requirements of complex real-life systems and applications, thus more advanced control system designs continued to evolve [59]. However, the complexity and required deep theoretical knowledge of advanced control systems, among-which is backstepping control that is a common method when dealing with nonlinear systems [60], presented a challenge towards their wide adoption. Backstepping control has numerous variants that evolved to achieve additional desired characteristics on top of its basic asymptotic stability [60] and exponential stability [61]. Complex backstepping-based controllers suffer from an intricate tuning process that is required to achieve the desirable system performance. For that, optimal tuning methods have been proposed, such as the particle swarm optimization (PSO) technique [62], but such methods require exact knowledge of the system's parameters, which is not always feasible.

1.8 Thesis Statement

Leveraging the technological advances in UAVs technology in terms of robustness, accuracy, operational cost, and lately, power efficiency, and motivated by applications requiring fast action with minimal water surface disruption (e.g., [26]), we are proposing the employment of a quadrotor UAV to manipulate a passive

floating object via a cable, whereby the quadrotor performs the function of a locomotive. Aside from hybrid-electric propulsion, the umbilical power line solution naturally integrates into this system, where the cable can be used for both force and power transmission, thus adding another option to address the power limitation of UAVs. The hereby proposed problem generalizes the fixed-point tether described in [14] to a moving-frame tether, namely planar motion in the horizontal and vertical directions, and is subject to additional constraints such as maintaining contact with the water surface. The formulated problem and proposed solution pave the way in front of numerous UAV–USV interaction applications, some of which are described next.

The proposed marine locomotive UAV system can be used in coordination with nearby ships and marine structures to increase their maneuverability and decrease their response time, as well as nearshore and other water surfaces such as rivers and across waterfalls. The proposed system can help in performing a variety of tasks including rescue operations, floating objects recovery, building and inspecting marine structures, water samples collection, delicately placing and relocating marine sensors and buoys with minimal water surface disruption, fishing activities, and water surface clean-up efforts, to name a few. In this context, we are motivated by the marine application in [26], which proposed a sensor for measuring oil slick thickness during marine oil spill events. The proposed sensor is fixed to a floating buoy that is pulled by another vessel to skim the water surface. One main challenge in the proposed solution lies in the vessel’s motion ahead of the sensor, which tends to disturb the oil layer and thus reduces the measurement’s accuracy. Note that while USV-based solutions offer extended mission time, the proposed hybrid system offers a faster response with an aerial view that provides additional critical information about the mission, such as the tethered UAV system described in [63] that is used to visually detect and localize oil spills.

To address the control problem of this robotic system, we propose a special Surge Velocity Control System (SVCS). The SVCS is based on a control structure named Integrated Control System (ICS) design. The ICS is a robust adaptive backstepping control system design in an inner- and outer-loop configuration that employs the estimates of the model parameters within the control design problem for enhanced robustness using a model-based simplified estimation law. We note that even though perfectly accurate parameter estimation cannot be achieved, the close approximation of the system parameters provided by the simplified estimation law is considered valuable for the control system, since it increases the robustness bounds and simplifies the tuning process. We also formulate the *integrated* control and estimation system in a compact two-degrees-of-freedom form, which has a PID feedback component and a feedforward model compensation component, then draw the similarity link between the backstepping controller gains and the PID gains. This proposition opens the door for researchers to use well-established PID tuning rules to tackle the problem of tuning Lyapunov-based

adaptive backstepping controllers.

1.9 Contributions

This dissertation offers technical contributions to two scientific fields: marine robotics and control systems. In marine robotics, first, the novel problem of the marine locomotive UAV is formulated, which paves the way for further research into the interaction between UAVs and the marine environment. Second, the system is defined in a sea/ocean environment that accounts for the presence of gravity waves and surface current, which naturally extends to wave-free water mediums. Third, the buoy and quadrotor UAV coupled dynamics are modelled with high fidelity using the Lagrangian formulation with appropriate constraints for the tethered UAV–buoy system. Fourth, the attainable equilibrium states are derived with a proper definition of the system’s operational limits and constraints in terms of cable tension, water surface contact, and buoy velocity. Fifth, we design and validate a buoy surge velocity control system, supervised by a state machine that switches between operational modes, which results in accurate tracking performance even in the presence of disturbing waves, water currents, and feedback noise, while reducing the system’s energy consumption by maintaining a constant UAV altitude [64]. The controller relies on polar coordinates with respect to the buoy’s reference frame to realize correlated tracking, which outperforms traditional Cartesian-based and unsupervised UAV-only velocity controllers that do not lend themselves well to this application. The system is first presented in the two-dimensional (2D) vertical plane to facilitate the understanding of its complex elements and dynamics, then presented in the full-scale three-dimensional (3D) space for real life application. Lastly, we make available a physics engine that can be used for simulating tethered UAV–buoy locomotives via a custom-built simulator¹.

On the control systems front, we present a compact formulation of the adaptive backstepping control law, which consists of two components: PID feedback and feedforward model compensation, for a class of second-order nonlinear systems, which cover a wide spectrum of real-life practical systems [65]. The practicality of this work lies in the unique link that it establishes between backstepping controller gains and the PID gains, whereby tuning the adaptive backstepping gains can be achieved by standard PID tuning rules (e.g., Ziegler–Nichols). The contributions are presented in details in the introductory sections of each chapter.

¹github.com/AUBVRL/Marine-Locomotive-UAV

1.10 Thesis Structure

This dissertation is comprised of three chapters in addition to the introductory and concluding chapters. In Chapter 2, the two-dimensional (2D) problem of the tethered UAV–buoy system is introduced, with elaboration on multiple aspects of the system including, detailed modeling, buoy oscillation behavior, surge velocity controller design with state-machine supervision, simulation, and practical considerations. The purpose of the 2D model is to motivate and facilitate the understanding of this complex robotic system, while establishing general concepts and descriptions with minimal complexity. After the establishment of the basic 2D-model, the extended full-scale three-dimensional version of the tethered UAV–buoy system is presented in Chapter 3. This chapter describes the water environment in a more realistic way, and presents the models of UAV and the buoy as 6-Degrees-of-Freedom (DOF) rigid bodies. In addition the controller is extended to allow directional surge velocity control in the horizontal plane. In Chapter 4, the theory of the used ICS for tethered UAV–buoy system is presented in detail, and the findings on the PID and backstepping gains similarity are elaborated. Finally Chapter 5 concludes the dissertation and provides an outlook into future work.

CHAPTER 2

TWO DIMENSIONAL MARINE LOCOMOTIVE UAV

2.1 Introduction

For simplicity and clarity purposes before leaping into the the three-dimensional (3D) space, this chapter presents the two-dimensional (2D) planar model of the tethered UAV–buoy system, and it is structured as follows. A detailed description of the tethered UAV–buoy system dynamical model is presented in Section 2.2. The designed control system is detailed in Section 2.3. Section 2.4 presents numerical simulation results that demonstrate the validity of the derived system model and the effectiveness of the designed controller. Section 2.5 discusses some practical considerations for the implementation of the proposed system, and finally Section 2.6 concludes this chapter and paves the way in front of the 3D system that is the subject of next chapter.

2.2 System Modeling

The dynamic model of the tethered UAV–buoy system requires the integration of multiple domains including the fluid medium; the dynamics of the floating buoy, the UAV, and the cable; and the combined system of rigid bodies. In this chapter, we introduce the required subsystems to formulate the problem on hand.

2.2.1 Preliminaries

This section introduces some of the critical notations that are used throughout the paper. We let the set of positive-real numbers $\{x \in \mathbb{R} \mid x > 0\}$ be denoted as $\mathbb{R}_{>0}$, and the set of non-negative real numbers $\{x \in \mathbb{R} \mid x \geq 0\}$ be denoted as $\mathbb{R}_{\geq 0}$. Also, let s_{\bullet} , c_{\bullet} , and t_{\bullet} respectively be the sine, cosine, and tangent functions for some angle (\bullet). In addition, let $\|\cdot\|$ denote the L_2 norm. Finally, note that the

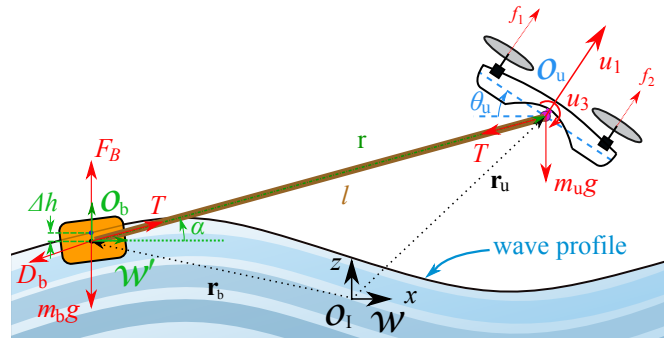


Figure 2.1: Planar model of a quadrotor UAV pulling a floating buoy through a tether.

symbol u refers to velocity's x -component entities, whereas the symbol \mathbf{u} refers to control effort entities.

2.2.2 Problem Definition

Consider the two-dimensional (2D) space in the water vertical plane where the problem is set up as shown in Fig. 2.1, and let $\mathcal{W} = \{x, z\}$ represent the inertial frame of reference whose origin, \mathcal{O}_1 , is at the local mean sea level horizontal line. Considering the tethered UAV–buoy system depicted in Fig. 2.1, the buoy is physically connected to the UAV by means of a cable of length $l \in \mathbb{R}_{>0}$, forming an angle $\alpha \in (0, \pi)$ with the positive x -axis, which is defined as the elevation angle. Let $\mathbf{r}_b = [x_b, z_b]^\top \in \mathbb{R}^2$ and $\mathbf{r}_u = [x_u, z_u]^\top \in \mathbb{R}^2$ respectively be the coordinates of the buoy's center of mass, (\mathcal{O}_b), and that of the UAV, (\mathcal{O}_u), in \mathcal{W} ; for ease of use, we set $V := \dot{x}_b$ to depict the buoy's horizontal velocity. Let \mathcal{B}_b and \mathcal{B}_u be the body-fixed reference frames of the buoy at \mathcal{O}_b , and of the quadrotor at \mathcal{O}_u , respectively. The floating buoy has a volume $\gamma_b \in \mathbb{R}_{>0}$, a bounded mass $m_b \in (0, \rho_w \gamma_b)$, and a moment of inertia $J_b \in \mathbb{R}_{>0}$ in \mathcal{B}_b ; the quadrotor UAV has a mass m_u and a moment of inertia $J_u \in \mathbb{R}_{>0}$ in \mathcal{B}_u . Also let the orientation, measured clockwise, of \mathcal{B}_b and \mathcal{B}_u with respect to \mathcal{W} be described by the angles θ_b and $\theta_u \in (-\pi, \pi]$, respectively. Let $\mathbf{V}_b = [u_b, w_b]^\top \in \mathbb{R}^2$ and $\Omega_b \in \mathbb{R}$ be the linear and angular velocities of the buoy in \mathcal{B}_b , respectively. Furthermore, let the translational rotation matrix from any body frame to \mathcal{W} be described as:

$$\mathbf{R}_\bullet = \begin{bmatrix} c_\bullet & -s_\bullet \\ s_\bullet & c_\bullet \end{bmatrix}. \quad (2.1)$$

Both the buoy and the UAV are subject to gravitational acceleration, g , and cable tension, $T, \in \mathbb{R}_{\geq 0}$. Moreover, the buoy is subjected to hydrostatic and hydrodynamic forces that are described later, and the UAV propulsion can be simplified to only include the total thrust $u_1 \in \mathbb{R}_{\geq 0}$, and a single torque that

induces a pitch motion $\mathbf{u}_3 \in \mathbb{R}$ since the motion of the system is restricted under the scope of this work to the vertical plane. Considering the relatively faster response of the UAV actuators as compared to the UAV itself, their dynamics are neglected in modeling.

Remark 1. *The scope of this work covers the manipulation of a floating buoy, thus the buoy's average density should not exceed the density of water, which is achieved with the constraint $m_b \in (0, \rho_w \Upsilon_b)$.*

Assumption 1. *The cable is inextensible; it is attached to the buoy's center of mass at one end and to the UAV's center of mass at the other via revolute joints to prevent moment transmission; and for relatively small systems considered in this work, it can be of negligible mass. Considerations for heavy cables (slung payload) are provided in Section 2.5.*

2.2.3 Polar Coordinates

We let $\mathcal{W}' = \{r', \alpha'\}$ be a rectilinear moving polar frame fixed to \mathcal{O}_b , shown in Fig. 2.1; this frame does not rotate, and it is parallel to the inertial frame \mathcal{W} . The position of the UAV in \mathcal{W} with respect to \mathcal{W}' is defined as: $\mathbf{r} = \mathbf{r}_u - \mathbf{r}_b \in \mathbb{R}^2$, and we let its coordinates in \mathcal{W}' be $\mathbf{r}' = \{r, \alpha\}$, such that:

$$\mathbf{r} = \|\mathbf{r}\|, \quad \alpha = \text{atan2}(z_u - z_b, x_u - x_b). \quad (2.2)$$

We also let the rates vector, $\dot{\mathbf{r}}'$, be:

$$\dot{\mathbf{r}}' := \begin{bmatrix} \dot{r} \\ r\dot{\alpha} \end{bmatrix} = \mathbf{R}_\alpha^\top \begin{bmatrix} \dot{x}_u - \dot{x}_b \\ \dot{z}_u - \dot{z}_b \end{bmatrix}, \quad (2.3)$$

where \mathbf{R}_α^\top is the transformation matrix that rotates vectors in \mathcal{W} to \mathcal{W}' , and we finally let the acceleration vector, $\ddot{\mathbf{r}}'$ be:

$$\ddot{\mathbf{r}}' := \begin{bmatrix} \ddot{r} - r\dot{\alpha}^2 \\ r\ddot{\alpha} + 2\dot{r}\dot{\alpha} \end{bmatrix} = \mathbf{R}_\alpha^\top \begin{bmatrix} \ddot{x}_u - \ddot{x}_b \\ \ddot{z}_u - \ddot{z}_b \end{bmatrix}. \quad (2.4)$$

2.2.4 Water Medium Model

The water medium under consideration here is the sea/ocean, where the main aspects of interest are gravity waves and water surface current.

Gravity Wave Model

Assumption 2. *In the considered problem environment, the water depth is assumed to be much larger than the wavelength of gravity waves, which are assumed to be of moderate height. This permits adopting linear wave theory in this*

work [66]. In addition, the wave direction is limited to be in the vertical (x - z) plane.

Based on Assumption 2, the water elevation variation, ζ , at time t and horizontal position x due to gravity waves is statistically described as:

$$\zeta(x, t) = \sum_n^N A_n \sin(\omega_n t - d_n k_n x + \sigma_n), \quad (2.5)$$

where $A_n, \omega_n, k_n \in \mathbb{R}_{\geq 0}$, $d_n \in \{-1, 1\}$, and $\sigma_n \in (-\pi, \pi]$ are respectively the wave amplitude, circular frequency, wave number, wave direction coefficient, and random phase angle of wave component number $n \in S_n$ with $S_n = \{1 \leq n \leq N \mid N \in \mathbb{N}\}$. Furthermore, based on Assumption 2, the wave number in deep water is given by the dispersion relation as $k_n = \omega_n^2/g$. The horizontal and vertical fluid particles' wave-induced velocities can be prescribed as [66]:

$$\begin{aligned} u_w(x, z, t) &= \sum_n^N \omega_n A_n e^{k_n z} \sin(\omega_n t - d_n k_n x + \sigma_n), \\ w_w(x, z, t) &= \sum_n^N \omega_n A_n e^{k_n z} \cos(\omega_n t - d_n k_n x + \sigma_n). \end{aligned} \quad (2.6)$$

where ω_n relates to the wave period, T_n , via $\omega_n = 2\pi/T_n$. We note that linear wave theory is not applicable near breaking waves, which usually develop nearshore and in undeveloped seas with strong winds (white horses).

Finally, the water surface plane at point $\{x, \zeta\}$ has a tilt angle in pitch with respect to the inertial frame, which is calculated by differentiating (2.5) with respect to x :

$$\theta_w(x, t) = \text{atan} \left(\sum_n^N -A_n d_n k_n \cos(\omega_n t - d_n k_n x_b + \sigma_n) \right). \quad (2.7)$$

Water Current

For brevity purposes, a simple yet comprehensive model of the water current is adopted. The water surface current, acting in the horizontal x -direction, is given as:

$$u_c = u_s + u_1, \quad (2.8)$$

where $u_1 \in \mathbb{R}$ is the lumped sum of different water current components, and $u_s \in \mathbb{R}$ is the component generated from Stokes drift [67]. The Stokes drift velocity is one component that emerges from nonlinear wave theory, and is defined as the average transport velocity of a wave over one period:

$$u_s(z) = \sum_n^N d_n A_n^2 \omega_n k_n e^{2k_n z}. \quad (2.9)$$

2.2.5 Buoy's Dynamic Model

A floating buoy is subjected to different types of forces, with the main ones being radiation, damping, and restoration forces. These forces substantially depend on the immersed volume of the buoy, $\Upsilon_{\text{im}} \in [0, \Upsilon_{\text{b}}]$, which is a function of the buoy's elevation, defined as $\Delta h = \zeta(x_{\text{b}}, t) - z_{\text{b}}(t)$. The buoyancy force in \mathcal{W} , $\mathbf{F}_{\text{B}} = [0, \rho_{\text{w}} g \Upsilon_{\text{im}}]^{\top}$, and the cable's tension in \mathcal{W}' , $\mathbf{T}'_{\text{b}} = [T, 0]^{\top}$, are assumed to be applied at \mathcal{O}_{b} . The radiation forces consist of the added mass and added damping. In this section, a detailed description of the buoy model is presented after introducing the following assumptions.

Assumption 3. *The axes of the buoy's body frame coincide with its principle axes of inertia, which is a common practice to simplify the modeling of marine vehicles [67].*

Assumption 4. *The water-buoy friction dominates the energy dissipation in the system, and the system is assumed to operate under moderate weather conditions, thus energy losses due to air drag are neglected.*

Considering the buoy dynamics in \mathcal{B}_{b} with the state vector, $\boldsymbol{\nu}_{\text{b}} = [u_{\text{b}}, w_{\text{b}}, \Omega_{\text{b}}]^{\top}$, and applying Newton's second law of motion yields:

$$\mathbf{M}'_{\text{b}} \dot{\boldsymbol{\nu}}_{\text{b}} + \mathbf{C}'_{\text{b}} \boldsymbol{\nu}_{\text{b}} + \mathbf{D}'_{\text{b}} \tilde{\boldsymbol{\nu}}_{\text{b}} + \mathbf{G}'_{\text{b}} = \boldsymbol{\tau}'_{\text{b}}, \quad (2.10)$$

where \mathbf{M}'_{b} , \mathbf{C}'_{b} , and $\mathbf{D}'_{\text{b}} \in \mathbb{R}^{3 \times 3}$ are respectively the buoy's inertia, Coriolis, and damping matrices expressed in \mathcal{B}_{b} ; the relative velocity vector is defined as $\tilde{\boldsymbol{\nu}}_{\text{b}} = \boldsymbol{\nu}_{\text{b}} - [u_{\text{c}} + u_{\text{w}} c_{\theta_{\text{b}}} - w_{\text{w}} s_{\theta_{\text{b}}}, w_{\text{w}} c_{\theta_{\text{b}}} + u_{\text{w}} s_{\theta_{\text{b}}}, 0]^{\top}$; $\mathbf{G}'_{\text{b}} \in \mathbb{R}^3$ is the gravitational forces and moments vector; and $\boldsymbol{\tau}'_{\text{b}} \in \mathbb{R}^3$ includes external forces and moments.

The inertia matrix is $\mathbf{M}'_{\text{b}} = \text{diag}(m_{\text{b}} + a_{11}, m_{\text{b}} + a_{33}, J_{\text{b}} + a_{55})$, where a_{11} , a_{33} , and $a_{55} \in \mathbb{R}_{\geq 0}$ are the surge, heave, and pitch rate components of the generalized added mass matrix. The added mass can be described as the amount of fluid that is accelerated with the body, and can be written as a function of the buoy's mass and moment of inertia. Furthermore, for low frequency motion, $a_{11} \approx 0.05m_{\text{b}}$, and $a_{33} \approx m_{\text{b}}$ [67]. Given \mathbf{M}'_{b} , the Coriolis matrix is calculated as follows:

$$\mathbf{C}'_{\text{b}} = \begin{bmatrix} 0 & 0 & (m_{\text{b}} + a_{33})w_{\text{b}} \\ 0 & 0 & -(m_{\text{b}} + a_{11})u_{\text{b}} \\ -a_{33}w_{\text{b}} & a_{11}u_{\text{b}} & 0 \end{bmatrix}. \quad (2.11)$$

The total damping term of the buoy in \mathcal{B}_{b} is expressed as:

$$\mathbf{D}'_{\text{b}} = \mathbf{D}_{\text{P}} + \mathbf{D}_{\text{S}} + \mathbf{D}_{\text{W}}, \quad (2.12)$$

where $\mathbf{D}_{\text{P}} = \text{diag}(b_{11}, b_{33}, b_{55}) \in \mathbb{R}^{3 \times 3}$ is the radiation induced potential damping matrix with surge, heave, and pitch components, and $\mathbf{D}_{\text{S}} = \text{diag}(D_{\text{S},1}, D_{\text{S},2}, D_{\text{S},3}) \in \mathbb{R}^{3 \times 3}$ is the skin friction matrix, calculated as:

$$D_{\text{S},i} = C_{\text{S},i} A_{\text{wt}} \frac{1}{2} \rho_{\text{w}} |\tilde{\boldsymbol{\nu}}_{\text{b},i}|, \quad i = 1, 2, \quad (2.13)$$

where $C_{S,i} \in \mathbb{R}_{>0}$ is a constant, $A_{\text{wt}} \in \mathbb{R}_{\geq 0}$ is the buoy's wetted area, and $D_{S,3} \in \mathbb{R}_{\geq 0}$. $\mathbf{D}_{\mathbf{w}} \in \mathbb{R}^{3 \times 3}$ is the wave drift damping matrix, which will be dropped from (2.12) since its effect is already included in the Stokes drift velocity in (2.9). Assuming that the buoy, with a mean immersed height \bar{h}_{im} , vertically oscillates in the $\{x, z\}$ plane at $\omega_{o,3} \in \mathbb{R}_{\geq 0}$, such that $\omega_{o,3} < 0.2\sqrt{g/\bar{h}_{\text{im}}}$, which is practical for the problem on hand, we have $b_{33} \approx 2m_{\text{b}}\omega_{o,3}$. Moreover, the potential damping coefficients in the horizontal plane vanish at both limits 0 and ∞ of the oscillation frequency, thus the potential damping in the x -direction is $b_{11} \approx 0$ [67].

Referring to Assumption 1, the buoy dynamics in (2.10) can be expressed in \mathcal{W} with the state vector, $\boldsymbol{\eta}_{\text{b}} = [x_{\text{b}}, z_{\text{b}}, \theta_{\text{b}}]^{\top}$, as:

$$\mathbf{M}_{\text{b}}\ddot{\boldsymbol{\eta}}_{\text{b}} + \mathbf{C}_{\text{b}}\dot{\boldsymbol{\eta}}_{\text{b}} + \mathbf{D}_{\text{b}}\tilde{\boldsymbol{\eta}}_{\text{b}} + \mathbf{G}_{\text{b}} = \boldsymbol{\tau}_{\text{b}}, \quad (2.14)$$

where \mathbf{M}_{b} , \mathbf{C}_{b} , and $\mathbf{D}_{\text{b}} \in \mathbb{R}^{3 \times 3}$ are respectively the buoy's inertia, Coriolis, and damping matrices expressed in \mathcal{W} ; the relative velocity vector is defined as $\tilde{\boldsymbol{\eta}}_{\text{b}} = [\tilde{\eta}_{\text{b},1}, \tilde{\eta}_{\text{b},2}, \tilde{\eta}_{\text{b},3}]^{\top} = \dot{\boldsymbol{\eta}}_{\text{b}} - [u_{\text{c}} + u_{\text{w}}, w_{\text{w}}, 0]^{\top}$; \mathbf{G}_{b} and $\boldsymbol{\tau}_{\text{b}}$ are respectively the vectors of the buoy's gravitational and other external forces and moments in \mathcal{W} expressed as:

$$\mathbf{G}_{\text{b}} = [0, m_{\text{b}}g, 0]^{\top}, \quad \boldsymbol{\tau}_{\text{b}} = [Tc_{\alpha}, \rho_{\text{w}}g\Upsilon_{\text{im}} + Ts_{\alpha}, F_{\text{rs}}]^{\top}, \quad (2.15)$$

where $F_{\text{rs}} = f_{\text{p}}s_{\theta_{\text{u}}}$ is the pitch restoring moment with $f_{\text{p}} \in \mathbb{R}$ being the buoy's pitch restoring moment coefficient. We also define:

$$\begin{aligned} \mathbf{M}_{\text{b}} &= (\mathbf{R}_{\theta_{\text{b}}}^{-1})^{\top} \mathbf{M}'_{\text{b}} \mathbf{R}_{\theta_{\text{b}}}^{-1}, \\ \mathbf{D}_{\text{b}} &= (\mathbf{R}_{\theta_{\text{b}}}^{-1})^{\top} \mathbf{D}'_{\text{b}} \mathbf{R}_{\theta_{\text{b}}}^{-1}, \\ \mathbf{C}_{\text{b}}\dot{\boldsymbol{\eta}}_{\text{b}} &:= \frac{1}{2} \dot{\mathbf{M}}_{\text{b}}\dot{\boldsymbol{\eta}}_{\text{b}}, \end{aligned} \quad (2.16)$$

where $\dot{\mathbf{M}}_{\text{b}} = \dot{\boldsymbol{\eta}}_{\text{b}}^{\top} (\partial \mathbf{M}_{\text{b}} / \partial \boldsymbol{\eta}_{\text{b}})$ [67]. An explicit description of \mathbf{M}_{b} and \mathbf{D}_{b} is given by:

$$\mathbf{M}_{\text{b}} = \begin{bmatrix} M'_{\text{b},11}c_{\theta_{\text{b}}}^2 + M'_{\text{b},22}s_{\theta_{\text{b}}}^2 & s_{2\theta_{\text{b}}}(M'_{\text{b},22} - M'_{\text{b},11})/2 & 0 \\ s_{2\theta_{\text{b}}}(M'_{\text{b},22} - M'_{\text{b},11})/2 & M'_{\text{b},11}s_{\theta_{\text{b}}}^2 + M'_{\text{b},22}c_{\theta_{\text{b}}}^2 & 0 \\ 0 & 0 & M'_{\text{b},33} \end{bmatrix}, \quad (2.17)$$

where $M'_{\text{b},ii}$, $i = 1, 2, 3$ are elements of the buoy inertia matrix in \mathcal{B}_{b} , \mathbf{M}'_{b} . The buoy's damping matrix in the inertial frame \mathcal{W} is defined as:

$$\mathbf{D}_{\text{b}} = \begin{bmatrix} D'_{\text{b},11}c_{\theta_{\text{b}}}^2 + D'_{\text{b},22}s_{\theta_{\text{b}}}^2 & s_{2\theta_{\text{b}}}(D'_{\text{b},22} - D'_{\text{b},11})/2 & 0 \\ s_{2\theta_{\text{b}}}(D'_{\text{b},22} - D'_{\text{b},11})/2 & D'_{\text{b},11}s_{\theta_{\text{b}}}^2 + D'_{\text{b},22}c_{\theta_{\text{b}}}^2 & 0 \\ 0 & 0 & D'_{\text{b},33} \end{bmatrix}, \quad (2.18)$$

where $D'_{\text{b},ii}$, $i = 1, 2, 3$ are elements of the buoy damping matrix in \mathcal{B}_{b} , \mathbf{D}'_{b} . We also let $M_{\text{b},ij}$, $D_{\text{b},ij}$, and $C_{\text{b},ij}$, $i, j = 1, 2, 3$ be elements of \mathbf{M}_{b} , \mathbf{D}_{b} , and \mathbf{C}_{b} , respectively.

2.2.6 UAV's Dynamic Model

We let the UAV's thrust vector in the Cartesian frame, $\mathbf{f}_{u,C} = [\mathbf{u}_x, \mathbf{u}_z]^\top$, be calculated as:

$$\mathbf{f}_{u,C} = \mathbf{R}_{\theta_u} \mathbf{f}_u, \quad (2.19)$$

with its elements being explicitly represented as:

$$\begin{aligned} \mathbf{u}_x &= \mathbf{u}_1 s_{\theta_u}, \\ \mathbf{u}_z &= \mathbf{u}_1 c_{\theta_u}. \end{aligned} \quad (2.20)$$

In the polar frame, the UAV's thrust vector, $\mathbf{f}_{u,P} = [\mathbf{u}_r, \mathbf{u}_\alpha]^\top$, is expressed as:

$$\begin{aligned} \mathbf{f}_{u,P} &= \mathbf{R}_\alpha^\top \mathbf{f}_{u,C} \\ &= \mathbf{R}_\alpha^\top \mathbf{R}_{\theta_u} \mathbf{f}_u, \end{aligned} \quad (2.21)$$

with its elements being explicitly represented as:

$$\begin{aligned} \mathbf{u}_r &= \mathbf{u}_1 s_{\alpha+\theta_u}, \\ \mathbf{u}_\alpha &= \mathbf{u}_1 c_{\alpha+\theta_u}. \end{aligned} \quad (2.22)$$

The tether's tension on the UAV expressed in \mathcal{W}' , $\mathbf{T}'_u = [-T, 0]^\top$, is applied at \mathcal{O}_u . Finally, the local wind speed that disturbs the UAV's motion is defined in \mathcal{W} as $\mathbf{U}_{wd} = [u_{wd}, 0]^\top$.

Referring to Assumption 1, and applying Newton's second law of motion on the UAV quadrotor system in \mathcal{W} with the state vector, $\boldsymbol{\eta}_u = [x_u, z_u, \theta_u]^\top$, yields:

$$\mathbf{M}_u \ddot{\boldsymbol{\eta}}_u + \mathbf{D}_u \dot{\boldsymbol{\eta}}_u + \mathbf{G}_u = \boldsymbol{\tau}_u, \quad (2.23)$$

where $\mathbf{M}_u = \text{diag}(m_u, m_u, J_u) \in \mathbb{R}_{>0}^{3 \times 3}$ and $\mathbf{D}_u = \text{diag}(D_{u,1}, D_{u,2}, D_{u,3}) \in \mathbb{R}_{\geq 0}^{3 \times 3}$ are the UAV's inertia and damping friction matrices, respectively; the UAV's relative velocity vector is $\tilde{\boldsymbol{\eta}}_u = [\tilde{\eta}_{u,1}, \tilde{\eta}_{u,2}, \tilde{\eta}_{u,3}]^\top = \dot{\boldsymbol{\eta}}_u - [u_{wd}, 0, 0]^\top$; and \mathbf{G}_u and $\boldsymbol{\tau}_u \in \mathbb{R}^3$ are vectors of the UAV's gravitational and other external forces and moments in \mathcal{W} , respectively, expressed as:

$$\mathbf{G}_u = [0, m_u g, 0]^\top, \quad \boldsymbol{\tau}_u = [\mathbf{u}_x - T c_\alpha, \mathbf{u}_z - T s_\alpha, \mathbf{u}_3]^\top. \quad (2.24)$$

The damping matrix element of interest, $D_{u,1}$, is approximated as:

$$D_{u,1} = C_{u,1} A_{cs,1}^u \frac{1}{2} \rho_a |\tilde{\eta}_{u,1}|, \quad (2.25)$$

where $C_{u,1} \in \mathbb{R}_{>0}$ is a constant, $A_{cs,1}^u \in \mathbb{R}_{\geq 0}$ is the UAV's cross-sectional area across the $\{zy\}$ -plane, and ρ_a is the air density. For more details on the quadrotor UAV model, see [68].

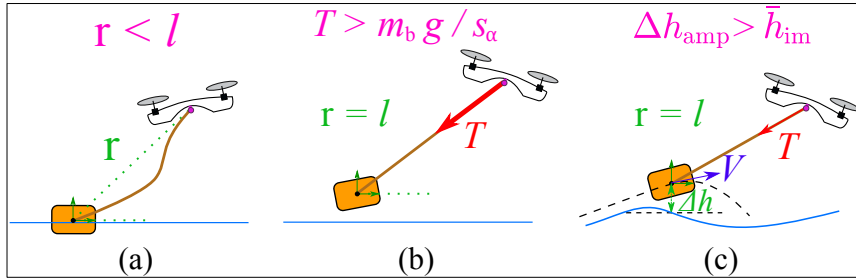


Figure 2.2: Depiction of the UAV–buoy system in violation of three constraints: (a) slack cable, (b) hanging buoy, and (c) ‘fly-over’ phenomenon.

2.2.7 System Constraints

In order to fully define the marine locomotive problem as a coupled UAV–buoy system, specific constraints are required and are presented hereafter, with their violations depicted in Fig. 2.2.

Taut-Cable Constraint

This section introduces the resulting coupled dynamics of the UAV–buoy system, which is achieved when the tether links the two bodies and holds positive tension, i.e. with a taut-cable constraint that is opposite to what is shown in Fig. 2.2a.

Definition 1. *Based on Assumption 1, the cable remains taut, i.e. it maintains tension, at time t if $r(t) = l$. The taut-cable condition is expressed as:*

$$T > 0, \quad (2.26)$$

under which the UAV–buoy system is labeled as ‘coupled’, otherwise it is labeled as ‘decoupled’. If the cable is considered extensible, then the taut cable constraint is achieved for $0 \leq r(t) - l \leq \epsilon_l$, where ϵ_l is the cable’s elastic stretching length.

With Assumption 1 and the taut-cable condition in (2.26), we have $r = l$, and the polar coordinates of the UAV can be defined with respect to the buoy’s center of mass, \mathcal{O}_b , as:

$$x_u = x_b + l c_\alpha, \quad z_u = z_b + l s_\alpha, \quad (2.27)$$

and its velocity can be obtained as:

$$\dot{x}_u = \dot{x}_b - (l s_\alpha) \dot{\alpha}, \quad \dot{z}_u = \dot{z}_b + (l c_\alpha) \dot{\alpha}. \quad (2.28)$$

No Buoy-Hanging Constraint

The buoy is required to remain at the water surface level at all times, that is, the UAV must not lift the buoy into the air by means of the cable tension alone, as

shown in Fig. 2.2b. This constraint can be forced by limiting the allowed cable tension by the following inequality, deduced from (2.14) and (2.15) as:

$$T < m_b g / s_\alpha. \quad (2.29)$$

As noted in Remark 1, the buoy floats by itself, which means that no minimum cable tension is required to maintain the buoy at the water surface.

No ‘Fly-Over’ Constraint

The buoy is required to remain in contact with the water surface at all times, that is, the UAV must not force it to ‘fly-over’ the waves, as in Fig. 2.2c, when it encounters them within a specific frequency range. This constraint is described as:

$$\Upsilon_{\text{im}} > 0, \quad (2.30)$$

which guarantees keeping the buoy partially immersed at all times. ‘Fly-over’ is a phenomenon that marks the flight of a planing hull over the waves level, thus losing contact with the water surface [69]. This phenomenon appears when the wave encounter frequency is near the resonant frequency of the hull, and is related to its Froude number [70].

To detect the occurrence of this phenomenon, the following analysis is presented. If the discontinuity in the buoyant force is neglected, the buoy’s heave dynamics can be simplified and expressed as a second-order transfer function with natural frequency, ω_b , and damping ratio, μ_b , deduced from (2.14) as:

$$\omega_b = \frac{\rho_w g A_{\text{cs},3}^b}{m_b + a_{33}}, \quad \mu_b = \frac{D_{b,33}}{2\sqrt{(m_b + a_{33})\rho_w g A_{\text{cs},3}^b}}, \quad (2.31)$$

where $A_{\text{cs},3}^b$ is the mean horizontal cross-sectional area of the buoy at the water surface level. We also define $\omega_{e,n}$, the wave encounter frequency for the n^{th} wave component, as [66]:

$$\omega_{e,n} = \omega_n - d_n \frac{\omega_n^2 V}{g}, \quad n \in S_n. \quad (2.32)$$

The ‘fly-over’ phenomenon occurs at an exciting frequency where the increase in oscillations amplitude due to dynamic magnification, Δh_{amp} , exceeds the mean immersed height of the buoy, \bar{h}_{im} , that is:

$$\Delta h_{\text{amp}} := \sum_n^N A_n \left(\frac{1}{\sqrt{(1 - \bar{\omega}_n^2)^2 + (2\mu_b \bar{\omega}_n)^2}} - 1 \right) > \bar{h}_{\text{im}}, \quad (2.33)$$

where $\bar{\omega}_n = \omega_{e,n} / \omega_b$. Further elaboration on the implications of this condition on the system modeling and performance requires knowledge of the buoy characteristics in terms of shape and weight, as well as wave characteristics in terms of height and wave length, which are presented in Section 2.4.

Conflict of Constraints

Given the constraints in (2.26) and (2.29), and the cable tension at equilibrium that can be determined from (2.14), as to be detailed in Section 2.3.1, as $\bar{T} = \bar{D}_{b,11}\bar{V}_r/c_{\bar{\alpha}}$, we consider the case when:

$$m_b g/s_{\alpha} < \bar{D}_{b,11}\bar{V}_r/c_{\bar{\alpha}},$$

in which both the taut-cable constraint and the no buoy-hanging constraints conflict with each other, which simplifies to:

$$\frac{\bar{D}_{b,11}\bar{V}_r t_{\alpha}}{m_b g} > 1. \quad (2.34)$$

The inequality in (2.34) restricts the design and application of the marine locomotive UAV system, since it can be interpreted as not being suitable for large buoy velocity-to-weight ratios.

Another constraint conflict might arise if the cable is very heavy, which results in slacking at the middle of the cable and disparate nonzero tension at each extremity of the cable. To check if a conflict exists, the tension at equilibrium can be calculated by applying the methodology described in [21] for any relative UAV–buoy position and cable weight, which is then compared to the resulting minimal tension at the buoy side with constraint (2.29).

2.2.8 The Tethered UAV-Buoy System Model

The formulation of the tethered UAV–buoy system, in its coupled form, is obtained via the Euler-Lagrange formulation, while incorporating the results of Sections 2.2.5, 2.2.6, and 2.2.7. The Lagrangian function is obtained from the kinetic ($\mathcal{K}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}_{\geq 0}$) and potential ($\mathcal{U}(\mathbf{q}) \in \mathbb{R}$) energies as $\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}) = \mathcal{K}(\mathbf{q}, \dot{\mathbf{q}}) - \mathcal{U}(\mathbf{q})$, where $\mathbf{q} = [x_b, z_b, \alpha, \theta_u, \theta_b]^{\top} \in \mathbb{R}^5$ is the generalized coordinates vector. The motion equations of the UAV–buoy system can then be derived as:

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right) - \frac{\partial \mathcal{L}}{\partial \mathbf{q}} + \frac{\partial \mathcal{P}}{\partial \dot{\mathbf{q}}} = \boldsymbol{\tau}, \quad (2.35)$$

where $\boldsymbol{\tau} \in \mathbb{R}^5$ is the external forces vector; $\mathcal{P} \in \mathbb{R}$ is a power function that captures dissipative forces, such that $\frac{\partial \mathcal{P}}{\partial \dot{\mathbf{q}}} := \mathbf{D}\tilde{\dot{\mathbf{q}}}$, where \mathbf{D} is the global damping matrix that can be formulated based on (2.16) without including a wind-induced component per Assumption 4; and $\tilde{\dot{\mathbf{q}}}$ is defined as:

$$\tilde{\dot{\mathbf{q}}} = [V_{\text{rel}}, \dot{z}_b - w_w, \dot{\alpha}, \dot{\theta}_u, \dot{\theta}_b]^{\top}. \quad (2.36)$$

To facilitate the derivation of the Euler-Lagrange equations, the kinetic energy of the system is expressed as the sum of that of the buoy and that of the UAV:

$$\mathcal{K} = \frac{1}{2} \dot{\mathbf{q}}^{\top} \mathbf{M} \dot{\mathbf{q}} := \frac{1}{2} \dot{\boldsymbol{\eta}}_b^{\top} \mathbf{M}_b \dot{\boldsymbol{\eta}}_b + \frac{1}{2} \dot{\boldsymbol{\eta}}_u^{\top} \mathbf{M}_u \dot{\boldsymbol{\eta}}_u, \quad (2.37)$$

where \mathbf{M} is the global inertia matrix of the UAV–buoy system, which can be formulated by referring to (2.28) and using the elements of \mathbf{M}_b and \mathbf{M}_u . The system’s potential energy and external forces and moments vector can be formulated based on (2.15) and (2.24) as:

$$\begin{aligned}\mathcal{U} &= m_u g(z_b + l s_\alpha) + m_b g z_b, \\ \boldsymbol{\tau} &= [\mathbf{u}_x, \mathbf{u}_z + \rho_w g \Upsilon_{\text{im}}, l \mathbf{u}_\alpha, \mathbf{u}_3, f_p s_{\theta_u}]^\top.\end{aligned}\quad (2.38)$$

Finally, the following equations of motion that result from Euler-Lagrange formulation (2.35) are obtained:

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{D}\tilde{\mathbf{q}} + \mathbf{G} = \boldsymbol{\tau}, \quad (2.39)$$

where $\mathbf{C}\dot{\mathbf{q}} := \frac{1}{2}\dot{\mathbf{M}}\dot{\mathbf{q}}$ is the Coriolis matrix with $\dot{\mathbf{M}} = \dot{\mathbf{q}}^\top \frac{\partial \mathbf{M}}{\partial \mathbf{q}}$, and the global vector of gravity forces \mathbf{G} is:

$$\mathbf{G} := \frac{\partial \mathcal{U}}{\partial \mathbf{q}} = [0, (m_b + m_u)g, m_u g l c_\alpha, 0, 0]^\top. \quad (2.40)$$

Assumption 5. *Based on assumption 4, the air effect on the tether element is neglected.*

Assumption 6. *Since floating objects are stable by design, we can safely consider that the buoy’s pitch dynamics are damped and stable, that is: $D_{b,33} \neq 0$ and $f_p < 0$. As a result, the buoy is assumed to remain tangent to the water surface at all times.*

With Assumption 6 and the dominance of waves with relatively long wave period and moderate wave height, the time derivative of the buoy pitch angle, $\dot{\theta}_b$, is small and thus its effect can be neglected in $\dot{\mathbf{M}}$, which yields a Coriolis matrix that is a function of α only.

With constraints (2.26) and (2.30) satisfied, the dynamic model equations in the coupled form are given by:

$$\begin{aligned}(M_{b,11} + m_u)\ddot{x}_b + M_{b,12}\ddot{z}_b + D_{b,11}V_{\text{rel}} + D_{b,12}\tilde{\dot{z}}_b \\ - m_u l(c_\alpha \dot{\alpha}^2 + s_\alpha \ddot{\alpha}) = \mathbf{u}_x,\end{aligned}\quad (2.41a)$$

$$\begin{aligned}(M_{b,22} + m_u)\ddot{z}_b + M_{b,21}\ddot{x}_b + D_{b,22}\tilde{\dot{z}}_b + D_{b,21}V_{\text{rel}} \\ - m_u l(s_\alpha \dot{\alpha}^2 - c_\alpha \ddot{\alpha}) + (m_u + m_b - \rho_w \Upsilon_{\text{im}})g = \mathbf{u}_z,\end{aligned}\quad (2.41b)$$

$$m_u l^2 \ddot{\alpha} + m_u l(-s_\alpha \ddot{x}_b + c_\alpha \ddot{z}_b) + m_u g(l c_\alpha) = l \mathbf{u}_\alpha, \quad (2.41c)$$

$$J_u \ddot{\theta}_u = \mathbf{u}_3, \quad (2.41d)$$

$$M_{b,33}\ddot{\theta}_b + D_{b,33}\dot{\theta}_b = f_p s_{\theta_b}, \quad (2.41e)$$

where $\tilde{\dot{z}}_b := \dot{\eta}_{b,2}$. The UAV’s position and velocity vectors can then be obtained from (2.27) and (2.28), respectively.

Remark 2. If the taut-cable constraint (2.26) is not satisfied, the system in (2.41) decouples into (2.14) and (2.23) with $T = 0$, and the polar states \mathbf{r}' , $\dot{\mathbf{r}}'$, and $\ddot{\mathbf{r}}'$ are calculated from (2.2), (2.3), and (2.4), respectively. On the other hand, if the fly-over constraint (2.30) is not satisfied, the buoy's inertia matrix in (2.17) reduces to $\mathbf{M}_b = \text{diag}(m_b, m_b, J_b)$, the buoy's damping matrix \mathbf{D}_b in (2.18) reduces to a null matrix, and f_p becomes zero.

2.3 Control System Design

The control system design problem is defined as manipulating the surge velocity of the buoy, V , to track a desired reference and to maintain the UAV's elevation, z_u , at a constant level, while ensuring that the dynamics of the UAV–buoy system remain stable and contact between the buoy and water is maintained.

2.3.1 Attainable Setpoints

The control objective is to attain a steady-state mean velocity of the buoy, (\bar{V}), and mean UAV's elevation, (\bar{z}_u), such that $\lim_{t \rightarrow \infty} (\frac{1}{t} \int z_u(t) dt, \frac{1}{t} \int V(t) dt) = (\bar{z}_u, \bar{V})$. Next, we seek to find the set of other system states, namely, $\bar{\theta}_u$ and $\bar{\gamma}_{\text{im}}$, and control inputs, \bar{u}_1 and \bar{u}_3 , that will achieve the control objective. Other nonzero mean system variables in a steady-state surge motion are: \bar{T} , $\bar{D}_{b,11}$, and $\bar{D}_{b,21}$. Note that the bar sign ($\bar{\bullet}$) refers to the mean values of the variables at equilibrium, $\bar{\gamma}_{\text{im}}$ implicitly represents \bar{z}_b , and the buoy's pitch angle, θ_b , is not considered in the setpoint analysis per Assumption 6.

Definition 2. Under specific sea conditions, namely u_c and $\zeta(A_n, \omega_n)$ with $n \in S_n$, and certain safety margins $\epsilon_T(u_c, \zeta) \geq 0$ for the cable's tension, which guarantees the coupled state of the system; and $\epsilon_\gamma \in (0, 1)$ for the buoy's immersed volume, to ensure a minimum buoy immersion that is suitable to the desired system application; the set of admissible configurations consists of the equilibrium points $(\bar{V}, \bar{\gamma}_{\text{im}}, \bar{\alpha}, \bar{\theta}_u)$, such that:

$$\bar{T} := \bar{T}(\bar{V}_{\text{rel}}, \bar{\alpha}) > \epsilon_T, \quad \bar{\gamma}_{\text{im}} := \bar{\gamma}_{\text{im}}(\bar{V}_{\text{rel}}, \bar{\alpha}) > \epsilon_\gamma \Upsilon_b, \quad (2.42)$$

where $\bar{V}_{\text{rel}} = \bar{V} - u_c$.

Assumption 7. The equilibrium state is analyzed under the no-wave condition: $A_n = 0$ with $n \in S_n$, that is, $u_w = w_w = 0$.

Theorem 1. Consider the system described in (2.41), subject to constraints (2.26) and (2.30), and the margins specified in (2.42); by Assumptions 6 and 7, the set of attainable equilibrium states is the union of all $(\bar{V}, \bar{\gamma}_{\text{im}}, \bar{\alpha}, \bar{\theta}_u)$ that satisfy:

$$\bar{\theta}_u(\bar{V}_{\text{rel}}, \bar{\alpha}) = \text{atan}\left(\frac{\bar{D}_{b,11} \bar{V}_{\text{rel}} c_{\bar{\alpha}}}{m_u g c_{\bar{\alpha}} + \bar{D}_{b,11} \bar{V}_{\text{rel}} s_{\bar{\alpha}}}\right), \quad (2.43)$$

and the steady-state thrust value and immersed volume are calculated as:

$$\bar{u}_1 = \begin{cases} \text{any } \mathbb{R}_{>0}, & \text{if } \bar{\alpha} = \frac{\pi}{2} \\ \bar{D}_{b,11} \bar{V}_{\text{rel}} / s_{\bar{\theta}_u}, & \text{otherwise,} \end{cases} \quad (2.44a)$$

$$\bar{\gamma}_{\text{im}} = \begin{cases} \frac{m_b + m_u}{\rho_w} - \frac{\bar{u}_1}{\rho_w g}, & \text{if } \bar{\alpha} = \frac{\pi}{2} \\ \frac{m_b}{\rho_w} - \frac{\bar{V}_{\text{rel}}}{\rho_w g} \left(\bar{D}_{b,11} t_{\bar{\alpha}} - \bar{D}_{b,21} \right), & \text{otherwise.} \end{cases} \quad (2.44b)$$

Given $\bar{\gamma}_{\text{im}}$, we can solve for \bar{z}_b per specific buoy geometry. In addition, the cable tension at equilibrium is a function of \bar{V}_{rel} and $\bar{\alpha}$, and expressed as:

$$\bar{T} = \begin{cases} \text{any } \mathbb{R}_{>0}, & \text{if } \bar{\alpha} = \frac{\pi}{2} \\ \bar{D}_{b,11} \bar{V}_{\text{rel}} / c_{\bar{\alpha}}, & \text{otherwise.} \end{cases} \quad (2.45)$$

Proof. The dynamic equilibrium of system (2.41) is attained when $\ddot{x}_b = \dot{z}_b = \ddot{\theta}_u = \dot{\theta}_u = \ddot{\theta}_b = \dot{\theta}_b = \ddot{\alpha} = \dot{\alpha} = 0$, and since we are considering surface motion of the buoy along with Assumption 7, we additionally have $\dot{z}_b = 0$. Thus, we conclude that $u_1 = \bar{u}_1$, and $u_3 = \bar{u}_3 := 0$, and by substituting in (2.41), we get:

$$\bar{D}_{b,11} \bar{V}_{\text{rel}} - \bar{u}_1 s_{\bar{\theta}_u} = 0, \quad (2.46a)$$

$$\bar{D}_{b,21} \bar{V}_{\text{rel}} + (m_b + m_u)g - \bar{u}_1 c_{\bar{\theta}_u} - (\rho_w \bar{\gamma}_{\text{im}})g = 0, \quad (2.46b)$$

$$m_u g c_{\bar{\alpha}} - \bar{u}_1 c_{\bar{\alpha} + \bar{\theta}_u} = 0. \quad (2.46c)$$

$\bar{\theta}_u(\bar{V}_{\text{rel}}, \bar{\alpha})$ in (2.43) is obtained by rearranging and dividing (2.46a) by (2.46c); in the case when $\bar{\alpha} \neq \frac{\pi}{2}$, \bar{u}_1 can be subsequently obtained from (2.46a). As for $\bar{\gamma}_{\text{im}}$, it can be obtained after substituting for \bar{u}_1 from (2.44a) and for $\bar{\theta}_u$ from (2.43) in (2.46b). Finally, the cable tension at equilibrium can be obtained from the first row of (2.14) by canceling the zero-valued states. Note that when $\bar{\alpha} = \frac{\pi}{2}$, the system of equations (2.46) has a solution only if $\bar{V}_{\text{rel}} = \bar{\theta}_u = 0$, while \bar{u}_1 can be any $\mathbb{R}_{>0}$ that respects the system constraints, and can be chosen to manipulate $\bar{\gamma}_{\text{im}}$ based on the first case of (2.44b). \square

Now we seek to define the set of possible attainable steady-state velocities, $S_{\bar{V}}$, under Assumption 7, that satisfy the safety margins specified in (2.42). The cable tension at equilibrium, \bar{T} , can be obtained from (2.45); then we can determine the minimum absolute surface velocity, \bar{V} , under a specific sea state, i.e. current and waves, that guarantees the taut-cable condition. In addition, the maximum absolute limit of $S_{\bar{V}}$ is attained from (2.42) and (2.44b). Finally, we get $S_{\bar{V}} = S_{\bar{V}_n} \cup S_{\bar{V}_p}$, such that:

$$\begin{aligned} S_{\bar{V}_p} &= \left(\frac{\epsilon_T c_{\bar{\alpha}}}{\bar{D}_{b,11}} + u_c, \frac{(m_b + m_u - \epsilon_\gamma \rho_w) g t_{\bar{\theta}_u}}{\bar{D}_{b,11}} + u_c \right), \text{ if } \bar{\alpha} \leq \frac{\pi}{2} \\ S_{\bar{V}_n} &= \left(\frac{(m_b + m_u - \epsilon_\gamma \rho_w) g t_{\bar{\theta}_u}}{\bar{D}_{b,11}} + u_c, \frac{\epsilon_T c_{\bar{\alpha}}}{\bar{D}_{b,11}} + u_c \right), \text{ if } \bar{\alpha} > \frac{\pi}{2}. \end{aligned} \quad (2.47)$$

Remark 3. *In practice, the maximum attainable absolute velocity can be limited by the UAV’s maximum thrust, which can be derived from (2.43) and (2.44a), and the tether’s yield strength. It is also noted that the motion across waves of various characteristics may alter the velocity bounds as to be discussed in Section 2.4.2. In addition, the violation of the buoy’s velocity upper bound can be alternatively prevented by referring to constraint (2.29) and the cable tension calculation from (2.23), and limiting the UAV’s maximum thrust such that:*

$$u_1 < \frac{m_b(1 - \epsilon_m)g}{t_\alpha s_{\theta_u}}, \quad (2.48)$$

where $\epsilon_m \in [0, 1)$ is a safety margin that represents a fraction of the buoy’s mass, and accounts for the unmodeled dynamic forces affecting the buoy’s heave motion that might violate the system constraints in (2.29) and (2.30).

2.3.2 Operational Modes and State Machine

To achieve acceleration and deceleration motions, the UAV–buoy system is required to manipulate the cable tension, switch between coupled and decoupled states, and achieve bidirectional velocity control; hence, the UAV must change its positioning with respect to the buoy back and forth. Thus, the locomotive UAV control system is to be designed to operate in both position control and velocity control modes, which necessitates the use of a state machine to achieve an autonomous performance of the UAV–buoy system. Note that a cable can only transmit tensile forces, thus allowing only pulling actions.

Next, we provide the required definitions to describe the system states, present a complete cycle of the system’s operational states to achieve the control objectives, and we introduce a state machine that allows the execution of appropriate commands.

Operational Modes

Definition 3. *The UAV’s location with respect to the buoy is assigned one of the following two configurations:*

- *We call ‘front’ the configuration at which the UAV is positioned to the front of the buoy, i.e. $\alpha \in (0, \frac{\pi}{2})$.*
- *We call ‘rear’ the configuration at which the UAV is positioned to the rear of the buoy, i.e. $\alpha \in (\frac{\pi}{2}, \pi)$.*

We let \bar{r} be the reference radial position of the UAV with respect to the buoy. The UAV–buoy system can be in one out of four operational modes shown in Fig. 2.3:

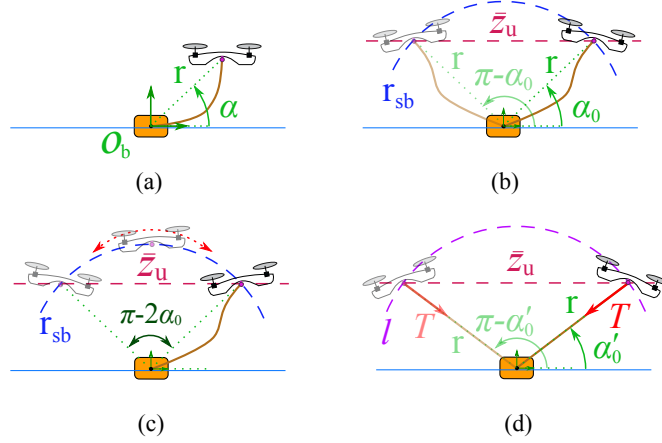


Figure 2.3: UAV–buoy system operational states in the locomotion task: (a) free UAV motion around the buoy within the cable limit, used in initializing the system, (b) ready to pull forward (or backward), the UAV is in the right position to generate tension in the cable when asked to do so, (c) switching UAV’s positioning between front and rear, while following the trajectory marked in dashed blue to avoid cable entanglement, and (d) coupled and pulling forward (or backward) to manipulate the buoy surge velocity.

- (a) We call ‘*free*’ the mode during which the UAV is allowed to move freely around the buoy, while $r < l$.
- (b) We call ‘*ready to pull*’ the mode during which the UAV is commanded to maintain a specific elevation (\bar{z}_u), and a reference standby radius, r_{sb} , which is slightly less than the cable length l to consume any cable slack. The elevation angle is α_0 if the configuration is ‘front’ and $(\pi - \alpha_0)$ if the configuration is ‘rear’.
- (c) We call ‘*repositioning*’ the mode during which the UAV moves from one side of the buoy to the other (fore/aft), travels a total arc of $(\pi - 2\alpha_0)$, while maintaining a constant reference radius with respect to the buoy, $\bar{r} = r_{sb}$, until it returns to the initial elevation, \bar{z}_u .
- (d) We call ‘*pulling*’ the mode during which the UAV is performing a pulling action on the buoy with a reference elevation, \bar{z}_u , and radius, $r = l$. The resulting elevation angle is α'_0 if the configuration is ‘front’ and $(\pi - \alpha'_0)$ if the configuration is ‘rear’.

State Machine

The UAV–buoy system is supervised by a state machine that governs switching between different control modes and commanded actions. Fig. 2.4 illustrates a

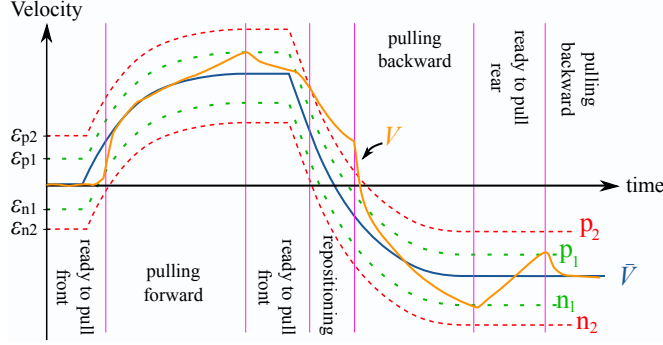


Figure 2.4: Demonstrative diagram showing the modes’ transition behavior and the buoy’s velocity tracking performance during a theoretical scenario. The dotted boundary lines govern the actions of the state machine.

typical velocity profile, with a hypothetical tracking performance, and threshold lines that govern the state machine actions to showcase the state switching mechanism. Let the threshold levels be denoted by p_1 and p_2 for the top two lines, and n_1 and n_2 for the bottom two lines. The first and second velocity error thresholds are denoted by ϵ_{th1} and ϵ_{th2} respectively.

The proposed state machine benefits from the threshold velocity lines to choose the suitable mode of action as described in Definition 3, in a way that respects the system dynamics and insures the system safety [71], with a pseudo-code provided in Algorithm 1.

2.3.3 Controller Design

The control system of the tethered UAV–buoy system consists of an outer-loop and an inner-loop controller in a cascaded structure. The outer-loop controller has two functions: 1) it controls the UAV’s relative position when the system mode is ‘free’, ‘ready to pull’, or ‘repositioning’ by controlling r and α , with setpoint $(\bar{z}_{u0}, \bar{r}_0)$; and 2) it controls the buoy’s velocity when the system mode is ‘pulling’, by regulating the elevation angle, α , and the cable tension, T , which are two flat outputs of the coupled system [39], with setpoint $(\bar{z}_{u0}, \bar{V}_0)$. On the other hand, the inner-loop controller controls and stabilizes the UAV’s pitch angle, θ_u . The proposed controller (SVCS) incorporates the state machine in Section 2.3.2, and it is designed based on polar coordinates. The SVCS architecture is presented in Fig. 2.5, which can be summarized as follows:

- A setpoint is defined and the state machine returns the system mode.
- A preprocessing unit generates 1) an elevation angle $\bar{\alpha}$ that accounts for the actual buoy elevation variation, 2) a reference radial distance \bar{r} , 3) a smoothed reference velocity \bar{V} , and 4) an estimate for the required cable tension, \hat{T}_c , to compensate for water drag, if applicable.

Algorithm 1 State machine for the locomotive UAV's control system

Input: $V, \bar{V}, configuration$

Output: $MODE$

Initialization :

$MODE \leftarrow \text{'free'}$

LOOP Process

```
1: if ( $V < \bar{V} - \epsilon_{th1}$ ) and  $configuration == \text{'front'}$  then  
2:    $MODE \leftarrow \text{'pulling'}$   
3: else if ( $V > \bar{V} + \epsilon_{th1}$ ) and  $configuration == \text{'front'}$  then  
4:    $MODE \leftarrow \text{'ready to pull'}$   
5: else if ( $V > \bar{V} + \epsilon_{th2}$ ) and  $configuration == \text{'front'}$  then  
6:    $MODE \leftarrow \text{'repositioning'}$   
7: else if ( $V > \bar{V} + \epsilon_{th1}$ ) and  $configuration == \text{'rear'}$  then  
8:    $MODE \leftarrow \text{'pulling'}$   
9: else if ( $V < \bar{V} - \epsilon_{th1}$ ) and  $configuration == \text{'rear'}$  then  
10:   $MODE \leftarrow \text{'ready to pull'}$   
11: else if ( $V < \bar{V} - \epsilon_{th2}$ ) and  $configuration == \text{'rear'}$  then  
12:   $MODE \leftarrow \text{'repositioning'}$   
13: end if  
14: return  $MODE$ 
```

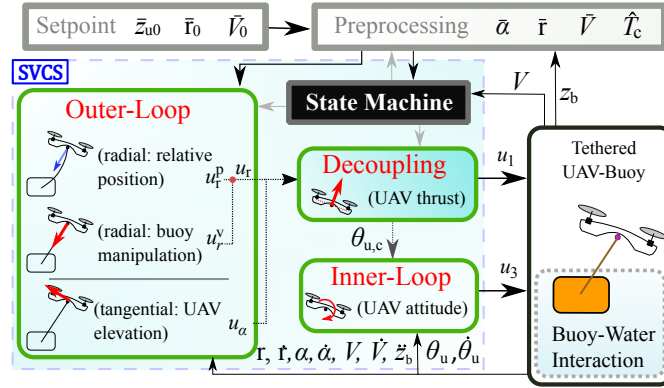


Figure 2.5: Architecture of the Surge Velocity Control System (SVCS) for the tethered UAV–buoy system.

- The outer-loop controller generates radial and tangential components of the desired force that is needed for cable tension control, u_r^v , in case of velocity control or simply radial force, u_r^p , in case of position control (radial), and the elevation angle control, u_α (tangential). Note that the switching between u_r^p and u_r^v is governed by the operational mode of the system such that:

$$\mathbf{u}_r = (1 - f_{pl}\mathcal{G}(s))\mathbf{u}_r^p + f_{pl}\mathbf{u}_r^v, \quad (2.49)$$

where $\mathcal{G}(s)$ is the transfer function of a low-pass filter, and $f_{pl} \in \{0, 1\}$ is the ‘pulling’ mode flag.

- The outer-loop controller outputs are decoupled into a command total thrust, u_1 , and a desired pitch angle, $\theta_{u,c}$.
- Finally, the inner-loop attitude controller stabilizes the pitch angle of the UAV and produces the moment command input, u_3 .

Reference Signals and Velocity Setpoint

It is desired for the UAV to maintain the same altitude during operation in order to respect aviation safety margins and save energy by reducing unnecessary vertical motion [64]. The cable length and nominal elevation angle are chosen accordingly. However, due to the vertical oscillatory motion of the buoy in accordance with the encountered waves, we must actively provide the controller with suitable elevation angle, $\bar{\alpha}$, to hold the desired UAV elevation, which is computed as:

$$\bar{\alpha} = \text{asin}((\bar{z}_u - z_b)/\bar{r}). \quad (2.50)$$

Note that the preprocessing unit outputs the supplementary angle of $\bar{\alpha}$, i.e. $\bar{\alpha} \leftarrow \pi - \bar{\alpha}$, if the system configuration is ‘rear’. Furthermore, the velocity setpoint,

\bar{V}_0 , and the radial position, \bar{r}_0 , are smoothed by second-order and fourth-order low-pass filters, respectively, in order to respect the system dynamics in terms of buoy–water friction and the UAV’s maximum thrust, thus preventing excessive coupling and decoupling of the system [39]. Finally, in order to improve the performance of the state machine, \bar{V} is sent to the controller only when the UAV is ready to enter the ‘pulling’ mode.

UAV-Buoy Relative Position Control Law

To design the relative position control law, we must refer to the UAV dynamics in \mathcal{W}' where the states of interest are explicitly expressed. By differentiating \mathbf{r} twice then multiplying it by m_u , we get:

$$m_u \ddot{\mathbf{r}} = m_u \ddot{\mathbf{r}}_u - m_u \ddot{\mathbf{r}}_b. \quad (2.51)$$

Referring to Definition 1, we must have $r(t) = l$ for the system to be coupled, that is $\dot{r} = \ddot{r} = 0$. Thus, by referring to (2.4), $\ddot{\mathbf{r}}$ reduces to the form:

$$\ddot{\mathbf{r}} = \mathbf{R}_\alpha \begin{bmatrix} -r\dot{\alpha}^2 \\ r\ddot{\alpha} \end{bmatrix}. \quad (2.52)$$

Referring to (2.23), and combining it with (2.51), then projecting along the radial and tangential directions of (2.52) by means of \mathbf{R}_α^\top , we can write the equations of motion of the UAV in \mathcal{W}' in the polar coordinates notation as:

$$\begin{aligned} m_u(\ddot{r} - r\dot{\alpha}^2) &= m_u(-\ddot{x}_b c_\alpha - \ddot{z}_b s_\alpha) - m_u g s_\alpha + \mathbf{u}_r - T, \\ m_u(r^2 \ddot{\alpha} + 2r\dot{r}\dot{\alpha}) &= m_u r(\ddot{x}_b s_\alpha - \ddot{z}_b c_\alpha) - m_u g r c_\alpha + r\mathbf{u}_\alpha. \end{aligned} \quad (2.53)$$

Consider the UAV–buoy’s relative position dynamics in (2.53) for the generic case, i.e. nonzero tension, and the UAV’s attitude dynamics in (2.23), with states vector $\mathbf{X}_1 = [r, \alpha, \theta_u]^\top$ and $\mathbf{X}_2 = [\dot{r}, \dot{\alpha}, \dot{\theta}_u]^\top$, and control input vector $\mathbf{U} = [u_r^p, u_\alpha, u_3]^\top$, and subject to unknown external disturbances like wind gusts, gravity waves, and water currents. Note that the relation between u_r^p and \mathbf{u}_r was given in (2.49). When represented as a nonlinear second-order time-varying system, the state space form is described as:

$$\begin{aligned} \dot{\mathbf{X}}_1 &= \mathbf{X}_2, \\ \dot{\mathbf{X}}_2 &= \mathbf{H} + \Phi \Theta + \mathbf{b}\mathbf{U} + \boldsymbol{\delta}, \end{aligned} \quad (2.54)$$

where $\mathbf{b} = \text{diag}(m_u, m_{ur}, J_u)^{-1}$ is the input-multiplied matrix, $\Phi = [-1/(m_u c_\alpha), 0, 0]^\top$ is the regressor vector, $\Theta = \hat{T}$ is the parameters vector; $\boldsymbol{\delta} = [\delta_r, \delta_\alpha, \delta_\theta]^\top$ is the vector of lumped system disturbances and modeling errors

across each channel, where $\hat{\boldsymbol{\delta}} = [\hat{\delta}_r, \hat{\delta}_\alpha, \hat{\delta}_\theta]^\top$ is its estimate; and $\mathbf{H} \in \mathbb{R}^3$ denotes the nonlinear and gravitational terms vector defined as:

$$\mathbf{H} = \begin{bmatrix} r\dot{\alpha}^2 - \ddot{x}_b c_\alpha - \ddot{z}_b s_\alpha - g s_\alpha \\ (-2\dot{r}\dot{\alpha} + \ddot{x}_b s_\alpha - \ddot{z}_b c_\alpha - g c_\alpha)/r \\ 0 \end{bmatrix}.$$

Assumption 8. *The modeling errors and external disturbances and their derivatives are bounded.*

Assumption 9. *The lumped error vector $\boldsymbol{\delta}$ is constant or slowly varying during a finite time interval, that is: $\lim_{t_1 < t < t_2} \dot{\delta}_\alpha, \dot{\delta}_r, \dot{\delta}_\theta \approx 0$.*

Let $\theta'_{u,c}$ be the desired UAV pitch angle to be generated by the outer-loop controller along with the total thrust command, \mathbf{u}_1 , which are calculated as:

$$u_1 = \sqrt{\mathbf{u}_\alpha^2 + \mathbf{u}_r^2}, \quad \theta'_{u,c} = \frac{\pi}{2} - \alpha - \arctan(\mathbf{u}_\alpha, \mathbf{u}_r). \quad (2.55)$$

Let $\theta_{u,c} = \theta_{u,m} \tanh(\theta'_{u,c}/\theta_{u,m})$ be a smooth and bounded version of $\theta'_{u,c}$, with $\theta_{u,m} \in (0, \frac{\pi}{2})$ being the absolute upper limit of the UAV's attitude angle. The reference state vector to be followed is defined as $\bar{\mathbf{X}}_1 = [\bar{r}, \bar{\alpha}, \theta_{u,c}]^\top$. Let the state error vector be defined as:

$$\mathbf{e}_1 = \mathbf{X}_1 - \bar{\mathbf{X}}_1. \quad (2.56)$$

The proposed control law including the radial and tangential thrust components for the outer-loop UAV's relative position controller, and the UAV's pitching torque, is defined as [72]:

$$\begin{aligned} \mathbf{U} &= \mathbf{b}^{-1} [-\mathbf{k}_P \mathbf{e}_1 - \mathbf{k}_I \mathbf{e}_1^I - \mathbf{k}_D \dot{\mathbf{e}}_1 + \ddot{\bar{\mathbf{X}}}_1 - \mathbf{H} - \Phi \Theta], \\ \dot{\mathbf{e}}_1^I &= \mathbf{e}_1 + \mathbf{k}_1^{-1} \dot{\mathbf{e}}_1, \end{aligned} \quad (2.57)$$

where $\mathbf{k}_P, \mathbf{k}_D, \mathbf{k}_I$, and $\mathbf{k}_1 \in \mathbb{R}_{>0}^{3 \times 3}$ are controller gains that are defined next.

Theorem 2. *Consider the UAV–buoy's relative position dynamics in (2.53), and the state space representation of the system in (2.54). Suppose that Assumptions 8 and 9 hold true; the control law in (2.55) and (2.57) generates the total thrust, \mathbf{u}_1 , and the UAV's desired pitch angle, $\theta_{u,c}$, that can stabilize the system, and reduce the tracking error to zero in finite time for a set of gains $\mathbf{k}_1, \mathbf{k}_2$, and $\gamma \in \mathbb{R}_{>0}^{3 \times 3}$, such that $\mathbf{k}_P = \mathbf{I}_3 + \mathbf{k}_1 \mathbf{k}_2$, $\mathbf{k}_I = \gamma \mathbf{k}_1$, $\mathbf{k}_D = \mathbf{k}_1 + \mathbf{k}_2$, with \mathbf{I}_3 being the identity matrix. If Assumption 9 does not hold, the tracking error reduces to a small region neighboring the origin in finite time.*

Proof. The backstepping control design, involving two steps, is employed, and the Lyapunov function $\mathcal{V}_1 = \frac{1}{2} \mathbf{e}_1^\top \mathbf{e}_1$ is proposed. Its derivative is expressed as: $\dot{\mathcal{V}}_1 = \mathbf{e}_1^\top \dot{\mathbf{e}}_1$. Since $\dot{\mathbf{e}}_1$ does not explicitly include a control input, we continue the

control design process for a second step. The virtual control input to stabilize \mathbf{e}_1 is defined as: $\Upsilon = \dot{\hat{\mathbf{X}}} - \mathbf{k}_1 \mathbf{e}_1$. Next, we define the virtual rates error as: $\mathbf{e}_2 = \dot{\hat{\mathbf{X}}}_1 - \Upsilon$.

By defining a second Lyapunov function:

$$\mathcal{V}_2 = \frac{1}{2} \mathbf{e}_1^\top \mathbf{e}_1 + \frac{1}{2} \mathbf{e}_2^\top \mathbf{e}_2 + \frac{1}{2} \tilde{\boldsymbol{\delta}}^\top \boldsymbol{\gamma}^{-1} \tilde{\boldsymbol{\delta}},$$

where $\tilde{\boldsymbol{\delta}} = \hat{\boldsymbol{\delta}} - \boldsymbol{\delta}$, then by differentiating and combining it with \mathcal{V}_1 , we get:

$$\begin{aligned} \dot{\mathcal{V}}_2 &= \mathbf{e}_1^\top \dot{\mathbf{e}}_1 + \mathbf{e}_2^\top \dot{\mathbf{e}}_2 + \tilde{\boldsymbol{\delta}}^\top \boldsymbol{\gamma}^{-1} \dot{\tilde{\boldsymbol{\delta}}} \\ &= \mathbf{e}_1^\top (\mathbf{e}_2 - \mathbf{k}_1 \mathbf{e}_1) + \mathbf{e}_2^\top (\mathbf{H} + \mathbf{b}\mathbf{U} + \boldsymbol{\Phi}\boldsymbol{\Theta} + \boldsymbol{\delta} - \dot{\Upsilon}) + \tilde{\boldsymbol{\delta}}^\top \boldsymbol{\gamma}^{-1} \dot{\tilde{\boldsymbol{\delta}}}. \end{aligned}$$

Next, we choose the control inputs and the lumped modeling and disturbances errors' update rates such that $\dot{\mathcal{V}}_2$ becomes negative semi-definite:

$$\mathbf{U} = \mathbf{b}^{-1} (-\mathbf{H} - \boldsymbol{\Phi}\boldsymbol{\Theta} - \hat{\boldsymbol{\delta}} + \dot{\Upsilon} - \mathbf{e}_1 - \mathbf{k}_2 \mathbf{e}_2), \quad \dot{\hat{\boldsymbol{\delta}}} = \boldsymbol{\gamma} \mathbf{e}_2, \quad (2.58)$$

and we get $\dot{\mathcal{V}}_2 = -\mathbf{e}_1^\top \mathbf{k}_1 \mathbf{e}_1 - \mathbf{e}_2^\top \mathbf{k}_2 \mathbf{e}_2$. Thus, the asymptotic convergence of \mathcal{V}_2 to zero can be obtained via Barbalat's lemma under Assumption 9. If strong wind and wave disturbances exist, meaning the violation of Assumption 9, the control law will still achieve stability and finite tracking error, which can be reduced by increasing the controller gains up to a level that overcomes the disturbances mismatch effect on $\dot{\mathcal{V}}_2$. Finally, by substituting $\dot{\Upsilon}$ and \mathbf{e}_2 in (2.58), and setting $\mathbf{e}_1^f := \hat{\boldsymbol{\delta}}(\boldsymbol{\gamma} \mathbf{k}_1)^{-1}$, the PID-like control law in (2.57) is obtained. \square

Buoy Surge Velocity Control Law

If the cable is taut, its length becomes constant and \dot{r} and \ddot{r} can be set to zero. The UAV's motion equations in \mathcal{W}' in polar coordinates notation become:

$$\begin{aligned} -m_u l \dot{\alpha}^2 &= m_u (-\ddot{x}_b c_\alpha - \ddot{z}_b s_\alpha) - m_u g s_\alpha + \mathbf{u}_r - T, \\ m_u l^2 \ddot{\alpha} &= m_u l (\ddot{x}_b s_\alpha - \ddot{z}_b c_\alpha) - m_u g l c_\alpha + l \mathbf{u}_\alpha. \end{aligned} \quad (2.59)$$

Consider the UAV dynamics in \mathcal{W}' , while following the polar coordinates notation as presented in (2.59) and (2.53), and let the buoy's velocity error be defined as $e_V = V - \bar{V}$. The buoy velocity model can be expressed in the generic case, i.e. variable radial position, as:

$$\dot{V} = H_V - T/(m_u c_\alpha) + \mathbf{u}_r^v/(m_u c_\alpha), \quad (2.60)$$

where $H_V = (r\dot{\alpha}^2 - \ddot{r} - \ddot{z}_b s_\alpha - g s_\alpha)/c_\alpha$. A control law can be designed for the surge velocity in a similar fashion as described in Section 2.3.3, with a difference

that only one step is required in the backstepping process. The resulting control law is given by:

$$\mathbf{u}_r^v = \hat{T}_c + m_u c_\alpha (-H_V + \dot{\bar{V}} - k_{PV} e_V - k_{IV} e_V^I), \quad \dot{e}_V^I = e_V, \quad (2.61)$$

where k_{PV} and $k_{IV} \in \mathbb{R}_{>0}$ are controller gains. The results of Theorem 2 relative to stability and tracking apply.

Remark 4. *Cable tension can be either directly measured (e.g. load cell) to improve the tracking performance and the system's overall safety, or it can be estimated via an observer design based on cable disturbance estimation methods. However, this internal force, T , and its estimate, \hat{T} , should not be confused with the term \hat{T}_c used in the control law (2.61), and representing the required tensile force to manipulate the buoy. One simple realization is obtained based on (2.45), such that:*

$$\hat{T}_c = D_{b,11,0} \bar{V} / c_\alpha, \quad (2.62)$$

where $D_{b,11,0} = \bar{C}_{S,1} A_{wt,0} \frac{1}{2} \rho_w |\bar{V}|$, with $A_{wt,0}$ being the zero-tension whetted area, and $\bar{C}_{S,1}$ being the surge skin friction coefficient at \bar{V} , which yields a fair, yet not very accurate, estimate. However, the proposed controller can compensate for the estimation error as will be proven next. For a sample cable tension estimation based on disturbance observation, readers are referred to [73].

Remark 5. *Practically, robust performance of the proposed control laws is guaranteed by choosing large-enough \mathbf{k}_2 gains for a wide operating range, even if Assumption 9 is violated [74].*

2.3.4 Parameter Estimation

Let $V_{\text{rel}} := \tilde{\eta}_{b,1} = V - u_c - u_w$ represent the buoy–water relative surge velocity. If the condition in (2.26) holds, the cable tension is expressed as:

$$T = \begin{cases} \left(M_{b,11} \ddot{x}_b + M_{b,12} \ddot{z}_b + D_{b,11} V_{\text{rel}} + D_{b,12} \tilde{\eta}_{b,2} \right. \\ \quad \left. + C_{b,11} \dot{x}_b + C_{b,12} \dot{z}_b \right) / c_\alpha, & |\alpha - \frac{\pi}{2}| > \epsilon_\alpha, \\ (u_1 c_{\theta_u} - m_u g - m_u \ddot{z}_u) / s_\alpha, & |\alpha - \frac{\pi}{2}| \leq \epsilon_\alpha, \end{cases} \quad (2.63)$$

where $\epsilon_\alpha \in \mathbb{R}_{\geq 0}$ is a constant that prevents singularity in a small region near $\alpha = \frac{\pi}{2}$. The cable tension can be determined from the first row of the buoy dynamics in (2.14), so that its expression is more relevant to the coupled UAV–buoy system since it shows a direct link with V_{rel} , which yields the first case of (2.63). However, this form is not applicable near the vertical cable configuration ($\alpha = \pi/2$) due to singularity, thus the actual cable tension, T , is computed via (2.23), which yields the second case of (2.63).

Table 2.1: Tethered UAV–buoy model parameters

Parameter	Value	Unit	Parameter	Value	Unit
l_b	0.8	m	m_u	1.8	kg
h_b	0.25	m	J_u	0.03	kg m ²
m_b	12.5	kg	$\theta_{u,m}$	$\pi/4$	rad
a_{11}	0.625	kg	l	7	m
a_{33}	12.5	kg	ϵ_T	5	N
b_{11}	0	Ns/m	ϵ_γ	0.05	-
b_{33}	27.5	Ns/m	ϵ_m	0.1	-
ν_w	1.787×10^{-6}	m ² /s	ρ_w	1000	kg/m ³
g	9.81	m/s ²	ρ_a	1.22	kg/m ³

2.4 Simulations

In this section, we provide simulation results that demonstrate the fidelity of the tethered UAV–buoy system model and the performance of the designed controller. We first define the settings and parameters used for the devised simulation scenarios, which include various operating conditions to validate the proposed system. To challenge the control law’s performance towards real-life implementation, the tethered UAV–buoy system model is incorporated in the simulator developed in this work, while including deviation from the described model used by the control law, including the UAV’s propellers motor dynamics, wind gusts, and non-exact state feedback.

2.4.1 Simulation Settings

To validate the proposed UAV–buoy system with the designed SVCS, a series of simulations is performed in the MATLAB Simulink[®] environment. We consider a mini-quadrotor UAV and a small-sized simplified homogeneous cuboid buoy with the dimensions and parameters listed in Table 2.1. We note that manipulating small-sized buoys, nets, debris, and surface sensors are few applications of this baseline configuration, which make it practical to implement and deploy the hereby marine locomotive UAV system. More details on the system components sizing are provided later in Section 2.5.2. The quadrotor UAV motor dynamics are modeled as a first-order low-pass filter with a time constant $\tau_m = 0.05$ s, and its total thrust and pitch torque are bounded such that $\|\mathbf{u}_1\| \leq 160$ N, and $\|\mathbf{u}_3\| \leq 11.2$ N m. The mass of the buoy is chosen such that the buoy is one quarter immersed under no external loads based on the balance between the gravitational and buoyancy forces, that is $m_b := \rho_w \gamma_b / 4$. The buoy’s immersed volume is

then defined as:

$$\gamma_{\text{im}}(\Delta h) = \begin{cases} \gamma_{\text{b}} & \text{if } \Delta h > \frac{h_{\text{b}}}{2}, \\ 0 & \text{if } \Delta h < -\frac{h_{\text{b}}}{2}, \\ \gamma_{\text{b}}/2 + l_{\text{b}}h_{\text{b}}\Delta h & \text{otherwise,} \end{cases} \quad (2.64)$$

The wetted area is calculated as:

$$A_{\text{wt}}(\Delta h) = \begin{cases} 4l_{\text{b}}h_{\text{b}} & \text{if } \Delta h > \frac{h_{\text{b}}}{2}, \\ 0 & \text{if } \Delta h < -\frac{h_{\text{b}}}{2}, \\ l_{\text{b}}h_{\text{b}} + 2l_{\text{b}}(\frac{h_{\text{b}}}{2} + \Delta h) & \text{otherwise.} \end{cases} \quad (2.65)$$

The resulting added mass and damping are calculated as described in Section 2.2.5, and their values are presented in Table 2.1. The buoy's skin friction coefficients in its body x - and z -directions can be estimated as $C_{S,i} = 0.075/(\log_{10} \text{Re} - 2)^2$, where $\text{Re} = \frac{|V_{\text{r}}|l_{\text{b}}}{\nu_{\text{w}}} \in \mathbb{R}_{\geq 0}$ is the Reynolds number, limited to turbulent flows ($\text{Re} > 10^5$), with ν_{w} being the water's kinematic viscosity [75]. To detect the coupling state of the system (coupled / decoupled), we rely on the tension estimation in the second case of (2.63).

To properly evaluate the performance of the proposed SVCS design, a Cartesian-based nominal controller (CBNC) that uses a PID control law in its outer-loop, and without supervision of a state-machine, is implemented for benchmarking purposes. It consists of a velocity (\dot{x}) controller and an elevation (z) controller, with gains $\mathbf{k}_{P,CBNC} = \text{diag}(7, 3)$, $\mathbf{k}_{I,CBNC} = \text{diag}(1.2, 1)$, and $\mathbf{k}_{D,CBNC} = \text{diag}(5, 2)$, respectively. The SVCS gains are selected as $\mathbf{k}_1 = \text{diag}(16.9, 4.6, 7.5)$, $\mathbf{k}_2 = \text{diag}(2.6, 2.4, 2.5)$, $\boldsymbol{\gamma} = \text{diag}(0.5, 0.3, 0.3)$, $k_{PV} = 25$, and $k_{IV} = 12$.

The feedback signals are assumed to be available from sensor measurements and estimations, and are modeled as follows: the UAV's pose is virtually obtained from an on-board Global Positioning System / Inertial Navigation System (GPS/INS) module, and the elevation angle and the radial distance are virtually obtained from a stereo camera system. In simulation, this is mimicked by augmenting the feedback states by a filtered Gaussian noise with the corresponding state-of-the-art accuracy of each sensor before being used by the controller. With $\text{mav}(\bullet)$ denoting the mean absolute value of the estimation error of entity (\bullet), we set $\text{mav}(\tilde{x}_{\text{u}}) = 0.02 \text{ m} = \text{mav}(\tilde{z}_{\text{u}}) = 0.02 \text{ m}$, $\text{mav}(\tilde{\theta}_{\text{u}}) = 0.5^\circ$, $\text{mav}(\tilde{\alpha}) = 0.16^\circ$, and $\text{mav}(\tilde{r}) = 0.02 \text{ m}$. Subsequently, the buoy's states are determined from (2.2), (2.3), and (2.4). More details on state estimation is given in Section 2.5.1.

2.4.2 Velocity Bounds

The constraints' bounds (ϵ_T , ϵ_V , and ϵ_m) presented in Table 2.1 are mainly based on the expected buoy–water relative velocity, in addition to the buoy's

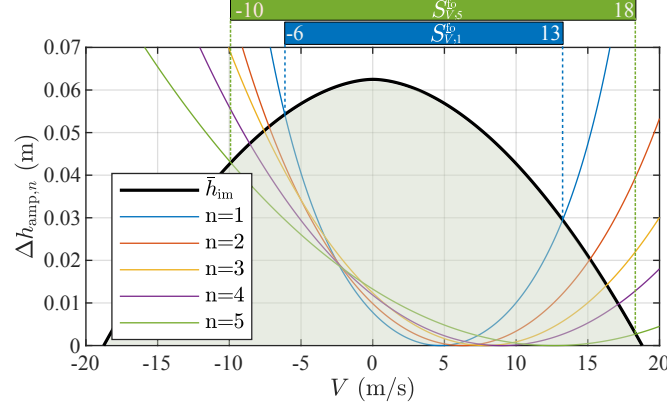


Figure 2.6: Buoy’s heave dynamic amplification, $\Delta h_{\text{amp},n}$, under excitation of different fully-developed single wave components $\{A_n, T_n\}$ of the sets $A = \{0.27, 0.61, 1.2, 1.5, 3.3\}/2\text{ m}$, and $T = \{3, 4, 5, 5.7, 8\}\text{ s}$. The mean buoy’s immersed height, \bar{h}_{im} , draws the boundary dome for the ‘fly-over’-free region.

shape, weight, and skin friction. A possible command velocity range of $S_{\bar{V}} = (-19.0, -3.1) \cup (2.1, 18.0)\text{ m s}^{-1}$ is calculated from (2.47) under no-wave condition (Assumption 7).

In the presence of waves, the feasible working velocity with no violation of constraint (2.30) reduces from above, and can be quantified by referring to (2.33) and (2.44b) as follows. We solve for \bar{V}_{im} to get \bar{h}_{im} , then find Δh_{amp} for some $(\bar{V}, \bar{\alpha})$, under different wave conditions. Fig. 2.6 provides the buoy’s heave dynamic amplification results under excitation of a single fully-developed wave component [76], with $\bar{\alpha} = 45^\circ$ and $u_c = 0$. To have a unified representation of \bar{h}_{im} , the Stokes drift effect is neglected in calculating \bar{V}_{rel} . The natural frequency of the buoy, calculated from (2.31), is $\omega_b = 8.9\text{ rad s}^{-1}$.

Fig. 2.6 can be interpreted as follows: for a given sea condition with wave amplitude and period $\{A_n, T_n\}$, the buoy hops over the waves (‘fly-over’ condition) when its horizontal (forward or backward) velocity, V , falls outside the shaded area (dome) formed by the \bar{h}_{im} curve, for a given $\Delta h_{\text{amp},n}$ (colored plots corresponding to various wave amplitudes and periods). Sample zones, where the buoy ‘fly-over’ condition does not occur, are marked on top of the figure as $S_{V,n}^{\text{fo}}$. The comprehensive results captured by Fig. 2.6 show that the system operation is direction-dependent, and they also serve as a reference for predicting the performance of the buoy in terms of heave oscillation and ‘fly-over’ phenomenon under different wave conditions, ranging from high-frequency low-amplitude waves to low-frequency high-amplitude ones, and even for superposition of various waves, as will be demonstrated in the subsequent sections. We note that the above analysis is provided for a buoy of known characteristics (Table 2.1), and serves as a guideline for the system performance.

2.4.3 Simulation Scenarios

We validate the fidelity of the derived system model and evaluate the performance of the designed controller in five cases: C1, C2, C3, C4, and C5. All cases include a wind gust of $u_{\text{wd}} = -3 \text{ m s}^{-1}$ and a water current component $u_1 = -0.5 \text{ m s}^{-1}$ (except for C5). The scenarios are described as follows:

- C1: wind gust and water current only.
- C2: wind gust, water current, and moderate waves with two wave components ($N = 2$), such that: $A_1 = 0.135 \text{ m}$, $d_1 = 1$, $T_1 = 3 \text{ s}$, and $\sigma_1 = \pi$; $A_2 = 0.75 \text{ m}$, $d_2 = 1$, $T_2 = 5.7 \text{ s}$, and $\sigma_2 = 0$.
- C3: high-frequency small-amplitude waves (head-seas), with $A_1 = 0.135 \text{ m}$, $d_1 = -1$, $T_1 = 3 \text{ s}$, and $\sigma_1 = 0$.
- C4: high-amplitude low-frequency waves (head-seas), with $A_1 = 1.65 \text{ m}$, $d_1 = -1$, $T_1 = 7 \text{ s}$, and $\sigma_1 = 0$.
- C5: buoy motion damping in the presence of a wind gust and strong water current of $u_1 = -5 \text{ m s}^{-1}$.

Note that the wave components definitions in each scenario is independent from the other scenarios. Sample visual illustrations of the environments in C1 and C2 are given in Fig. 2.7, which are generated via the custom-built simulator that we specifically developed to serve as a physics engine and provide live animations for tethered UAV–buoy locomotives. In both cases, C1 and C2, the buoy is commanded to accelerate to reach an inertial velocity $\bar{V} = 5 \text{ m s}^{-1}$, after which it gradually decelerates to 0 m s^{-1} then to -4 m s^{-1} . The desired reference mean sea level altitude is $\bar{z}_u = 5.0 \text{ m}$, which corresponds to a mean elevation angle of $\bar{\alpha}_0 = 45^\circ$. The system is initiated in the decoupled state, and its velocity is initiated to be equivalent to the zero-time water velocity via (2.6) and (2.8). Based on Assumption 2.39, the buoy’s pitch angle is set as $\theta_b = \theta_w$.

While cases C1 and C2 provide a baseline evaluation of the proposed robotic system and its controller, cases C3, C4, and C5 challenge its performance in extreme cases, i.e. under fast oscillations (C3), high amplitude undulations (C4), and large water current speeds (C5). A low-slope ramp velocity input ($V = 0.25 t$) is applied in C3 and C4 to carefully capture the performance of the system at different velocities, and head-seas are considered to emphasize and validate the universality of the buoy’s dynamic heave performance captured in Fig. 2.6. In C5, the system is applied in a river-like environment, with an objective to fix the buoy’s location with respect to the riverbank as the river streams at a high velocity. Two initial conditions are considered: 1) buoy with zero initial velocity, and 2) buoy that is initially moving at the water current speed. Note that even though the SVCS is not a position controller, C5 validates its application to such cases.

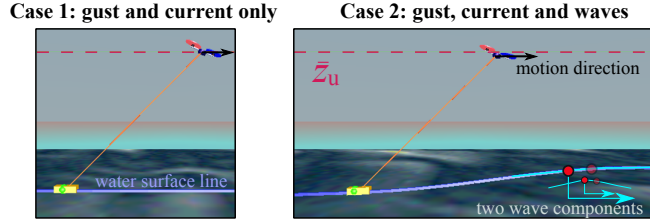


Figure 2.7: Sample screenshots from animations of two simulation scenarios (C1 and C2) in true scale. Animations are generated via a custom-built simulator that is specifically developed to serve as a physics engine for tethered UAV–buoy locomotives.

2.4.4 Simulation Results and Discussion

The simulation results for C1 and C2 are shown in Fig. 2.8a and Fig. 2.8b, respectively. In both cases, the quadrotor UAV equipped with the SVCS is able to pull the buoy at the desired velocities (V) without overshoot, with minimal fluctuations in velocity (V) and elevation (z_u), while not violating constraint (2.30) as indicated by the immersed volume plot (γ_{im}/γ_b), and without unnecessarily decoupling the system (as seen in the r subplot). The resulting commands to the UAV, u_{1c} and u_{2c} , are bounded and free of high-frequency chattering. On the other hand, the Cartesian-based controller without state machine supervision (CBNC) results in significantly larger velocity (V) and elevation (z_u) fluctuations, reaching up to 2 m s^{-1} and 2.2 m , respectively, in the wavy environment (C2).

The proposed controller adjusts the elevation angle, α , while the buoy elevation, z_b , changes – driven by the contour-following behaviour of the buoy under long waves excitation – to prevent unnecessary UAV vertical motion (z_u) as evident in Fig. 2.8b. The adjustments are such that $\bar{\alpha}$ varies in response to changes in the buoy’s elevation, z_b , which is proportional to the wave encounter frequency. It is also observed that the elevation angle (α) and pitch angle (θ_u) are smooth and stable, and exhibit small tracking error for the SVCS. We note that the reference UAV pitch angle, $\theta_{u,c}$, for the CBNC is not plotted for figure clarity purposes, since both systems possess the same inner-loop controller, in addition to the fact that CBNC has no reference elevation angle, $\bar{\alpha}$, and radial position, \bar{r} .

The SVCS-controlled UAV achieves the desired surge velocity of the buoy by adjusting the cable tension, T , in an appropriate and relatively smooth manner as seen in (T). Contrarily, the CBNC has no direct control of the cable tension and the radial position of the UAV, which leads to repeated large input pulses that deteriorate the transient performance. Finally, it is observed that the change in the immersed volume of the buoy greatly depends on the encounter frequency. It is also noticed that the buoy remains in contact with the water surface ((γ_{im}/γ_b)) for both controllers.

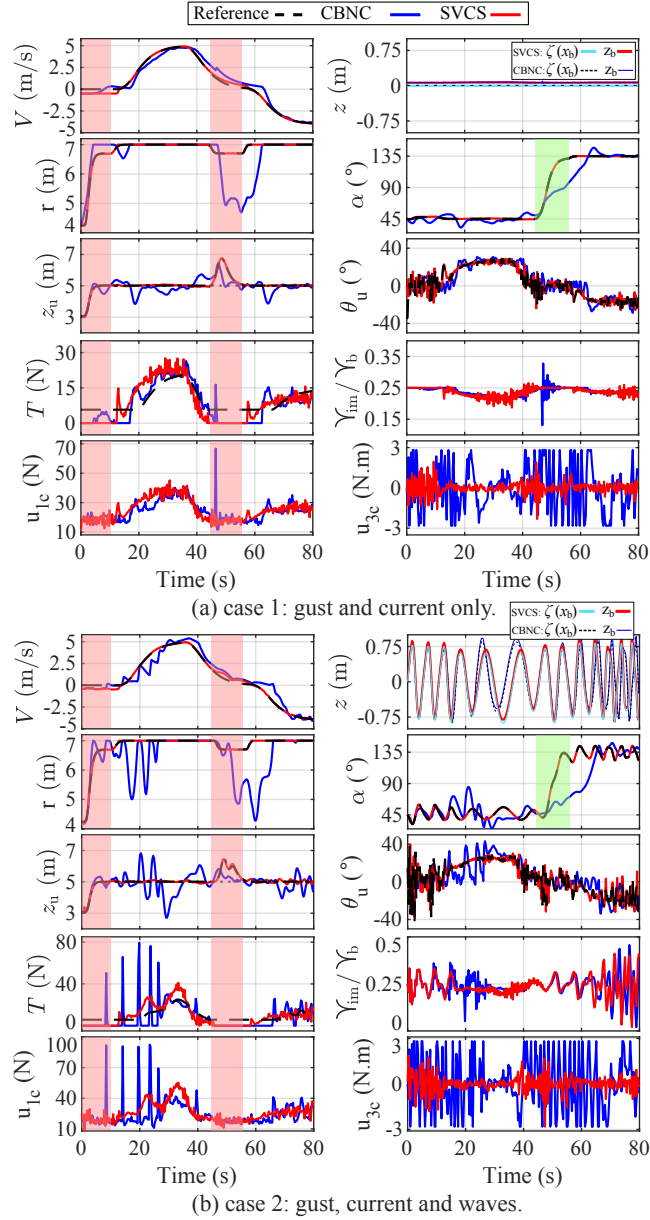


Figure 2.8: States and control inputs for the simulation scenarios C1 and C2, with both the state machine-supervised surge velocity control system (SVCS) and a standard Cartesian-based nominal UAV controller (CBNC). The region in red marks when the mode is not ‘pulling’, and the region in green marks when the mode is ‘repositioning’.

Table 2.2: Tracking Error and Consumed Energy

Case	V , mean tracking error (m/s)		z_u , mean tracking error (m)		Total consumed energy (kJ)	
	CBNC	SVCS	CBNC	SVCS	CBNC	SVCS
C1	0.36	0.05	0.28	0.03	111.8	58.9
C2	0.61	0.06	0.43	0.06	93.4	61.2

To quantify the performance of the two controllers, the trajectory tracking errors of V and z_u , and the energy consumed by the UAV (calculated per [77]) are reported in Table 2.2. The SVCS results in an average reduction in the tracking error of 88% and in energy consumption of 42% versus the CBNC.

Remark 6. *Expressing the SVCS in polar coordinates yields a correlated control performance, which means that each control channel, (u_r and u_α), independently affects one control parameter (α or V). However, this is not the case for the Cartesian-based controller (CBNC), where each of the \dot{x} - and z -control channels has a dual effect on each control parameter, which results in a degraded performance.*

In summary, the CBNC does not cope with the introduction of waves to prevent them from disturbing the system in an unpredictable manner, nor it respects the system configuration, whereas the SVCS shows its disturbance-rejection property in significantly attenuating the waves' effect even without knowing their characteristics. All of the above factors, combined, justify the design of the relatively complex SVCS for the proposed marine locomotive UAV system.

The performance of the SVCS-equipped locomotive system against high-frequency and high-amplitude waves are shown in Fig. 2.9a and Fig. 2.9b, respectively. The first separation of the buoy from the water surface occurs at $t = 21$ s and $V = 5$ m s⁻¹ in C3, and $t = 41$ s and $V = 11$ m s⁻¹ in C4, which are marked by the yellow strips in their respective subplots. The tracking accuracy in V and z_u demonstrates that the proposed SVCS performs well in the considered extreme scenarios, as long as they are within the working zones established in Fig. 2.6. Beyond these zones, i.e. after the instances marked by the yellow strips in Fig. 2.9, the buoy 'fly-over' deteriorates the system performance, which manifests as jumps of the buoy above the waves as exhibited in the z subplot.

The validation of the system in missions that require damping the buoy's motion under strong water current conditions is shown in Fig. 2.10. We notice that the SVCS is able to achieve the desired objective of fixing the buoy's location with respect to the riverbank as the river streams at a high velocity, with minimal error. The SVCS is superior to the CBNC in terms of both tracking performance and stability, where CBNC cannot prevent the quadrotor from crashing in C5.1,

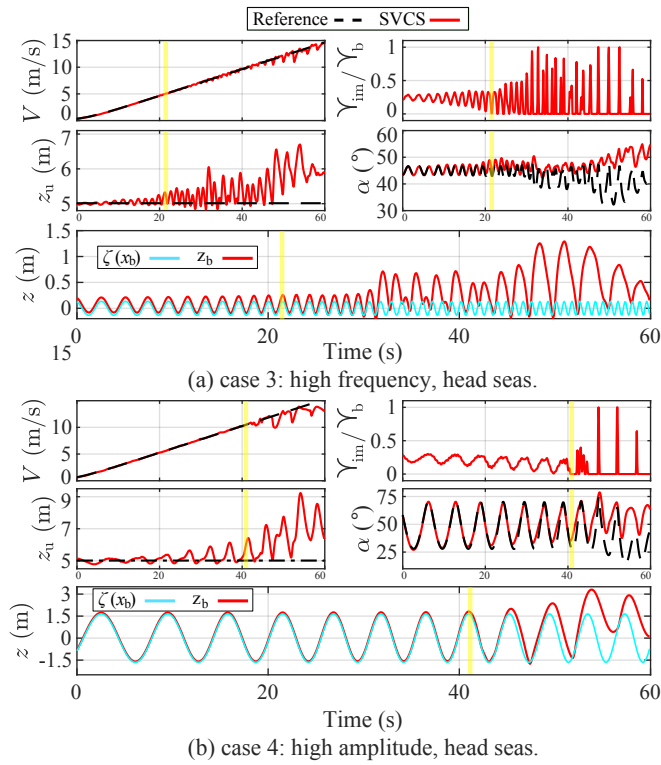


Figure 2.9: Simulation scenarios C3 (high-frequency small-amplitude waves) and C4 (high-amplitude low-frequency waves) to illustrate the SVCS performance against extreme sea conditions. The yellow strips mark the first buoy–water separation (‘fly-over’) in each case.

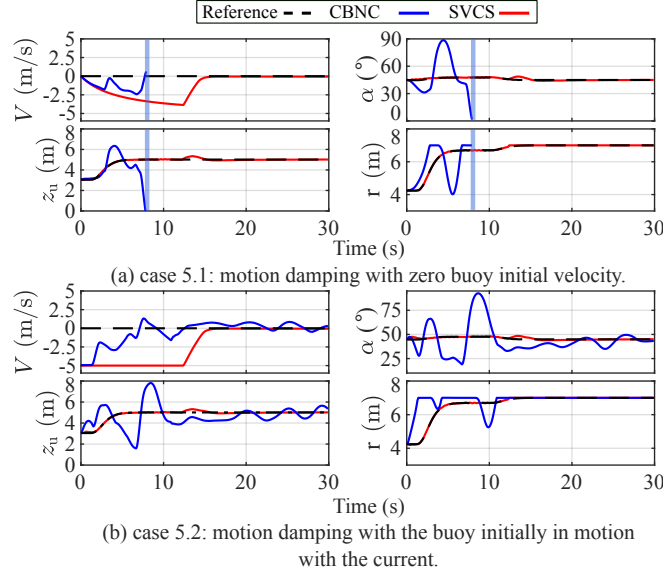


Figure 2.10: Simulation scenario C5 (buoy’s motion damping against strong currents) to compare the SVCS and CBNC performance against extreme current conditions. The blue strip marks the crash of the CBNC-controlled UAV.

which is marked by the blue strips in Fig. 2.10(a). It is also observed that the CBNC system does not respect the cable length constraint, which may lead to jamming and/or fooling of the cable.

2.4.5 Extended Simulations

For further validation of the system, we present a series of simulations that vary a) the sizing of the system components (UAV, buoy, and cable) and b) the waves environment. Let S_i represent the physical system index where $i = \{1, 2, 3, 4\}$, and let E_j represent the environment index where $j = \{1, 2, 3\}$, the variations are captured as follows:

(a) System size variants (UAV, buoy, and cable):

1. Baseline mini-quadrotor and a small-sized buoy, as described in Table 2.1, with cable length $l = 7$ m (S1) and $l = 12$ m (S2).
2. Medium-sized system with the GAIA-160 multirotor [47] that has the following characteristics: $m_u = 19.5$ kg, and $J_u = 1.52$ kg m²; and a medium-sized buoy of the following characteristics: $h_b = 0.3$ m, $l_b = 1.5$ m, and $m_b = 33.75$ kg; with cable lengths of $l = 15$ m (S3) and $l = 20$ m (S4).

(b) Wave environment variants:

Here, we consider the same environment as C2, but with different combinations of wave components, such that:

- E1. ($N = 2$), $A_1 = 0.305$ m, $d_1 = 1$, $T_1 = 4$ s, and $\sigma_1 = \pi$; $A_2 = 0.6$ m, $d_2 = 1$, $T_2 = 5$ s, and $\sigma_2 = 0$.
- E2. ($N = 2$), $A_1 = 0.305$ m, $d_1 = 1$, $T_1 = 4$ s, and $\sigma_1 = \pi$; $A_2 = 1.65$ m, $d_2 = 1$, $T_2 = 8$ s, and $\sigma_2 = 0$.
- E3. ($N = 2$), $A_1 = 0.6$ m, $d_1 = 1$, $T_1 = 5$ s, and $\sigma_1 = \pi$; $A_2 = 1.65$ m, $d_2 = 1$, $T_2 = 8$ s, and $\sigma_2 = 0$.

For brevity purposes, only the velocity tracking error of the extended simulation validation is reported in Table 2.3, albeit the full state results of the medium-sized system are given in Fig. 2.11. Failed cases, which entail the UAV’s height dropping below the water surface level, are designated with the letter ‘F’ in Table 2.3, and the remaining simulation scenarios resulted in stable tracking performance that is similar to Fig. 2.8. The complete set of simulation data and results can be found in the project’s online repository.

From Table 2.3, it is noticed that the baseline system is only able to handle the wave environment E1, and fails to deal with the high amplitude waves of E2 and E3. The performance is enhanced by using a longer cable, as seen from the results of S2 and S4, as compared to S1 and S3, respectively. Furthermore, the medium-sized system with a longer cable achieves better performance in terms of tracking velocity, as seen by comparing S3 and S4 to S1 and S2, respectively. Additionally, Fig. 2.11 demonstrates a better performance of S4 in the high-amplitude wave environment E3, as can be seen from the α , r , and z_u subplots. We can also notice smaller spikes in thrust, u_{1c} , and tension, T , when a longer cable is used (S4), which is due to lower variations in the elevation angle, α .

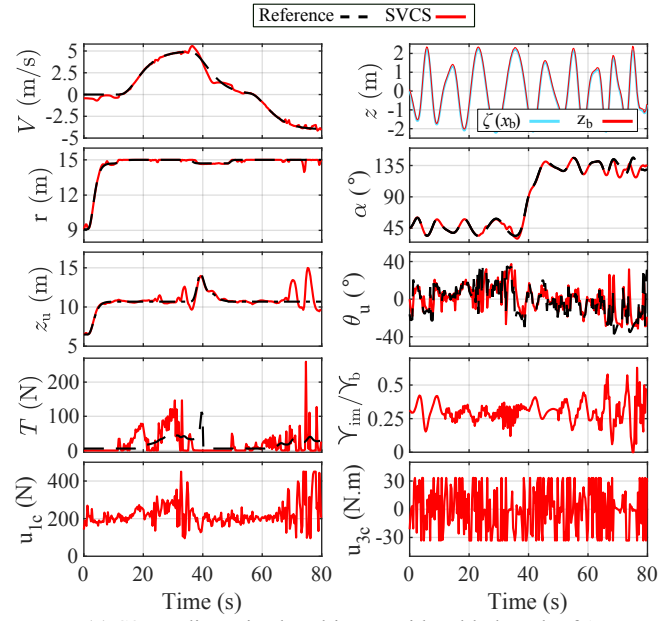
The extended simulations demonstrate that the UAV–buoy system can be sized to fit specific physical environmental constraints. For instance, a longer cable allows the system to handle higher amplitude waves, but at the expense of increased cable weight and higher flying altitudes.

2.5 Practical Considerations

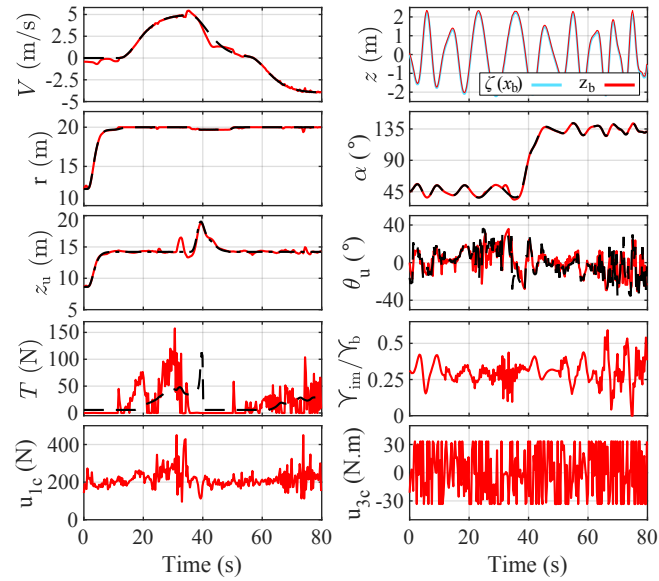
For the proposed system to lend itself well to physical implementation, we target in this Section critical aspects that are essential to experimentally validate the proposed system, in preparation for its deployment in real-life.

2.5.1 States Estimation

The SVCS requires the following states for feedback: r , \dot{r} , \ddot{r} , α , $\dot{\alpha}$, θ_u , $\dot{\theta}_u$, V , \ddot{x}_b , and \ddot{z}_b . Unlike most applications of the tethered UAV problem, the tether



(a) S3: medium-sized multirotor with cable length of 15 m.



(b) S4: medium-sized multirotor with cable length of 20 m.

Figure 2.11: States and control inputs for the medium-sized system described in S3 and S4, with SVCS, under the wave conditions described in E3.

Table 2.3: Velocity tracking error (cm/s) of the extended simulations. ‘F’ refers to failed cases.

Case	E1	E2	E3
	$A_1 = 0.305$ m $A_2 = 0.6$ m	$A_1 = 0.305$ m $A_2 = 1.65$ m	$A_1 = 0.6$ m $A_2 = 1.65$ m
S1	14.9	F	F
S2	15.2	22.1	23.1
S3	14.4	21.0	17.7
S4	20.7	18.7	16.6

of the UAV–buoy system is not anchored to a fixed point as in [14, 39, 78]; furthermore, knowing that the system is allowed to decouple, resulting in a slack cable, observer-based methods that target system state estimation in the taut-cable case, as in [71], cannot be solely employed for state feedback. However, since the UAV–buoy system works above the water surface, it is practical to assume that GPS coverage is available, which allows for UAV pose estimation in the inertial frame using a GPS sensor and an inertial measurement unit (IMU) that is equipped with a magnetometer [71]. We also note that if the system is designed to operate in the vicinity of a marine structure, Real-Time Kinematic (RTK) GPS solutions can also be utilized to attain more accurate inertial state estimation [79]. To solve the state estimation problem when the cable is slack, the UAV can be equipped with a stereo camera to detect and estimate the buoy location in the camera frame using special-purpose algorithms [80, 81], from which the UAV’s relative radial coordinates to the buoy (r and α), and the buoy’s velocity (V) can be estimated [82]. We note that a monocular camera can provide adequate accuracy for the control problem on hand only if the buoy’s dimensions are known a priori [83]. As for laser-baser sensory equipment, they are susceptible to sun rays exposure and water surface refraction, which deems them unsuitable for such applications. Last but not least, using encoders placed on the UAV can help with measuring the cable’s length and elevation angle in the taut-cable case, and a force sensor (e.g. load cell) allows measurement of the cable’s tension [71], thus providing the control system with additional information to improve its performance.

2.5.2 Power Considerations

To make the system more energy efficient, it can be designed to allow the UAV to land on the buoy, or to float directly on the water surface, during long standby periods [43]. Several multicopter platforms with extended flight times have been designed using hybrid-electric technology [84], which answers the possible need for

large power of the hereby proposed system. Another alternative to further extend the work-time of the system is to integrate an umbilical power cable within the tether, which is commercially available and integrated with multirotor UAVs [35]. Furthermore, using an umbilical power line with power banks stationed on the buoy can be more efficient than increasing the on-board power capacity of the UAV under specific conditions related to system design and mission requirements. Also, a small relative buoy–water velocity and a streamlined buoy shape can result in better energy efficiency.

We note that in case of large umbilical power transmission cables, their mass cannot be neglected and must be compensated for in the control law of the coupled dynamics model as in [12] and [85]; and in the decoupled form, the UAV controller should be modified to compensate for the cable mass as in [86]. For illustration purposes, in the example presented in this work, a 7 m long 1000 W power cable weighs less than 0.5 kg, which adds to the main tensile wire of similar weight to get a total of about 1 kg.

To get a better assessment of the power and energy needs, we present a basic and practical power solution for the proposed tethered UAV–buoy system. To address the system’s power requirement, the buoy can be equipped with a gasoline power generator with a fuel tank [87] or a battery power bank [88]. Assuming an average UAV thrust of 40 N and a flight time of 30 min, the required energy would be about 5 MJ as per [77], which can be supplied either by 0.15 L of gasoline or 9 kg of Li-Po batteries. Note that a small 1000 W generator with its accessories weighs about 3 kg [87]. Given the dimensions and overall weight of the baseline buoy with specifications given in Table 2.1, the generator-based solution can easily fit within the buoy, while the Li-Po batteries-based solution cannot. Finally, we note that the choice of a suitable power solution is naturally application-specific.

2.5.3 Platform

The locomotive UAV system can be deployed from ships and marine structures. It can be an independent system, and if designed and equipped to work autonomously, it can link itself to the target floating object using an on-board cable and perform manipulation afterward. Such designs have higher mobility, easier deployability, and independence from potentially-bulky buoys.

2.5.4 Extension to 3D Space

This subsection defines the problem of the tethered UAV–buoy problem with reasonable assumptions that limit the representation of the system to the 2D vertical space, which facilitates the understanding of the problem. The extension of this work to the 3D space is attainable, given that the proposed system is a combination of currently available technologies that have been employed in various real-life applications, which include hybrid UAVs [84], tethered UAVs

[13, 22, 89], umbilical power cables to UAVs [35, 88], integration of the umbilical power cable to USVs [21, 63], visual pose stabilization of tethered UAVs [90], and finally extended oversea multirotor UAV flight [48].

On the dynamic modeling and control system design side, we note that in the 3D space, the wave and current models should be modified to be multi-directional [66, 67], which induce lateral disturbances on the system that the controller needs to be upgraded to account for. From a motion perspective, the UAV could experience new patterns of motion in 3D space, other than moving backward and forward, to compensate for sideways disturbances and follow curved trajectories. Last but not least, the 2D polar coordinates used in the system definition and control law derivation need to be extended to the 3D space as done in [71, 91], which merely increases the mathematical complexity of the problem.

2.6 Conclusion

The novel problem of a marine locomotive UAV system is defined, in which a quadrotor UAV is tethered to a floating buoy to control its surge velocity. The system dynamics are separately modeled for each subsystem including the water medium, the buoy, and the UAV, then combined via the Euler-Lagrange formulation. The attainable setpoints and constraints of the proposed system are defined, then a precision motion control system (SVCS) is designed to manipulate the surge velocity of the buoy within certain limits, which require maintaining the cable in a taut state and keeping the buoy in contact with the water surface. A simulation environment is defined, and the proposed SVCS is validated and compared to a nominal Cartesian-based UAV controller, while showing superior tracking performance and disturbance rejection in various wave, surface current, and wind conditions.

The proposed system paves the way in front of a wide variety of novel marine applications for multirotor UAVs, where their high speed and maneuverability, as well as their ease of deployment and wide field of vision, give them a superior advantage. It best suits applications that require remote and fast manipulation with minimal water surface disruption.

CHAPTER 3

THREE DIMENSIONAL MARINE LOCOMOTIVE UAV

3.1 Introduction

The 2D-planar model of the tethered UAV–buoy system that was proposed in Chapter 2 presented a thorough analysis of the subsystem’s dynamics and the systems’ stability, the effect of waves, the proposed working bounds, steady-state values of the system states, and several practical considerations for implementing the system [31]. However, the 2D tethered UAV–buoy model does not capture the complete dynamics of the real system. For instance, the 2D model has limited representations of the Euler angles of both the UAV and the buoy, it does not encompass composite waves and currents with varying directions, and it does not allow the system to follow trajectories that are in the the three-dimensional (3D) space. In addition, the orientation of the buoy is not captured in the 2D-planar model, which can deviate from the actual direction of the tether. Furthermore, the Surge Velocity Control System (SVCS) designed in Chapter 2 does not include the tether’s azimuth angle within its controlled state variables, thus the controller is unable to provide directional manipulation of the buoy in the horizontal plane if deployed on a 3D model. Therefore, for real-world applications, it is necessary to extend the 2D planar model of the tethered UAV–buoy system to the 3D space, so that it includes all position and orientation states of the UAV, the buoy, and the tether, as well as the surface water model.

The contributions of this chapter are presented next. First, we extend the model of the 2D-planar tethered UAV–buoy system dynamics to the 3D space by incorporating the full six degrees-of-freedom (6-DOF) rigid-body model for each of the UAV and buoy, and a 2-DOF model of the tether, to arrive at a 11-DOF dynamical model that allows for a more realistic representation of the system’s physics. Second, the ocean/sea environment is modeled to include not only the surface water current, but also the full effect of oscillating gravity waves,

even in the vertical direction. Third, a rigorous mathematical derivation of the composite system is presented through the Lagrangian formulation, while defining the required conditions and constraints to regulate the system’s motion. Fourth, a spherical coordinates-based dual controller is designed to allow the free movement of the UAV around the buoy while the tether is slack, and to manipulate the buoy’s surge velocity while the tether is taut. This controller allows the UAV to apply directional tension through the tether on the buoy so that it can trigger motion in all directions of the water surface.

The remainder of this Chapter is organized as follows. Section 3.2 presents a multi-physics description of the 3D tethered UAV–buoy system components and the water environment. Section 3.3 introduces the control system design for controlling the relative UAV–buoy position and the buoy’s surge velocity. Comprehensive numerical co-simulation results of the derived system model and designed control system are presented in Section 3.4. Section 3.5 concludes the chapter and identifies future development tracks.

3.2 System Modeling

The heterogeneous marine robotic system of a tethered UAV–buoy system has multi-physics elements, which must be integrated with a proper definition of the marine environment that it operates in. This section defines the system components and their interconnection including the water environment, the USV/buoy, the tether, and the UAV, and it finally culminates with the coupled dynamical model of the tethered UAV–buoy system.

3.2.1 Preliminaries

Before defining the problem, we briefly specify general notations to support the model derivation process. For some angle (\bullet) , we let c_\bullet , s_\bullet , and t_\bullet respectively be the cosine, sine, and tangent functions. Also, we define $\mathbb{R}_{>0}$ to be the set of positive-real numbers such that $\{x \in \mathbb{R} \mid x > 0\}$, and $\mathbb{R}_{\geq 0}$ to be the set of non-negative real numbers such that $\{x \in \mathbb{R} \mid x \geq 0\}$.

3.2.2 Problem Definition

Consider the 3D space above the water surface in which the problem is defined as shown in Fig. 3.1. Let \mathcal{O}_I be the origin of the inertial frame of reference, $\mathcal{W} = \{x, y, z\}$, located at the horizontal plane of the local mean sea level. Let $\mathbf{r}_u = \{x_u, y_u, z_u\} \in \mathbb{R}^3$ and $\mathbf{r}_b = \{x_b, y_b, z_b\} \in \mathbb{R}^3$ be the coordinates of the quadrotor UAV’s center of mass, (\mathcal{O}_u) , and that of the buoy, (\mathcal{O}_b) , in \mathcal{W} , respectively. Let \mathcal{B}_u and \mathcal{B}_b be the body-fixed reference frames of the UAV at \mathcal{O}_u , and of the buoy at \mathcal{O}_b , respectively. The UAV is physically connected to the buoy by

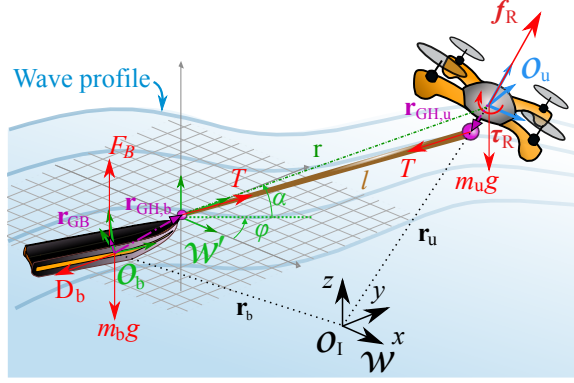


Figure 3.1: Three-dimensional model of a tethered quadrotor UAV–buoy system with tensile force interaction in a marine environment.

means of an inextensible massless cable of length $l \in \mathbb{R}_{>0}$, and the connections to the cable are modeled as two frictionless revolute joints. The quadrotor UAV has a mass m_u and a moment of inertia tensor $\mathbf{J}_u = \text{diag}(J_{u,xx}, J_{u,yy}, J_{u,zz}) \in \mathbb{R}_{>0}^3$ in \mathcal{B}_u . Let the floating buoy have a volume $\gamma_b \in \mathbb{R}_{>0}$, a bounded mass $m_b \in (0, \rho_w \gamma_b)$, with ρ_w being the water density, and a moment of inertia tensor $\mathbf{J}_b = \text{diag}(J_{b,xx}, J_{b,yy}, J_{b,zz}) \in \mathbb{R}_{>0}^3$ in \mathcal{B}_b . Also, let the orientation of \mathcal{B}_u and \mathcal{B}_b with respect to \mathcal{W} be described by the Euler angles $\mathbf{\Xi}_u = [\phi_u, \theta_u, \psi_u]^\top$ and $\mathbf{\Xi}_b = [\phi_b, \theta_b, \psi_b]^\top \in (-\pi, \pi]^3$, respectively. Let $\mathbf{V}_u = [u_u, v_u, w_u]^\top \in \mathbb{R}^3$ and $\mathbf{\Omega}_u = [p_u, q_u, r_u]^\top \in \mathbb{R}^3$ be the UAV's linear and angular velocities in \mathcal{B}_u , respectively; and let $\mathbf{V}_b = [u_b, v_b, w_b]^\top \in \mathbb{R}^3$ and $\mathbf{\Omega}_b = [p_b, q_b, r_b]^\top \in \mathbb{R}^3$ be the buoy's linear and angular velocities in \mathcal{B}_b , respectively. Furthermore, let the translational velocity transformation matrix from any body frame, \mathcal{B}_\bullet , to \mathcal{W} be described as:

$$\mathbf{R}_{1,\bullet} = \begin{bmatrix} c_{\theta_\bullet} c_{\psi_\bullet} - s_{\phi_\bullet} s_{\theta_\bullet} s_{\psi_\bullet} & -c_{\phi_\bullet} s_{\psi_\bullet} & s_{\theta_\bullet} c_{\psi_\bullet} + s_{\phi_\bullet} c_{\theta_\bullet} s_{\psi_\bullet} \\ c_{\theta_\bullet} s_{\psi_\bullet} + s_{\phi_\bullet} s_{\theta_\bullet} c_{\psi_\bullet} & c_{\phi_\bullet} c_{\psi_\bullet} & s_{\theta_\bullet} s_{\psi_\bullet} - s_{\phi_\bullet} c_{\theta_\bullet} c_{\psi_\bullet} \\ -c_{\phi_\bullet} s_{\theta_\bullet} & s_{\phi_\bullet} & c_{\phi_\bullet} c_{\theta_\bullet} \end{bmatrix}, \quad (3.1)$$

and the rotational velocity transformation matrix from \mathcal{W} to any body frame be described as:

$$\mathbf{R}_{2,\bullet} = \begin{bmatrix} c_{\theta_\bullet} & 0 & -c_{\phi_\bullet} s_{\theta_\bullet} \\ 0 & 1 & s_{\phi_\bullet} \\ s_{\theta_\bullet} & 0 & c_{\phi_\bullet} c_{\theta_\bullet} \end{bmatrix}. \quad (3.2)$$

We also define the rotation matrix about the vertical z -axis only (yaw) from one body frame, \mathcal{B}_\bullet , to \mathcal{W} as:

$$\mathbf{R}_{z_\bullet} = \begin{bmatrix} c_{\psi_\bullet} & -s_{\psi_\bullet} & 0 \\ s_{\psi_\bullet} & c_{\psi_\bullet} & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (3.3)$$

Both the UAV and the buoy are subject to cable tension, $T \in \mathbb{R}_{\geq 0}$, and to gravitational acceleration, g . The tensile force is applied on the tether hinge in each platform located at distances, $\mathbf{r}_{\text{GH,u}}$ and $\mathbf{r}_{\text{GH,b}}$, for the UAV and the buoy, measured in their individual body frames, respectively. Moreover, the UAV's propulsion is described by the input vector $\mathbf{u}_u = [\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3, \mathbf{u}_4]^\top \in \mathbb{R}^4$ including the thrust force vector $\mathbf{f}_u = [0, 0, \mathbf{u}_1]^\top \in \mathbb{R}_{\geq 0}^3$, and three torques that induce a rotational motion about each of the body frame axes such that $\boldsymbol{\tau}_u = [\mathbf{u}_2, \mathbf{u}_3, \mathbf{u}_4]^\top \in \mathbb{R}^3$. The dynamics of the UAV actuators (motor propellers) are neglected in the modeling given that their response time is considerably smaller than that of the UAV ($> 10\times$). Finally, the buoy is subjected to hydrodynamic and hydrostatic forces that are subsequently described.

3.2.3 Spherical Coordinates

The spherical coordinates system for the tethered UAV–buoy system, \mathcal{W}' , is illustrated in Fig. 3.2. The Cartesian position of the UAV in \mathcal{W} with respect to \mathcal{W}' is defined as: $\mathbf{r}_{\text{rel}} = [x_{\text{rel}}, y_{\text{rel}}, z_{\text{rel}}]^\top = \mathbf{r}_u - \mathbf{r}_b \in \mathbb{R}^3$, where the subscript (rel) denotes ‘relative.’ We let its spherical coordinates in \mathcal{W}' , $\mathbf{r} = [r, 0, 0]^\top$, be represented by the triplet $\{r, \alpha, \varphi\}$ with unit vectors $\{\hat{\mathbf{e}}_r, \hat{\mathbf{e}}_\alpha, \hat{\mathbf{e}}_\varphi\}$, such that the elevation angle $\alpha \in [-\pi/2, \pi/2]$ is defined between r and the horizontal plane, and the azimuth angle $\varphi \in (-\pi, \pi]$ is defined between the projection of r on the $\{x, y\}$ plane and the positive x -axis, which is mathematically represented as:

$$\begin{aligned} r &= \sqrt{x_{\text{rel}}^2 + y_{\text{rel}}^2 + z_{\text{rel}}^2}, \\ \alpha &= \text{atan}(z_{\text{rel}}/\sqrt{x_{\text{rel}}^2 + y_{\text{rel}}^2}), \\ \varphi &= \text{atan2}(y_{\text{rel}}, x_{\text{rel}}). \end{aligned} \quad (3.4)$$

To transform vectors from spherical to Cartesian coordinates, two consecutive rotations are needed such that:

$$\mathbf{R}_{\text{S2C}} = \mathbf{R}_\varphi \mathbf{R}_\alpha = \begin{bmatrix} c_\alpha c_\varphi & -s_\alpha c_\varphi & -s_\varphi \\ c_\alpha s_\varphi & -s_\alpha s_\varphi & c_\varphi \\ s_\alpha & c_\alpha & 0 \end{bmatrix}, \quad (3.5)$$

where \mathbf{R}_φ and \mathbf{R}_α are defined in C.1 and C.2, by which the Cartesian coordinates can be retrieved from the spherical coordinates such that:

$$\mathbf{r}_{\text{rel}} = \mathbf{R}_{\text{S2C}} \mathbf{r}, \quad (3.6)$$

which expands to:

$$\mathbf{r}_{\text{rel}} = \begin{bmatrix} r c_\alpha c_\varphi \\ r c_\alpha s_\varphi \\ r s_\alpha \end{bmatrix}. \quad (3.7)$$

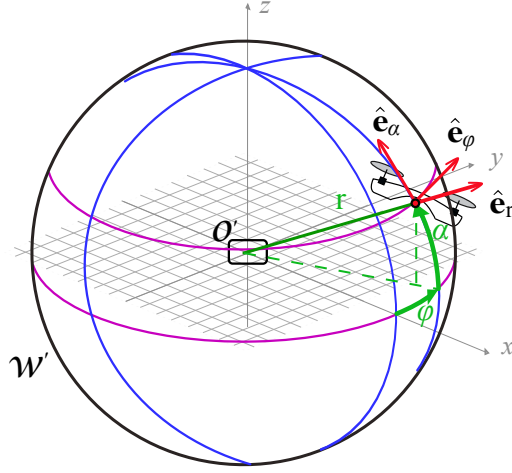


Figure 3.2: Unit vectors in spherical coordinates system: $\{r, \alpha, \varphi\}$. The system is left-handed, with $r \geq 0$, $\alpha \in (-\frac{\pi}{2}, \frac{\pi}{2}]$, and $\varphi \in (-\pi, \pi]$.

Conversely, the spherical coordinates can be retrieved from the Cartesian coordinates such that:

$$\mathbf{r} = \mathbf{R}_{\text{C2S}} \mathbf{r}_{\text{rel}}, \quad (3.8)$$

where $\mathbf{R}_{\text{C2S}} = \mathbf{R}_{\text{S2C}}^T$.

Hence, the UAV's inertial coordinates can be obtained from the buoy's inertial coordinates and the relative spherical coordinates as follows:

$$\mathbf{r}_u = \mathbf{r}_b + \mathbf{R}_{\text{S2C}} \mathbf{r}. \quad (3.9)$$

Similarly, the UAV's velocity and acceleration vectors are calculated via:

$$\dot{\mathbf{r}}_u = \dot{\mathbf{r}}_b + \mathbf{R}_{\text{S2C}} \dot{\mathbf{r}}, \quad (3.10a)$$

$$\ddot{\mathbf{r}}_u = \ddot{\mathbf{r}}_b + \mathbf{R}_{\text{S2C}} \ddot{\mathbf{r}}, \quad (3.10b)$$

where $\dot{\mathbf{r}}$ and $\ddot{\mathbf{r}}$ are explicitly given in C.4.

3.2.4 Water Medium Model

Consider the water medium of an ocean/sea environment, as visually depicted in Fig. 3.3, with two main elements that are detailed next: the water surface current and gravity waves.

Gravity Wave Model

Assuming a large water depth compared to the wavelength of gravity waves, at time t and horizontal position $\{x, y\}$, the water elevation variation, ζ , due to

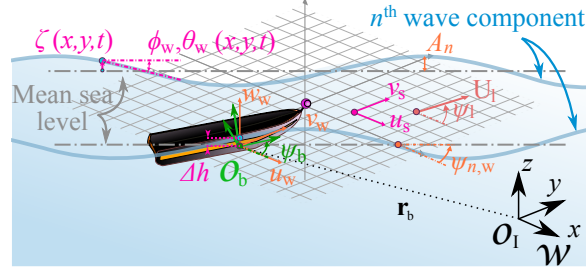


Figure 3.3: Water medium model elements and description, including: wave characteristics, wave velocity, floating object orientation, immersed height, Stokes-drift current, and lumped water surface current.

gravity waves is described statistically as [66]:

$$\zeta(x, y, t) = \sum_n^N A_n \sin(-k_n x \cos \psi_{n,w} - k_n y \sin \psi_{n,w} + \omega_n t + \sigma_n), \quad (3.11)$$

where A_n , k_n , $\omega_n \in \mathbb{R}_{\geq 0}$, $\psi_{n,w}$, and $\sigma_n \in (-\pi, \pi]$ are respectively the wave amplitude, wave number, circular frequency, wave propagation angle with respect to the inertial frame's x -axis, and random phase angle of wave component number $n \in S_n$ with $S_n = \{1 \leq n \leq N \mid N \in \mathbb{N}\}$, and N being the total number of active wave components. Furthermore, referring to the dispersion relation in deep water, the wave number is given as $k_n = \omega_n^2/g$. The wave-induced velocity vector of the fluid particles in \mathcal{W} , $\mathbf{U}_w = [u_w, v_w, w_w]^\top \in \mathbb{R}^3$, has its components described as [66]:

$$\begin{aligned} u_w(x, y, z, t) &= \sum_n^N \omega_n A_n e^{k_n z} \sin(-k_n x \cos \psi_{n,w} - k_n y \sin \psi_{n,w} + \omega_n t + \sigma_n) \cos \psi_{n,w}, \\ v_w(x, y, z, t) &= \sum_n^N \omega_n A_n e^{k_n z} \sin(-k_n x \cos \psi_{n,w} - k_n y \sin \psi_{n,w} + \omega_n t + \sigma_n) \sin \psi_{n,w}, \\ w_w(x, y, z, t) &= \sum_n^N \omega_n A_n e^{k_n z} \cos(-k_n x \cos \psi_{n,w} - k_n y \sin \psi_{n,w} + \omega_n t + \sigma_n), \end{aligned} \quad (3.12)$$

where ω_n relates to the wave period, T_n , via $\omega_n = 2\pi/T_n$.

Finally, the water surface plane at point $\{x, y, \zeta\}$ has tilt angles in roll and pitch with respect to the inertial frame, which are calculated by differentiating (3.11) with respect to x and y , respectively:

$$\begin{aligned} \phi_w(x, y, t) &= \text{atan}\left(\sum_n^N -A_n k_n \sin \psi_{n,w} \cos(\omega_n t - k_n x \cos \psi_{n,w} - k_n y \sin \psi_{n,w} + \sigma_n)\right), \\ \theta_w(x, y, t) &= \text{atan}\left(\sum_n^N -A_n k_n \cos \psi_{n,w} \cos(\omega_n t - k_n x \cos \psi_{n,w} - k_n y \sin \psi_{n,w} + \sigma_n)\right). \end{aligned} \quad (3.13)$$

Water Current

The water surface current acts in the $\{x, y\}$ (horizontal) plane of \mathcal{W} , such that $\mathbf{U}_c = [u_c, v_c, 0]^\top \in \mathbb{R}^3$, and is given as:

$$\mathbf{U}_c = \mathbf{U}_s + \mathbf{U}_1, \quad (3.14)$$

where $\mathbf{U}_s = [u_s, v_s, 0]^\top \in \mathbb{R}^3$ is generated from Stokes drift [67], with its components in the inertial frame, \mathcal{W} , defined as:

$$\begin{aligned} u_s(z) &= \sum_n^N \cos \psi_{n,w} A_n^2 \omega_n k_n e^{2k_n z}, \\ v_s(z) &= \sum_n^N \sin \psi_{n,w} A_n^2 \omega_n k_n e^{2k_n z}, \end{aligned} \quad (3.15)$$

and $\mathbf{U}_1 = [u_1, v_1, 0]^\top \in \mathbb{R}^3$ is the resulting sum of other water current components, determined as follows:

$$\begin{aligned} u_1 &= U_1 \cos \psi_1, \\ v_1 &= U_1 \sin \psi_1, \end{aligned} \quad (3.16)$$

where U_1 and ψ_1 are the average velocity and direction of the current in \mathcal{W} , respectively.

3.2.5 Buoy's Dynamic Model

Various types of forces affect the buoy's motion, with a major contribution from restoration, damping, and radiation forces. These forces substantially depend on the immersed volume of the buoy, $\Upsilon_{\text{im}} \in [0, \Upsilon_b]$, which is a function of the buoy's elevation, defined as $\Delta h = \zeta(x_b, y_b, t) - z_b$. Let the vector from the buoy's center of gravity to its center of buoyancy in \mathcal{W} be defined as $\mathbf{r}_{\text{GB}} = [x_{\text{GB}}, y_{\text{GB}}, z_{\text{GB}}]^\top \in \mathbb{R}^3$, where the buoyancy force is applied such that $\mathbf{F}_B = [0; 0; \rho_w g \Upsilon_{\text{im}}]$. Additionally, let the tether hinge location with respect to the buoy's center of gravity be defined in \mathcal{B}_b as $\mathbf{r}_{\text{GH},b} = [x_{\text{GH},b}, y_{\text{GH},b}, z_{\text{GH},b}]^\top \in \mathbb{R}^3$, as seen in Fig. 3.1, where the cable tension is applied to the buoy in \mathcal{W}' , such that $\mathbf{T}'_b = [T, 0, 0]^\top$.

Assumption 10. *The water-buoy friction dominates the air-buoy friction, thus we neglect the air drag on the buoy.*

Consider the buoy dynamics in \mathcal{B}_b , and let its state vector be $\boldsymbol{\nu}_b = [\boldsymbol{\nu}_{1,b}; \boldsymbol{\nu}_{2,b}]$, where $\boldsymbol{\nu}_{1,b} = \mathbf{V}_b$ and $\boldsymbol{\nu}_{2,b} = \boldsymbol{\omega}_b$. Applying Newton's second law of motion gives:

$$\mathbf{M}'_b \dot{\boldsymbol{\nu}}_b + \mathbf{C}'_b \boldsymbol{\nu}_b + \mathbf{D}'_b \tilde{\boldsymbol{\nu}}_b + \mathbf{G}'_b = \boldsymbol{\tau}'_b, \quad (3.17)$$

where \mathbf{M}'_b , \mathbf{D}'_b , and $\mathbf{C}'_b \in \mathbb{R}^{6 \times 6}$ are respectively the buoy's inertia, damping, and Coriolis matrices expressed in \mathcal{B}_b ; $\tilde{\boldsymbol{\nu}}_b = [\tilde{\boldsymbol{\nu}}_{1,b}; \tilde{\boldsymbol{\nu}}_{2,b}]$ is the relative velocity vector, with $\tilde{\boldsymbol{\nu}}_{1,b} = \boldsymbol{\nu}_{1,b} - (\mathbf{U}_w + \mathbf{R}_{\psi_w}^\top \mathbf{U}_s + \mathbf{R}_{\psi_1}^\top \mathbf{U}_1)$, and $\tilde{\boldsymbol{\nu}}_{2,b} = \boldsymbol{\nu}_{2,b}$; the gravitational forces and moments vector are included in $\mathbf{G}'_b \in \mathbb{R}^6$; and the external forces and moments are captured in $\boldsymbol{\tau}'_b \in \mathbb{R}^6$. The inertia matrix expands as $\mathbf{M}'_b = [\mathbf{M}'_{1,b}, \mathbf{O}_3; \mathbf{O}_3, \mathbf{M}'_{2,b}]$, where $\mathbf{O}_3 \in \mathbb{R}^{3 \times 3}$ is the null matrix, $\mathbf{M}'_{1,b} = m_b \text{diag}(1, 1, 1) + \mathbf{a}_{1,b}$, $\mathbf{M}'_{2,b} = \mathbf{J}_b + \mathbf{a}_{2,b}$, with $\mathbf{a}_{1,b} = \text{diag}(a_{11}, a_{22}, a_{33})$ and $\mathbf{a}_{2,b} = \text{diag}(a_{44}, a_{55}, a_{66}) \in \mathbb{R}^{3 \times 3}$ being the generalized added mass matrices. The Coriolis matrix, which depends on \mathbf{M}'_b , is calculated as $\mathbf{C}'_b = [\mathbf{O}_3, \mathbf{C}'_{12,b}; \mathbf{C}'_{21,b}, \mathbf{C}'_{2,b}]$, where:

$$\mathbf{C}'_{12,b} = \mathbf{C}'_{21,b} = \begin{bmatrix} 0 & (m_b + a_{33})w_b & -(m_b + a_{22})v_b \\ -(m_b + a_{33})w_b & 0 & (m_b + a_{11})u_b \\ (m_b + a_{22})v_b & -(m_b + a_{11})u_b & 0 \end{bmatrix} \quad (3.18)$$

$$\mathbf{C}'_{2,b} = \begin{bmatrix} 0 & (J_{zz,b})r_b & -(J_{yy,b})q_b \\ -(J_{zz,b})r_b & 0 & (J_{xx,b})p_b \\ (J_{yy,b})q_b & -(J_{xx,b})p_b & 0 \end{bmatrix}.$$

The buoy's total damping in \mathcal{B}_b is expressed as:

$$\mathbf{D}'_b = \mathbf{D}_P + \mathbf{D}_S + \mathbf{D}_W, \quad (3.19)$$

where the generalized radiation-induced potential damping matrix is expanded as $\mathbf{D}_P = [\mathbf{b}_{1,b}, \mathbf{O}_3; \mathbf{O}_3, \mathbf{b}_{1,b}] \in \mathbb{R}^{6 \times 6}$, with $\mathbf{b}_{1,b} = \text{diag}(b_{11}, b_{22}, b_{33})$ and $\mathbf{b}_{2,b} = \text{diag}(b_{44}, b_{55}, b_{66})$, and $\mathbf{D}_S = \text{diag}(D_{S,1}, \dots, D_{S,6}) \in \mathbb{R}^{6 \times 6}$ is the skin friction matrix, calculated as:

$$D_{S,i} = C_{S,i} A_{\text{wt}} \frac{1}{2} \rho_w |\tilde{\boldsymbol{\nu}}_{1,b,i}|, \quad i = \{1, 2, 3\}, \quad (3.20)$$

where $C_{S,i} \in \mathbb{R}_{>0}$ is the drag coefficient, $A_{\text{wt}} \in \mathbb{R}_{\geq 0}$ is the wetted area of the buoy, and $D_{S,4-6} \in \mathbb{R}_{\geq 0}$ can be approximated by considering the moments effect of $D_{S,1-3}$ over the buoy's surface. The effect of the wave drift damping matrix, $\mathbf{D}_W \in \mathbb{R}^{3 \times 3}$, is already included in the Stokes drift velocity in (3.15), thus it will be dropped from (3.19).

The buoy dynamics in (3.17) are expressed in \mathcal{W} with the state vector, $\boldsymbol{\eta}_b = [\boldsymbol{\eta}_{1,b}; \boldsymbol{\eta}_{2,b}]$ where $\boldsymbol{\eta}_{1,b} = \mathbf{r}_b$ and $\boldsymbol{\eta}_{2,b} = \tilde{\boldsymbol{\Xi}}_b$, as:

$$\mathbf{M}_b \ddot{\boldsymbol{\eta}}_b + \mathbf{C}_b \dot{\boldsymbol{\eta}}_b + \mathbf{D}_b \tilde{\boldsymbol{\eta}}_b + \mathbf{G}_b = \boldsymbol{\tau}_b, \quad (3.21)$$

where \mathbf{M}_b , \mathbf{D}_b , and $\mathbf{C}_b \in \mathbb{R}^{6 \times 6}$ are the buoy's inertia, damping, and Coriolis matrices, respectively, expressed in \mathcal{W} ; $\tilde{\boldsymbol{\eta}}_b = [\tilde{\boldsymbol{\eta}}_{1,b}; \tilde{\boldsymbol{\eta}}_{2,b}]$ is the relative velocity vector, with $\tilde{\boldsymbol{\eta}}_{1,b} := [\tilde{x}_b, \tilde{y}_b, \tilde{z}_b]^\top = \dot{\boldsymbol{\eta}}_{1,b} - (\mathbf{U}_w + \mathbf{U}_c)$; \mathbf{G}_b and $\boldsymbol{\tau}_b$ are respectively the vectors of the gravitational and external forces and moments on the buoy in

\mathcal{W} given by:

$$\begin{aligned}
\mathbf{G}_b &= [0, 0, m_b g, 0, 0, 0]^\top, \\
\boldsymbol{\tau}_b &= [\boldsymbol{\tau}_{1,b}; \boldsymbol{\tau}_{2,b}], \\
\boldsymbol{\tau}_{1,b} &= \mathbf{R}_{S2C} \mathbf{T}'_b + \mathbf{F}_B \\
&= [T c_\alpha c_\phi, T c_\alpha s_\phi, T s_\alpha + \rho_w g \Upsilon_{im}]^\top, \\
\boldsymbol{\tau}_{2,b} &= \mathbf{r}_{GB} \times \mathbf{F}_B + (\mathbf{R}_{1,b} \mathbf{r}_{GH,b}) \times (\mathbf{R}_{S2C} \mathbf{T}'_b).
\end{aligned} \tag{3.22}$$

We also define:

$$\begin{aligned}
\mathbf{M}_b &= \mathbf{R}_b \mathbf{M}'_b \mathbf{R}_b^{-1}, \\
\mathbf{D}_b &= \mathbf{R}_b \mathbf{D}'_b \mathbf{R}_b^{-1}, \\
\mathbf{C}_b \dot{\boldsymbol{\eta}}_b &:= \frac{1}{2} \dot{\mathbf{M}}_b \dot{\boldsymbol{\eta}}_b,
\end{aligned} \tag{3.23}$$

where $\mathbf{R}_b = [\mathbf{R}_{1,b}, \mathbf{O}_3; \mathbf{O}_3, \mathbf{R}_{2,b}^{-1}]$, $\dot{\mathbf{M}}_b = \dot{\boldsymbol{\eta}}_b^\top (\partial \mathbf{M}_b / \partial \boldsymbol{\eta}_b)$ [67]. Finally, we let $M_{b,ij}$, $C_{b,ij}$, and $D_{b,ij}$, $i, j = \{1, \dots, 6\}$ be elements of \mathbf{M}_b , \mathbf{C}_b , and \mathbf{D}_b , respectively.

3.2.6 UAV's Dynamic Model

We let the UAV's thrust vector in the Cartesian frame, $\mathbf{f}_{u,C} = [u_x, u_y, u_z]^\top$, be calculated as:

$$\mathbf{f}_{u,C} = \mathbf{R}_{1,u} \mathbf{f}_u, \tag{3.24}$$

with its elements being explicitly represented as:

$$\begin{aligned}
u_x &= u_1 (s_{\theta_u} c_{\psi_u} + s_{\phi_u} c_{\theta_u} s_{\psi_u}), \\
u_y &= u_1 (s_{\theta_u} s_{\psi_u} - s_{\phi_u} c_{\theta_u} c_{\psi_u}), \\
u_z &= u_1 (c_{\phi_u} c_{\theta_u}).
\end{aligned} \tag{3.25}$$

In the spherical frame, the UAV's thrust vector, $\mathbf{f}_{u,S} = [u_r, u_\alpha, u_\varphi]^\top$, is expressed as:

$$\mathbf{f}_{u,S} = \mathbf{R}_{C2S} \mathbf{f}_{u,C}, \tag{3.26}$$

with the element u_r , u_α , and u_φ being explicitly represented in C.5.

The tether's tension on the UAV expressed in \mathcal{W}' , $\mathbf{T}'_u = [-T, 0, 0]^\top$, is applied at location $\mathbf{r}_{GH,u} = [x_{GH,u}, y_{GH,u}, z_{GH,u}]^\top \in \mathbb{R}^3$ in \mathcal{B}_u , representing the distance from the UAV's center of gravity to its tether hinge. Finally, the local wind speed that disturbs the UAV's motion is defined in \mathcal{W} as $\mathbf{U}_{wd} = [u_{wd}, v_{wd}, 0]^\top$.

The quadrotor UAV system dynamics in \mathcal{W} are obtained from Newton's second law of motion with the state vector $\boldsymbol{\eta}_u = [\boldsymbol{\eta}_{1,u}; \boldsymbol{\eta}_{2,u}]$, where $\boldsymbol{\eta}_{1,u} = \mathbf{r}_u$ and $\boldsymbol{\eta}_{2,u} = \boldsymbol{\Xi}_u$, yields:

$$\begin{aligned}
\mathbf{M}_{1,u} \ddot{\boldsymbol{\eta}}_{1,u} + \mathbf{C}_{1,u} \dot{\boldsymbol{\eta}}_{1,u} + \mathbf{D}_{1,u} \tilde{\boldsymbol{\eta}}_{1,u} + \mathbf{G}_{1,u} &= \boldsymbol{\tau}_{1,u} + \boldsymbol{\delta}_{1,u}, \\
\mathbf{M}_{2,u} \ddot{\boldsymbol{\eta}}_{2,u} + \mathbf{C}_{2,u} \dot{\boldsymbol{\eta}}_{2,u} + \mathbf{D}_{2,u} \tilde{\boldsymbol{\eta}}_{2,u} + \mathbf{G}_{2,u} &= \boldsymbol{\tau}_{2,u} + \boldsymbol{\delta}_{2,u},
\end{aligned} \tag{3.27}$$

where:

$$\begin{aligned}\mathbf{M}_{1,u} &= m_u \text{diag}(1, 1, 1), \quad \mathbf{D}_{1,u} = \mathbf{R}_{1,u} \mathbf{D}'_{1,u} \mathbf{R}_{1,u}^\top, \\ \mathbf{C}_{1,u} \boldsymbol{\eta}_{1,u} &:= \mathbf{O}_3, \quad \mathbf{G}_{1,u} = [0, 0, m_u g]^\top,\end{aligned}$$

and

$$\begin{aligned}\mathbf{M}_{2,u} &= \mathbf{J}_u \mathbf{R}_{2,u}, \quad \mathbf{D}_{2,u} = \mathbf{D}'_{2,u} \mathbf{R}_{2,u}, \quad \mathbf{G}_{2,u} = [0, 0, 0]^\top, \\ \mathbf{C}_{2,u} \dot{\boldsymbol{\eta}}_u &:= \boldsymbol{\Omega}_u \times (\mathbf{J}_u \boldsymbol{\Omega}_u) + \mathbf{J}_u \left(\frac{\partial \mathbf{R}_{2,u}}{\partial \phi_u} \dot{\phi}_u + \frac{\partial \mathbf{R}_{2,u}}{\partial \theta_u} \dot{\theta}_u \right) \dot{\boldsymbol{\eta}}_{2,u}.\end{aligned}$$

$\mathbf{D}'_{1,u} = \text{diag}(D_{u,1}, D_{u,2}, D_{u,3})$ and $\mathbf{D}'_{2,u} = \text{diag}(D_{u,4}, D_{u,5}, D_{u,6}) \in \mathbb{R}_{\geq 0}^{3 \times 3}$ are the UAV's translational and rotational damping friction matrices, respectively. The relative velocity vectors of the UAV are $\tilde{\boldsymbol{\eta}}_{1,u} := [\tilde{x}_u, \tilde{y}_u, \tilde{z}_u]^\top = \dot{\boldsymbol{\eta}}_{1,u} - \mathbf{U}_{\text{wd}}$ in translation and $\tilde{\boldsymbol{\eta}}_{2,u} := [\tilde{\phi}_u, \tilde{\theta}_u, \tilde{\psi}_u]^\top = \dot{\boldsymbol{\eta}}_{2,u}$ in rotation. $\boldsymbol{\tau}_{1,u}$ and $\boldsymbol{\tau}_{2,u} \in \mathbb{R}^3$ are vectors of other external forces in \mathcal{W} and moments in \mathcal{B}_u of the UAV, respectively, and they include the rotors' thrust and tether tension effects, expressed as:

$$\begin{aligned}\boldsymbol{\tau}_{1,u} &= \mathbf{f}_{u,C} + \mathbf{R}_{S2C} \mathbf{T}'_u \\ &= \begin{bmatrix} \mathbf{u}_x - T c_\alpha c_\varphi \\ \mathbf{u}_y - T c_\alpha s_\varphi \\ \mathbf{u}_z - T s_\alpha \end{bmatrix}, \\ \boldsymbol{\tau}_{2,u} &= \boldsymbol{\tau}_u + (\mathbf{r}_{\text{GH},u}) \times (\mathbf{R}_{1,u}^\top \mathbf{R}_{S2C} \mathbf{T}'_u).\end{aligned}\tag{3.28}$$

The damping matrix elements, $D_{u,i}$, $i \in \{1, \dots, 3\}$, are approximated as:

$$D_{u,i} = C_{u,i} A_{\text{cs},i}^u \frac{1}{2} \rho_a |\tilde{v}_{u,i}|,\tag{3.29}$$

where $C_{u,i} \in \mathbb{R}_{>0}$ is a drag coefficient, $A_{\text{cs},i}^u \in \mathbb{R}_{\geq 0}$ is the UAV's cross-sectional area in the respective plane, ρ_a is the air density and $\tilde{v}_{u,i}$ is the UAV-wind relative velocity in the respective body frame axis. The adopted model captures the major elements required to represent the quadrotor in a tethered UAV–buoy system with slow-to-moderate dynamics, thus acrobatic maneuvers and their influence on the system dynamics are not considered. For more details on the quadrotor UAV model, readers can refer to [68].

3.2.7 System Constraints

In this section, we generalise the system's 2D constraints presented in [31] to the 3D space. These constraints help establish the bounds and operating conditions for when the coupled model of the tethered UAV–buoy system is applicable.

Taut-Cable Constraint

Since the cable is assumed inextensible, it remains under tension (taut) at time t if $r(t) = l$, which is equivalent to:

$$T > 0. \quad (3.30)$$

We label the tethered UAV–buoy system as ‘*coupled*’ when (3.30) is satisfied, and ‘*decoupled*’ otherwise.

No Buoy-Hanging Constraint

To keep the buoy floating on top of the water surface, the vertical tension component transmitted from the UAV through the tether must not exceed the weight of the buoy, otherwise it will be lifted into the air and the system will reduce to a UAV with a slung payload. This is prescribed by:

$$T < m_b g / s_\alpha, \quad (3.31)$$

as deduced from (3.21) and (3.22) at steady-state.

No ‘Fly-Over’ Constraint

‘Fly-over’ occurs when the buoy starts hopping over wave crests [69]. This phenomenon is avoided if:

$$\Upsilon_{\text{im}} > 0, \quad (3.32)$$

which means that the buoy remains partially immersed at all times. The ‘fly-over’ phenomenon is related to the total surface velocity of the buoy, described as:

$$\begin{aligned} V &= \sqrt{u_b^2 + v_b^2}, \\ \psi_V &= \arctan(v_b/u_b), \end{aligned} \quad (3.33)$$

where V and ψ_V are its magnitude and direction in the horizontal plane, respectively. Next, we define the wave encounter frequency for the n^{th} wave component, $\omega_{e,n}$, as [66]:

$$\omega_{e,n} = \omega_n - \frac{\omega_n^2 V}{g} c_{\psi_w, n - \psi_V}, \quad n \in S_n. \quad (3.34)$$

The excitation of the buoy’s heave dynamics at $\omega_{e,n}$ induces ‘fly-over’ if it approaches the heave’s natural frequency, as interpreted in [31].

3.2.8 The Tethered UAV–Buoy System Model

In a coupled form, the tethered UAV–buoy system dynamics can be formulated by referring to the Euler-Lagrange formulation, which leverages the results in Sections 3.2.3, 3.2.5, 3.2.6, and 3.2.7. First, we define the Lagrangian function as $\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}) = \mathcal{K}(\mathbf{q}, \dot{\mathbf{q}}) - \mathcal{U}(\mathbf{q})$, where $\mathcal{K}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}_{\geq 0}$ and $\mathcal{U}(\mathbf{q}) \in \mathbb{R}$ are the kinetic and potential energies of the system, with $\mathbf{q} = [x_b, y_b, z_b, \alpha, \varphi, \phi_u, \theta_u, \psi_u, \phi_b, \theta_b, \psi_b]^\top \in \mathbb{R}^{11}$ being the generalized coordinates vector. The equations of motion of the UAV–buoy system are obtained via:

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \right) - \frac{\partial \mathcal{L}}{\partial \mathbf{q}} + \frac{\partial \mathcal{P}}{\partial \dot{\mathbf{q}}} = \boldsymbol{\tau}, \quad (3.35)$$

where $\boldsymbol{\tau} \in \mathbb{R}^{11}$ is the external forces and moments vector. Per Assumption 10, let \mathbf{D} be the global damping matrix formulated based on (3.23) without including a wind-induced component, and the dissipative forces are captured by the power function, $\mathcal{P} \in \mathbb{R}$, such that $\frac{\partial \mathcal{P}}{\partial \dot{\mathbf{q}}} := \mathbf{D}\tilde{\dot{\mathbf{q}}}$. $\tilde{\dot{\mathbf{q}}}$ is defined as:

$$\begin{aligned} \tilde{\dot{\mathbf{q}}} &= \dot{\mathbf{q}} - [u_c + u_w, v_c + v_w, w_w, 0, 0, 0, 0, 0, 0, 0, 0]^\top \\ &= [\dot{x}_b - u_c - u_w, \dot{y}_b - v_c - v_w, \dot{z}_b - w_w, \dot{\alpha}, \dot{\varphi}, \dot{\phi}_u, \dot{\theta}_u, \dot{\psi}_u, \dot{\phi}_b, \dot{\theta}_b, \dot{\psi}_b]^\top. \end{aligned} \quad (3.36)$$

If tether is assumed to have a negligible mass, the coupled system's kinetic energy is obtained as the sum of the individual energies of the UAV and the buoy as follows:

$$\mathcal{K} = \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{M} \dot{\mathbf{q}} := \frac{1}{2} \dot{\boldsymbol{\eta}}_u^\top \mathbf{M}_u \dot{\boldsymbol{\eta}}_u + \frac{1}{2} \dot{\boldsymbol{\eta}}_b^\top \mathbf{M}_b \dot{\boldsymbol{\eta}}_b, \quad (3.37)$$

where the global inertia matrix of the UAV–buoy system, \mathbf{M} , is formulated by using the elements of \mathbf{M}_u and \mathbf{M}_b , as described in (C.8). Next, the potential energy of the system can be formulated by referring to (3.22) and (3.2.6) as:

$$\mathcal{U} = m_b g z_b + m_u g (z_b + l s_\alpha). \quad (3.38)$$

The details of the Euler-Lagrange formulation in (3.35) are detailed in Appendix D, which finally leads to the following equations of motion:

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{D}\tilde{\dot{\mathbf{q}}} + \mathbf{G} = \boldsymbol{\tau}, \quad (3.39)$$

where \mathbf{C} represents the global Coriolis matrix, and \mathbf{G} is the global vector of gravity forces and moments.

Assumption 11. *The design of a stable buoy is beyond the scope of this work, and only buoys with inherited stability are considered. Thus, we assume that the buoy's center of buoyancy always lies above its center of gravity, and that the roll and pitch dynamics of the buoy are damped and stable, which indicates that $D_{b,44} > 0$ and $D_{b,55} > 0$. Thus, we assume that the buoy remains tangent to the water surface.*

With Assumption 11 and the dominance of waves with relatively long periods and moderate heights, the time derivatives of the buoy's roll and pitch angles, $\dot{\phi}_b$ and $\dot{\theta}_b$, are small and thus their effects can be neglected in $\dot{\mathbf{M}}$.

Since the tether joints are considered revolute and frictionless, there is no coupling between the tether's rotational motion and the UAV and buoy's rotational dynamics, whereas their translational dynamics are coupled through (3.9). Additionally, as can be deduced by inspecting the elements of (3.35), the UAV and buoy's rotational dynamics are independent of each other, and can still be described by the dynamic models of their individual systems in (3.27) and (3.21), respectively. This is further elaborated in Appendix D. Given this, the tethered UAV–buoy system coupled dynamics can be represented by the first five states: x_b , y_b , z_b , α , and φ , which are independent of other rigid body orientation states that concern either the UAV's or the buoy's dynamics alone. Hereafter, we limit the representation of the coupled system to the first five state variables mentioned above, while the UAV and buoy's rotational dynamics can still be described by $\boldsymbol{\eta}_{2,u}$ in (3.27) and $\boldsymbol{\eta}_{2,b}$ in (3.21), respectively. The inertia matrix for the coupled system is explicitly represented as $\mathbf{M}_{1-5} = [M_1; M_2; M_3; M_4; M_5]$, where:

$$\begin{aligned} M_1 &= [M_{b,11} + m_u \quad M_{b,12} \quad M_{b,13} \quad -m_u l s_\alpha c_\varphi \quad -m_u l c_\alpha s_\varphi], \\ M_2 &= [M_{b,21} \quad M_{b,22} + m_u \quad M_{b,23} \quad -m_u l s_\alpha s_\varphi \quad m_u l c_\alpha c_\varphi], \\ M_3 &= \begin{bmatrix} M_{b,31} & M_{b,32} & M_{b,33} + m_u & m_u l c_\alpha & 0 \end{bmatrix}, \\ M_4 &= \begin{bmatrix} -m_u l s_\alpha c_\varphi & -m_u l s_\alpha s_\varphi & m_u l c_\alpha & m_u l^2 & 0 \end{bmatrix}, \\ M_5 &= \begin{bmatrix} -m_u l c_\alpha s_\varphi & m_u l c_\alpha c_\varphi & 0 & 0 & m_u l^2 c_\alpha^2 \end{bmatrix}. \end{aligned} \quad (3.40)$$

The Coriolis matrix is explicitly represented as:

$$\mathbf{C}_{1-5} = m_u l \begin{bmatrix} 0 & 0 & 0 & -c_\alpha c_\varphi \dot{\alpha} + 2s_\alpha s_\varphi \dot{\varphi} & -c_\alpha c_\varphi \dot{\varphi} \\ 0 & 0 & 0 & -c_\alpha s_\varphi \dot{\alpha} - 2s_\alpha c_\varphi \dot{\varphi} & -c_\alpha s_\varphi \dot{\varphi} \\ 0 & 0 & 0 & -s_\alpha \dot{\alpha} & 0 \\ 0 & 0 & 0 & 0 & l s_\alpha c_\alpha \dot{\varphi} \\ 0 & 0 & 0 & -2l s_\alpha c_\alpha \dot{\varphi} & 0 \end{bmatrix}, \quad (3.41)$$

and the damping matrix is explicitly represented as:

$$\mathbf{D}_{1-5} = \begin{bmatrix} D_{b,11} & D_{b,12} & D_{b,13} & 0 & 0 \\ D_{b,21} & D_{b,22} & D_{b,23} & 0 & 0 \\ D_{b,31} & D_{b,32} & D_{b,33} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \quad (3.42)$$

Additionally, the gravitation force vector and external forces and torques vector are explicitly represented as:

$$\mathbf{G}_{1-5} = [0, 0, (m_b + m_u)g, m_u g l c_\alpha, 0]^T, \quad (3.43)$$

and

$$\boldsymbol{\tau}_{1-5} = [\mathbf{u}_x, \mathbf{u}_y, \mathbf{u}_z, l\mathbf{u}_\alpha, l\mathbf{u}_\varphi]^\top, \quad (3.44)$$

respectively. If constraints (3.30) and (3.32) are satisfied, the coupled form of the dynamic model equations is given by:

$$(M_{b,11} + m_u)\ddot{x}_b + M_{b,12}\ddot{y}_b + M_{b,13}\ddot{z}_b + D_{b,11}\dot{\tilde{x}}_b + D_{b,12}\dot{\tilde{x}}_b + D_{b,13}\dot{\tilde{z}}_b - m_ul(c_\alpha c_\varphi \dot{\alpha}^2 - 2s_\alpha s_\varphi \dot{\alpha}\dot{\varphi} + c_\alpha c_\varphi \dot{\varphi}^2 + s_\alpha c_\varphi \ddot{\alpha} + c_\alpha s_\varphi \ddot{\varphi}) = \mathbf{u}_x, \quad (3.45a)$$

$$M_{b,21}\ddot{x}_b + (M_{b,22} + m_u)\ddot{y}_b + M_{b,23}\ddot{z}_b + D_{b,21}\dot{\tilde{x}}_b + D_{b,22}\dot{\tilde{y}}_b + D_{b,23}\dot{\tilde{z}}_b - m_ul(c_\alpha s_\varphi \dot{\alpha}^2 + 2s_\alpha c_\varphi \dot{\alpha}\dot{\varphi} + c_\alpha s_\varphi \dot{\varphi}^2 + s_\alpha s_\varphi \ddot{\alpha} - c_\alpha c_\varphi \ddot{\varphi}) = \mathbf{u}_y, \quad (3.45b)$$

$$M_{b,31}\ddot{x}_b + M_{b,32}\ddot{y}_b + (M_{b,33} + m_u)\ddot{z}_b + D_{b,31}\dot{\tilde{x}}_b + D_{b,32}\dot{\tilde{y}}_b + D_{b,33}\dot{\tilde{z}}_b - m_ul(s_\alpha \dot{\alpha}^2 - c_\alpha \ddot{\alpha}) + (m_u + m_b - \rho_w \Upsilon_{im})\mathbf{g} = \mathbf{u}_z, \quad (3.45c)$$

$$m_ul(-s_\alpha c_\varphi \ddot{x}_b - s_\alpha s_\varphi \ddot{y}_b + c_\alpha \ddot{z}_b + ls_\alpha c_\alpha \dot{\varphi}^2) + m_ul^2 \ddot{\alpha} + m_ul \mathbf{g}(lc_\alpha) = l\mathbf{u}_\alpha, \quad (3.45d)$$

$$m_ul(-c_\alpha s_\varphi \ddot{x}_b + c_\alpha c_\varphi \ddot{y}_b - ls_\alpha c_\alpha \dot{\alpha}\dot{\varphi}) + m_ul^2 c_\alpha^2 \ddot{\varphi} = l\mathbf{u}_\varphi. \quad (3.45e)$$

After solving the above differential equations, the UAV's position and velocity vectors can then be computed from (C.6) and (C.8), respectively. We note that if the tether's weight is to be considered, the tethered system's model should be updated as per [85].

3.3 Control System Design

The control system design problem is defined as manipulating the surge velocity of the buoy, u_b , to track a desired reference, while orienting the cable in the desired elevation angle, α , and azimuth angle, φ . This allows for applying a tension force in any required direction, while tracking the desired surge velocity. As an extension to the Surge Velocity Control System (SVCS) presented in [31], which was designed to only control u_b and α , the controller will be dubbed herein as Directional Surge Velocity Control System (DSVCS), given that it adds the pulling force direction (azimuth angle, φ) to its controlled states.

We note that solely controlling these states via setpoint tracking does not yield inertial velocity tracking. For example, external forces that result in sway motion (in the direction of the buoy's body-fixed y -axis) are not rejected by the controller. That said, the controller with the reduced states can be equipped with a path planner, which can apply lateral tension by choosing a specific azimuth angle to counter the external forces in the sway direction. Such an architecture would give the system the ability to track reference trajectories in the inertial frame, but this is beyond the scope of this work, thus we stick with the design of the DSVCS.

3.3.1 Operational Modes and State Machine

To complement the surge velocity controller, a UAV–buoy relative position controller is required to control the radial distance of the UAV in \mathcal{W}' , r , instead of u_b . Both controllers are part of the DSVCS. In [92], a state machine was proposed to allow the system to switch between the coupled and decoupled states by alternating between the two controllers. In addition, the state machine can trigger a change in the positioning of the UAV with respect to the buoy between front and rear to change the pulling direction. Contrarily, in the 3D problem of this work, we only consider the forward motion of the buoy, while the velocity vector steering will be handled by changing the cable’s azimuth angle. The purpose of the state machine is to allow the system to smoothly couple and decouple when requested by the controller, which will be sufficient to achieve the goals of the controller in this work. For that, we adopt the same state-machine of [92], with the exclusion of the ‘*repositioning*’ maneuver trigger part.

3.3.2 Controller Design

With the inner-loop/outer-loop cascaded structure shown in Fig. 3.4, the DSVCS allows tracking of the states u_b , α , and φ by orienting and scaling the UAV’s thrust vector. The setpoint for the controller is $(\bar{u}_{b0}, \bar{z}_{u0}, \bar{\varphi}_0)$ in the buoy’s surge velocity control mode, and $(\bar{r}_0, \bar{z}_{u0}, \bar{\varphi}_0)$ in the UAV’s relative position mode. A preprocessing unit then generates the resulting $\bar{\alpha}$ based on \bar{z}_{u0} , and a smoothed signal for the tracking setpoints \bar{u}_{b0} , \bar{r}_0 , and $\bar{\varphi}_0$. When the control mode changes, a fast and smooth transition function handles the switching between the two control laws [31].

The resulting force vector with its three components along \hat{e}_r , \hat{e}_α , and \hat{e}_φ is transformed to the inertial frame to determine the desired UAV’s thrust vector magnitude and orientation. At this point, the problem reverts to a basic UAV thrust decoupling problem, where the resulting thrust vector magnitude and tilt angles can be easily computed, as shown later, while the yaw angle remains free to set. Since the UAV is expected to maintain visual tracking of the buoy, the controller sets the desired yaw angle to equal the tether’s azimuth angle.

In summary, and as shown in Fig. 3.5, the proposed DSVCS can control three main variables: 1) the relative position between the UAV and the buoy (outer-loop), 2) the buoy’s surge velocity (outer-loop), and 3) the UAV’s attitude (inner-loop). A state machine selects which of the two outer-loop controllers to activate based on the coupling between the UAV and buoy.

Reference Signals and Velocity Setpoint

The UAV’s desired motion during a buoy manipulation task is limited to a horizontal plane of constant elevation (\bar{z}_{u0}), which reduces the UAV’s power consump-

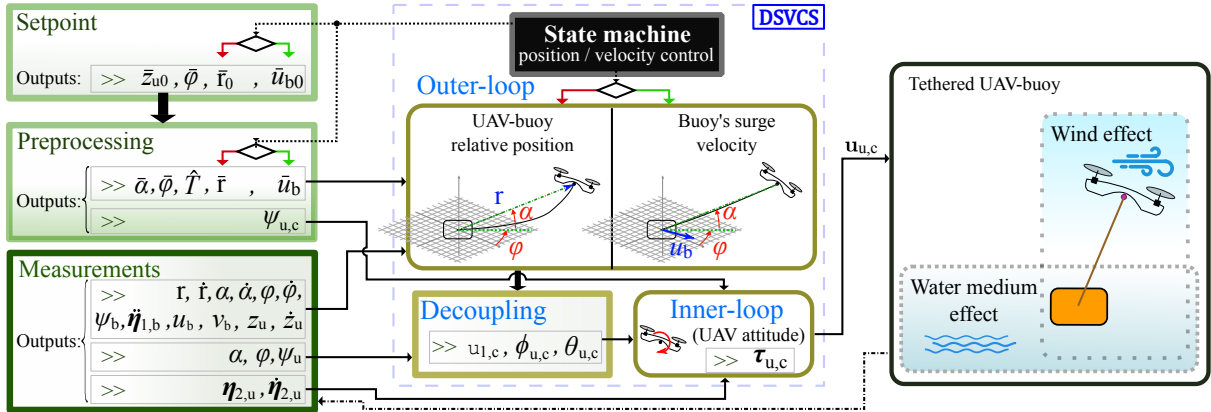


Figure 3.4: Architecture of the proposed Directional Surge Velocity Control System (DSVCS) for the tethered UAV–buoy system.

tion [64] and results in safer and more predictable paths. This can be achieved by computing the resulting reference elevation angle, α , based on the desired reference elevation, \bar{z}_u , as:

$$\bar{\alpha} = \text{asin}\left(\frac{(\bar{z}_u - z_b)}{\bar{r}}\right). \quad (3.46)$$

The azimuth angle is chosen to set the pulling direction, and the UAV's yaw angle can be independently manipulated without affecting the other states; here, we set it equal to azimuth angle in order to keep the UAV directed forward while an onboard camera can still point towards the buoy at all times:

$$\bar{\psi}_u = \bar{\varphi}. \quad (3.47)$$

Note that the steady-state velocity vector of the buoy in the horizontal plane of \mathcal{W} , $\bar{\psi}_V$, does not necessarily point in the same direction as $\bar{\varphi}$, since the buoy is free to slide sideways, such that:

$$\bar{\psi}_V = \bar{\varphi} + \epsilon_\psi, \quad (3.48)$$

where ϵ_ψ is an error angle that is related to the sideslip angle of the buoy, β_u . Finally, the radial position, \bar{r}_0 , and the velocity setpoint, \bar{u}_{b0} , are smoothed by low-pass filters of fourth- and second-order, respectively, which results in a smoother performance due to respecting the system dynamics [39].

UAV–Buoy Relative Position Control Law (Outer-Loop)

To design the relative position control law, we must refer to the UAV dynamics in \mathcal{W}' where the states of interest are explicitly expressed. If we reorder the realization of $\ddot{\mathbf{r}}_u$ in (3.10b) and multiply both sides by m_u , we get:

$$m_u \ddot{\mathbf{r}} = \mathbf{R}_{C2S}(m_u \ddot{\mathbf{r}}_u) - m_u (\mathbf{R}_{C2S} \ddot{\mathbf{r}}_b). \quad (3.49)$$

The first right-hand side term of (3.49) can be seen as the mapping of the translational terms of (3.27) into \mathcal{W}' (kinetics), and the second one can be seen as the mapping of the buoy's linear accelerations into \mathcal{W}' (kinematics). Expanding (3.49) yields:

$$\begin{aligned} m_u(\ddot{r} - r\dot{\alpha}^2 - rc_\alpha^2\dot{\varphi}^2) &= \mathbf{u}_r - T - m_u g s_\alpha - m_u(c_\alpha c_\varphi \ddot{x}_b + c_\alpha s_\varphi \ddot{y}_b + s_\alpha \ddot{z}_b), \\ m_u(r^2\ddot{\alpha} + 2r\dot{r}\dot{\alpha} + r^2s_\alpha c_\alpha \dot{\varphi}^2) &= r\mathbf{u}_\alpha - m_u g r c_\alpha - m_u r(-s_\alpha c_\varphi \ddot{x}_b - s_\alpha s_\varphi \ddot{y}_b + c_\alpha \ddot{z}_b), \\ m_u(r^2c_\alpha \ddot{\varphi} + 2rc_\alpha \dot{r}\dot{\varphi} - 2r^2s_\alpha \dot{\alpha}\dot{\varphi}) &= r\mathbf{u}_\varphi - m_u r(-s_\varphi \ddot{x}_b + c_\varphi \ddot{y}_b). \end{aligned} \quad (3.50)$$

Consider the case of nonzero tension for the relative position dynamics of the UAV–buoy system's in (3.50), with states vectors $\mathbf{X}_1 = [r, \alpha, \varphi]^\top$ and $\mathbf{X}_2 = [\dot{r}, \dot{\alpha}, \dot{\varphi}]^\top$, control input vector $\mathbf{U} = [\mathbf{u}_r, \mathbf{u}_\alpha, \mathbf{u}_\varphi]^\top$, subject to unknown external disturbances including water currents, gravity waves, and wind gusts. The equations of motion in the kinetic form are expressed as:

$$\mathbf{M}_{\text{dc}}\ddot{\mathbf{X}}_{\text{dc}} = \mathbf{H}_{\text{dc}} + \mathbf{\Phi}_{\text{dc}}\mathbf{\Theta}_{\text{dc}} + \mathbf{b}_{\text{dc}}\mathbf{U} + \boldsymbol{\delta}_{\text{dc}}, \quad (3.51)$$

where the subscript (dc) refers to the decoupled dynamics, $\ddot{\mathbf{X}}_{\text{dc}} = \ddot{\mathbf{X}}_1$, the inertia matrix $\mathbf{M}_{\text{dc}} = m_u \text{diag}(1, r^2, r^2c_\alpha)$, the parameter vector $\mathbf{\Theta}_{\text{dc}} = T$, the regressor vector $\mathbf{\Phi}_{\text{dc}} = [-1; 0; 0]$, and $\mathbf{b}_{\text{dc}} = \text{diag}(1, r, r)$. $\boldsymbol{\delta}_{\text{dc}}$ represents the vector of lumped modeling errors and disturbances, and the vector $\mathbf{H}_{\text{dc}} = [H_{\text{dc},1}; H_{\text{dc},2}; H_{\text{dc},3}]$ represents all of the nonlinear Euler, Coriolis, centrifugal, and gravitational forces and moments, and is given by:

$$\begin{aligned} H_{\text{dc},1} &= m_u(r\dot{\alpha}^2 + rc_\alpha^2\dot{\varphi}^2 - c_\alpha c_\varphi \ddot{x}_b - c_\alpha s_\varphi \ddot{y}_b - s_\alpha \ddot{z}_b - g s_\alpha), \\ H_{\text{dc},2} &= m_u r(-2\dot{r}\dot{\alpha} - rs_\alpha c_\alpha \dot{\varphi}^2 + s_\alpha c_\varphi \ddot{x}_b + s_\alpha s_\varphi \ddot{y}_b - c_\alpha \ddot{z}_b - g c_\alpha), \\ H_{\text{dc},3} &= m_u r(-2\dot{r}\dot{\varphi}c_\alpha + 2r\dot{\alpha}\dot{\varphi}s_\alpha + s_\varphi \ddot{x}_b - c_\varphi \ddot{y}_b). \end{aligned}$$

The state-space model in (3.51) is formulated as the following time-varying second-order nonlinear system:

$$\begin{aligned} \dot{\mathbf{X}}_1 &= \mathbf{X}_2, \\ \dot{\mathbf{X}}_2 &= \mathbf{H} + \mathbf{\Phi}\mathbf{\Theta} + \mathbf{b}\mathbf{U} + \boldsymbol{\delta}, \end{aligned} \quad (3.52)$$

where

$$\begin{aligned} \mathbf{b} &= \mathbf{M}_{\text{dc}}^{-1}\mathbf{b}_{\text{dc}}, \quad \mathbf{\Phi} = \mathbf{M}_{\text{dc}}^{-1}\mathbf{\Phi}_{\text{dc}}, \quad \mathbf{\Theta} = \mathbf{\Theta}_{\text{dc}}, \\ \boldsymbol{\delta} &= \mathbf{M}_{\text{dc}}^{-1}\boldsymbol{\delta}_{\text{dc}}, \quad \mathbf{H} = \mathbf{M}_{\text{dc}}^{-1}\mathbf{H}_{\text{dc}}. \end{aligned}$$

Assumption 12. *The external disturbances and modeling errors are bounded along with their derivatives.*

Assumption 13. *The tethered UAV–buoy system is expected to be deployed in relatively calm weather conditions, thus we can state that the lumped error vector δ is constant or slowly varying during a finite time interval, that is:*

$$\lim_{t_1 < t < t_2} \dot{\delta}_r, \dot{\delta}_\alpha, \dot{\delta}_\varphi \approx 0.$$

Let $\bar{\mathbf{X}}_1 = [\bar{r}, \bar{\alpha}, \bar{\varphi}]^\top$ be the reference state vector, and $\mathbf{e}_1 = \mathbf{X}_1 - \bar{\mathbf{X}}_1$ be the respective state error vector. The UAV’s relative position (outer-loop) is controlled by a backstepping control law, which includes the thrust components (radial, elevation, and azimuth) in spherical coordinates, and it is given by [72]:

$$\begin{aligned} \mathbf{U} &= \mathbf{b}^{-1} \left[-\mathbf{k}_P \mathbf{e}_1 - \mathbf{k}_I \mathbf{e}_1^I - \mathbf{k}_D \dot{\mathbf{e}}_1 + \ddot{\bar{\mathbf{X}}}_1 - \mathbf{H} - \Phi \hat{\Theta} \right], \\ \dot{\mathbf{e}}_1^I &= \mathbf{e}_1, \end{aligned} \quad (3.53)$$

where \mathbf{k}_P , \mathbf{k}_I , and $\mathbf{k}_D \in \mathbb{R}_{>0}^{3 \times 3}$ are controller gains that are defined next, and $\hat{\Theta}$ is the estimate of Θ .

Theorem 3. *Consider the state-space representation in (3.52) of the UAV–buoy’s relative position dynamics in (3.50). If Assumption 12 holds true, the control law in (3.53) generates the thrust vector element in \mathcal{W}' , $\{\mathbf{u}_r, \mathbf{u}_\alpha, \mathbf{u}_\varphi\}$, that can stabilize the outer-loop dynamics of the system and reduce the tracking error to a small region neighboring the origin in finite time for a set of gains \mathbf{k}_1 , \mathbf{k}_2 , and $\gamma \in \mathbb{R}_{>0}^{3 \times 3}$, such that $\mathbf{k}_P = \mathbf{I}_3 + \mathbf{k}_1 \mathbf{k}_2 + \gamma$, $\mathbf{k}_I = \gamma \mathbf{k}_1$, and $\mathbf{k}_D = \mathbf{k}_1 + \mathbf{k}_2$, with \mathbf{I}_3 being the identity matrix. Additionally, if Assumption 13 holds, the tracking error vanishes to zero in finite time.*

Proof. We employ the backstepping control design, which involves two steps. Let $\mathcal{V}_1 = \frac{1}{2} \mathbf{e}_1^\top \mathbf{e}_1$ be the first candidate Lyapunov function, and let $\dot{\mathcal{V}}_1 = \mathbf{e}_1^\top \dot{\mathbf{e}}_1$ be its time derivative. We proceed to a second step as $\dot{\mathbf{e}}_1$ does not include an explicit control input. To stabilize \mathbf{e}_1 , we define a virtual control input as $\Upsilon = \ddot{\bar{\mathbf{X}}}_1 - \mathbf{k}_1 \mathbf{e}_1$, followed by a virtual rates error as: $\mathbf{e}_2 = \dot{\bar{\mathbf{X}}}_1 - \Upsilon$.

We then define a second Lyapunov function as:

$$\mathcal{V}_2 = \frac{1}{2} \mathbf{e}_1^\top \mathbf{e}_1 + \frac{1}{2} \mathbf{e}_2^\top \mathbf{e}_2 + \frac{1}{2} \tilde{\delta}^\top \gamma^{-1} \tilde{\delta},$$

with $\tilde{\delta} = \hat{\delta} - \delta$, then we differentiate it to obtain:

$$\begin{aligned} \dot{\mathcal{V}}_2 &= \mathbf{e}_1^\top \dot{\mathbf{e}}_1 + \mathbf{e}_2^\top \dot{\mathbf{e}}_2 + \tilde{\delta}^\top \gamma^{-1} \dot{\tilde{\delta}} \\ &= \mathbf{e}_1^\top (\mathbf{e}_2 - \mathbf{k}_1 \mathbf{e}_1) + \mathbf{e}_2^\top (\mathbf{H} + \Phi \hat{\Theta} + \mathbf{b} \mathbf{U} + \delta - \dot{\Upsilon}) + \tilde{\delta}^\top \gamma^{-1} \dot{\tilde{\delta}}. \end{aligned}$$

Finally, the control inputs vector and the update rates of the lumped disturbances and modeling errors are chosen to satisfy the negative semi-definiteness condition of $\dot{\mathcal{V}}_2$, as:

$$\begin{aligned} \mathbf{U} &= \mathbf{b}^{-1} \left(-\mathbf{H} - \hat{\delta} - \Phi \hat{\Theta} + \dot{\Upsilon} - \mathbf{e}_1 - \mathbf{k}_2 \mathbf{e}_2 \right), \\ \dot{\hat{\delta}} &= \gamma \mathbf{e}_2, \end{aligned} \quad (3.54)$$

which results in $\dot{\mathcal{V}}_2 = -\mathbf{e}_1^\top \mathbf{k}_1 \mathbf{e}_1 - \mathbf{e}_2^\top \mathbf{k}_2 \mathbf{e}_2$. Via Barbalat's lemma with Assumption 13, the asymptotic convergence of \mathcal{V}_2 to zero is guaranteed. Note that if Assumption 13 is violated by the presence of strong wave disturbances or wind gusts, stability and finite tracking error are still achieved by increasing the controller gains, such that they overcome the disturbances mismatch effect on $\dot{\mathcal{V}}_2$. Finally, the control law in the PID-like form in (3.53) is obtained by substituting \mathbf{e}_2 and $\dot{\mathbf{Y}}$ in (3.54), then setting $\mathbf{e}_1^I := \hat{\boldsymbol{\delta}}(\boldsymbol{\gamma} \mathbf{k}_1)^{-1}$. \square

Buoy Surge Velocity Control Law (Outer-Loop)

If the cable is taut, its length becomes constant and \dot{r} and \ddot{r} can be set to zero. Substituting l for r in (3.50) yields the UAV's motion equations in \mathcal{W}' in spherical coordinates notation:

$$\begin{aligned} m_u(-l\dot{\alpha}^2 - lc_\alpha^2\dot{\varphi}^2) &= \mathbf{u}_r - m_u g s_\alpha - T - m_u(c_\alpha c_\varphi \ddot{x}_b + c_\alpha s_\varphi \ddot{y}_b + s_\alpha \ddot{z}_b), \\ m_u(l^2\ddot{\alpha} + l^2 s_\alpha c_\alpha \dot{\varphi}^2) &= l\mathbf{u}_\alpha - m_u g l c_\alpha - m_u l(-s_\alpha c_\varphi \ddot{x}_b - s_\alpha s_\varphi \ddot{y}_b + c_\alpha \ddot{z}_b), \\ m_u(l^2 c_\alpha \ddot{\varphi} - 2l^2 s_\alpha \dot{\alpha} \dot{\varphi}) &= l\mathbf{u}_\varphi - m_u l(-s_\varphi \ddot{x}_b + c_\varphi \ddot{y}_b), \end{aligned} \quad (3.55)$$

In order to control the buoy's surge velocity, u_b , it must be explicitly expressed in (3.55). The buoy's acceleration in the body frame is described as: $\dot{\boldsymbol{\nu}}_{1,b} = \mathbf{R}_{1,b}^\top \ddot{\boldsymbol{\eta}}_{1,b}$. When waves with relatively long periods and moderate heights are dominant, the resulting buoy's roll and pitch angles are small and cannot be used by the controller unless a proper measurement technique is available for the UAV. On the other hand, the USV/buoy heading can be visually estimated with good accuracy as was demonstrated in [27]. For this reason, we limit the transformation of $\mathbf{R}_{1,b}$ to yaw only, such that $\dot{\boldsymbol{\nu}}_{1,b} \approx \mathbf{R}_{z_b}^\top \ddot{\boldsymbol{\eta}}_{1,b}$. Hence, \ddot{x}_b and \ddot{y}_b in (3.55) can be transformed into \dot{u}_b and \dot{v}_b accordingly.

Consider the UAV dynamics in \mathcal{W}' in the coupled case, while following the spherical coordinates notation as presented in (3.55). By choosing the state vector $\mathbf{X}'_1 = [u_b, \alpha, \varphi]^\top$, we can rewrite (3.55) in the kinetic form as:

$$\mathbf{M}_{\text{cp}} \ddot{\mathbf{X}}_{\text{cp}} = \mathbf{H}_{\text{cp}} + \boldsymbol{\Phi}_{\text{cp}} \boldsymbol{\Theta}_{\text{cp}} + \mathbf{b}_{\text{cp}} \mathbf{U}' + \boldsymbol{\delta}_{\text{cp}}, \quad (3.56)$$

where the subscript (cp) refers to the coupled dynamics, $\ddot{\mathbf{X}}_{\text{cp}} = \ddot{\mathbf{X}}'_1$, the control vector $\mathbf{U}' = \mathbf{U}$, the parameter vector $\boldsymbol{\Theta}_{\text{cp}} = T$, the regressor vector $\boldsymbol{\Phi}_{\text{cp}} = [-1; 0; 0]$, and $\mathbf{b}_{\text{cp}} = \text{diag}(1, l, l)$. $\boldsymbol{\delta}_{\text{cp}}$ represents the vector of lumped modeling errors and disturbances, the inertia matrix is expressed as:

$$\mathbf{M}_{\text{cp}} = m_u \begin{bmatrix} c_\alpha(c_\varphi c_{\psi_b} + s_\varphi s_{\psi_b}) & 0 & 0 \\ -l s_\alpha(c_\varphi c_{\psi_b} + s_\varphi s_{\psi_b}) & l^2 & 0 \\ -l(s_\varphi c_{\psi_b} - c_\varphi s_{\psi_b}) & 0 & l^2 c_\alpha \end{bmatrix},$$

and the vector $\mathbf{H}_{\text{cp}} = [H_{\text{cp},1}; H_{\text{cp},2}; H_{\text{cp},3}]$ represents all of the nonlinear Euler,

Coriolis, centrifugal, and gravitational forces and moments, with:

$$\begin{aligned} H_{\text{cp},1} &= m_{\text{u}}(l\dot{\alpha}^2 + lc_{\alpha}^2\dot{\varphi}^2 - s_{\alpha}\ddot{z}_{\text{b}} - gs_{\alpha} + c_{\alpha}(c_{\varphi}s_{\psi_{\text{b}}} - s_{\varphi}c_{\psi_{\text{b}}})\dot{v}_{\text{b}}), \\ H_{\text{cp},2} &= m_{\text{u}}l(-ls_{\alpha}c_{\alpha}\dot{\varphi}^2 - c_{\alpha}\ddot{z}_{\text{b}} - gc_{\alpha} - s_{\alpha}(c_{\varphi}s_{\psi_{\text{b}}} - s_{\varphi}c_{\psi_{\text{b}}})\dot{v}_{\text{b}}), \\ H_{\text{cp},3} &= m_{\text{u}}l(2l\dot{\alpha}\dot{\varphi}s_{\alpha} - (s_{\varphi}s_{\psi_{\text{b}}} + c_{\varphi}c_{\psi_{\text{b}}})\dot{v}_{\text{b}}). \end{aligned}$$

The state-space model in (3.56) is formulated as a time-varying second-order nonlinear system as:

$$\begin{aligned} \dot{\mathbf{X}}'_1 &= \mathbf{X}'_2, \\ \dot{\mathbf{X}}'_2 &= \mathbf{H}' + \Phi'\Theta' + \mathbf{b}'\mathbf{U}' + \delta', \end{aligned} \quad (3.57)$$

where

$$\begin{aligned} \mathbf{b}' &= \mathbf{M}_{\text{cp}}^{-1}\mathbf{b}_{\text{cp}}, \quad \Phi' = \mathbf{M}_{\text{cp}}^{-1}\Phi_{\text{cp}}, \quad \Theta' = \Theta_{\text{cp}}, \\ \delta' &= \mathbf{M}_{\text{cp}}^{-1}\delta_{\text{cp}}, \quad \mathbf{H}' = \mathbf{M}_{\text{cp}}^{-1}\mathbf{H}_{\text{cp}}. \end{aligned}$$

The outer-loop velocity controller manipulates the state vector \mathbf{X}'_1 , and a control law for the surge velocity can be designed in a similar fashion as described in Section 3.3.2, with a difference that only one step is required in the backstepping process for the state u_{b} . Let $\bar{\mathbf{X}}'_1 = [\bar{u}_{\text{b}}, \bar{\alpha}, \bar{\varphi}]^{\top}$ be the reference state vector and $\mathbf{e}'_1 = \mathbf{X}'_1 - \bar{\mathbf{X}}'_1$ be the respective state vector error. The control law is defined as:

$$\begin{aligned} \mathbf{U}' &= \mathbf{b}'^{-1}[-\mathbf{k}'_{\text{p}}\mathbf{e}'_1 - \mathbf{k}'_{\text{I}}\mathbf{e}'_1 - \mathbf{k}'_{\text{D}}\dot{\mathbf{e}}'_1 + \ddot{\bar{\mathbf{X}}}'_1 - \mathbf{H}' - \Phi'\hat{\Theta}'], \\ \dot{\mathbf{e}}'_1 &= \mathbf{e}'_1, \end{aligned} \quad (3.58)$$

where $\ddot{\bar{\mathbf{X}}}'_1 = [\ddot{\bar{u}}_{\text{b}}, \ddot{\bar{\alpha}}, \ddot{\bar{\varphi}}]$; $\mathbf{k}'_{\text{p}} = \text{diag}(0, 1 + k'_{1,\alpha}k'_{2,\alpha} + \gamma'_{\alpha}, 1 + k'_{1,\varphi}k'_{2,\varphi} + \gamma'_{\varphi})$, $\mathbf{k}'_{\text{I}} = \text{diag}(\gamma'_{u_{\text{b}}}, \gamma'_{\alpha}k'_{1,\alpha}, \gamma'_{\varphi}k'_{1,\varphi})$, $\mathbf{k}'_{\text{D}} = \mathbf{k}'_1 + \mathbf{k}'_2$, with $\mathbf{k}'_1 = \text{diag}(0, k'_{1,\alpha}, k'_{1,\varphi})$, $k'_{1,\alpha}$ and $k'_{1,\varphi} \in \mathbb{R}_{>0}$, $\mathbf{k}'_2 \in \mathbb{R}_{>0}^{3 \times 3}$, and $\boldsymbol{\gamma}' = \text{diag}(\gamma'_{u_{\text{b}}}, \gamma'_{\alpha}, \gamma'_{\varphi}) \in \mathbb{R}_{>0}^{3 \times 3}$, are controller gains that are defined next; and $\hat{\Theta}'$ is the estimate of Θ' .

Theorem 4. *Consider the state-space representation in (3.57) of the UAV–buoy’s relative position dynamics in (3.55) when the system is coupled. If Assumption 12 holds true, the control law in (3.58) generates the thrust vector elements in the spherical reference frame \mathbf{u}_{r} , \mathbf{u}_{α} , and \mathbf{u}_{φ} , that can stabilize the outer-loop dynamics of the system, and reduce the tracking error to a small region neighboring the origin in finite time for a set of gains \mathbf{k}'_1 , \mathbf{k}'_2 , and $\boldsymbol{\gamma}' \in \mathbb{R}_{\geq 0}^{3 \times 3}$, defined in a similar way to their counterparts in Theorem 3. Additionally, if Assumption 13 holds, the tracking error is reduced to zero in finite time.*

Proof. We employ the backstepping control design, which involves one step for the u_{b} state, and two steps for the α and φ states. To keep a compact representation of the proof in vector form, the buoy velocity states are tackled in the second

step. First, the Lyapunov function $\mathcal{V}'_1 = \frac{1}{2}e_\alpha^2 + \frac{1}{2}e_\varphi^2$ is proposed, and let its derivative be expressed as $\dot{\mathcal{V}}'_1 = e_\alpha \dot{e}_\alpha + e_\varphi \dot{e}_\varphi$. We proceed to a second step in the control design process for the states' rates given that \dot{e}_α and \dot{e}_φ do not include an explicit control input. To stabilize \mathbf{e}_1 , we define a virtual control input as: $\mathbf{\Upsilon}' = \dot{\mathbf{X}}_1'^* - \mathbf{k}'_1 \mathbf{e}'_1$, where $\mathbf{k}'_1 = \text{diag}(0, k'_{1,\alpha}, k'_{1,\varphi})$ and $\dot{\mathbf{X}}_1'^* = [\bar{u}_b, \dot{\bar{\alpha}}, \dot{\bar{\varphi}}]^\top$. Next, we define the rates error as: $\mathbf{e}'_2 = \dot{\mathbf{X}}_1' - \mathbf{\Upsilon}'$. Note that the rates error of the buoy velocity's x -component simply reduces to $e'_{2,1} = u_b - \bar{u}_b$.

Here, we define a second Lyapunov function as:

$$\mathcal{V}'_2 = \frac{1}{2}e_\alpha^2 + \frac{1}{2}e_\varphi^2 + \frac{1}{2}\mathbf{e}'_2{}^\top \mathbf{e}'_2 + \frac{1}{2}\tilde{\boldsymbol{\delta}}'^\top \boldsymbol{\gamma}'^{-1} \tilde{\boldsymbol{\delta}}',$$

with $\tilde{\boldsymbol{\delta}}' = \hat{\boldsymbol{\delta}}' - \boldsymbol{\delta}'$, then we differentiate to have:

$$\begin{aligned} \dot{\mathcal{V}}'_2 &= e_\alpha \dot{e}_\alpha + e_\varphi \dot{e}_\varphi + \mathbf{e}'_2{}^\top \dot{\mathbf{e}}'_2 + \tilde{\boldsymbol{\delta}}'^\top \boldsymbol{\gamma}'^{-1} \dot{\tilde{\boldsymbol{\delta}}}' \\ &= e_\alpha (e'_{2,2} - k'_{1,\alpha} e_\alpha) + e_\varphi (e'_{2,3} - k'_{1,\varphi} e_\varphi) \\ &\quad + \mathbf{e}'_2{}^\top (\mathbf{b}'\mathbf{U}' + \mathbf{H}' + \boldsymbol{\Phi}'\hat{\boldsymbol{\Theta}}' + \boldsymbol{\delta}' - \dot{\boldsymbol{\Upsilon}}') + \tilde{\boldsymbol{\delta}}'^\top \boldsymbol{\gamma}'^{-1} \dot{\tilde{\boldsymbol{\delta}}}'. \end{aligned}$$

Finally, the control inputs vector and the update rates of the lumped disturbances and modeling errors are chosen to satisfy the negative semi-definiteness of $\dot{\mathcal{V}}'_2$, as:

$$\begin{aligned} \mathbf{U}' &= \mathbf{b}'^{-1} (-\boldsymbol{\Phi}'\hat{\boldsymbol{\Theta}}' - \mathbf{H}' - \hat{\boldsymbol{\delta}}' + \dot{\boldsymbol{\Upsilon}}' - \mathbf{e}'_1^* - \mathbf{k}'_2 \mathbf{e}'_2), \\ \dot{\tilde{\boldsymbol{\delta}}}' &= \boldsymbol{\gamma}' \mathbf{e}'_2, \quad \mathbf{e}'_1^* = [0, e_\alpha, e_\varphi]^\top, \end{aligned} \tag{3.59}$$

where the tuning gain $\mathbf{k}'_2 \in \mathbb{R}^{3 \times 3}$ is a diagonal matrix, and we get $\dot{\mathcal{V}}'_2 = -k'_{1,\alpha} e_\alpha^2 - k'_{1,\varphi} e_\varphi^2 - \mathbf{e}'_2{}^\top \mathbf{k}'_2 \mathbf{e}'_2$. \square

UAV's Attitude Controller (Inner-Loop)

Let $\phi_{u,c}$ and $\theta_{u,c}$ be the desired roll and pitch angles of the UAV, respectively, which can be obtained from the outputs of the outer-loop controller along with the UAV's total thrust command, $\mathbf{u}_{1,c}$. The transformation from thrust components in spherical coordinates to the UAV's thrust and tilt angles is performed by referring to (3.25) and (3.26) in two steps. We let the command thrust vector in the world frame, \mathcal{W} , be $\mathbf{U}_{C,c} = [\mathbf{u}_{x,c}, \mathbf{u}_{y,c}, \mathbf{u}_{z,c}]^\top = \mathbf{R}_{u,1}[0, 0, \mathbf{u}_{1,c}]^\top$, and in the spherical frame, \mathcal{W}' , be $\mathbf{U}_{S,c} = [\mathbf{u}_{r,c}, \mathbf{u}_{\alpha,c}, \mathbf{u}_{\varphi,c}]^\top = \mathbf{R}_{C2S}\mathbf{U}_{C,c}$.

In the first step, given the force commands in the spherical frame generated from the outer-loop controller, $\mathbf{U}_{S,c}$, we calculate $\mathbf{U}_{C,c} = \mathbf{R}_{S2C}\mathbf{U}_{S,c}$, which expands to:

$$\begin{aligned} \mathbf{u}_{x,c} &= c_\alpha c_\varphi \mathbf{u}_{r,c} - s_\alpha c_\varphi \mathbf{u}_{\alpha,c} - s_\varphi \mathbf{u}_{\varphi,c}, \\ \mathbf{u}_{y,c} &= c_\alpha s_\varphi \mathbf{u}_{r,c} - s_\alpha s_\varphi \mathbf{u}_{\alpha,c} + c_\varphi \mathbf{u}_{\varphi,c}, \\ \mathbf{u}_{z,c} &= s_\alpha \mathbf{u}_{r,c} + c_\alpha \mathbf{u}_{\alpha,c}. \end{aligned} \tag{3.60}$$

In the second step, we calculate $\phi_{u,c}$ and $\theta_{u,c}$ given that $\mathbf{U}_{C,c} = \mathbf{R}_{u,1}[0, 0, \mathbf{u}_{1,c}]^\top$, which expands to:

$$\begin{aligned}\mathbf{u}_{x,c} &= \mathbf{u}_{1,c}(s_{\theta_{u,c}}c_{\psi_{u,c}} + s_{\phi_{u,c}}c_{\theta_{u,c}}s_{\psi_{u,c}}), \\ \mathbf{u}_{y,c} &= \mathbf{u}_{1,c}(s_{\theta_{u,c}}s_{\psi_{u,c}} - s_{\phi_{u,c}}c_{\theta_{u,c}}c_{\psi_{u,c}}), \\ \mathbf{u}_{z,c} &= \mathbf{u}_{1,c}(c_{\phi_{u,c}}c_{\theta_{u,c}}).\end{aligned}\quad (3.61)$$

With mathematical manipulation, we get the following relationships:

$$\begin{aligned}\mathbf{u}_{1,c} &= \sqrt{\mathbf{u}_{x,c}^2 + \mathbf{u}_{y,c}^2 + \mathbf{u}_{z,c}^2}, \\ \phi_{u,c} &= \arctan(\mathbf{u}_{x,c}s_{\psi_u} - \mathbf{u}_{y,c}c_{\psi_u})/\mathbf{u}_{z,c}, \\ \theta_{u,c} &= \arcsin(\mathbf{u}_{x,c}c_{\psi_u} + \mathbf{u}_{y,c}s_{\psi_u})/\mathbf{u}_{1,c}.\end{aligned}\quad (3.62)$$

Let $\phi'_{u,c} = \phi_{u,m} \tanh(\phi_{u,c}/\phi_{u,m})$ and $\theta'_{u,c} = \theta_{u,m} \tanh(\theta_{u,c}/\theta_{u,m})$ be smooth and bounded versions of $\phi_{u,c}$ and $\theta_{u,c}$, where $\phi_{u,m}$ and $\theta_{u,m} \in (0, \frac{\pi}{2})$ are the absolute upper limits of the UAV's roll and pitch angles, respectively. The UAV's yaw dynamics can be controlled independently of the buoy's manipulation.

Consider the UAV's rotational dynamics in (3.27) with state vectors $\mathbf{X}_1'' = [\phi_u, \theta_u, \psi_u]^\top$ and $\mathbf{X}_2'' = [\dot{\phi}_u, \dot{\theta}_u, \dot{\psi}_u]^\top$, and control input vector $\mathbf{U}'' = [u_2, u_3, u_4]^\top$, subject to unknown external disturbances like wind gusts. The equations of motion in the kinetic form is expressed as:

$$\mathbf{M}_{ur}\ddot{\mathbf{X}}_{ur} = \mathbf{H}_{ur} + \Phi_{ur}\Theta_{ur} + \mathbf{b}_{ur}\mathbf{U}'' + \delta_{ur}, \quad (3.63)$$

where the subscript (ur) refers to the UAV's rotational dynamics, $\ddot{\mathbf{X}}_{ur} = \ddot{\mathbf{X}}_1''$, the inertia matrix $\mathbf{M}_{ur} = \mathbf{M}_{2,u}$, the parameter vector $\Theta_{ur} = T$, the regressor vector $\Phi_{ur} = (\mathbf{r}_{GH,u}) \times (\mathbf{R}_{1,u}^\top \mathbf{R}_{S2C}[-1; 0; 0])$, and $\mathbf{b}_{ur} = \text{diag}(1, 1, 1)$. δ_{ur} represents the vector of lumped modeling errors and disturbances, and $\mathbf{H}_{ur} = -\mathbf{C}_{2,u}\dot{\boldsymbol{\eta}}_{2,u} - \mathbf{D}_{2,u}\ddot{\boldsymbol{\eta}}_{2,u}$ represents all the nonlinear Coriolis, centrifugal, and damping moments. The state-space model in (3.63) is formulated as the following time-varying second-order nonlinear system:

$$\begin{aligned}\dot{\mathbf{X}}_1'' &= \mathbf{X}_2'', \\ \dot{\mathbf{X}}_2'' &= \mathbf{H}'' + \Phi''\Theta'' + \mathbf{b}''\mathbf{U}'' + \delta'',\end{aligned}\quad (3.64)$$

where

$$\begin{aligned}\mathbf{b}'' &= \mathbf{M}_{ur}^{-1}\mathbf{b}_{ur}, \quad \Phi'' = \mathbf{M}_{ur}^{-1}\Phi_{ur}, \quad \Theta'' = \Theta_{ur}, \\ \delta'' &= \mathbf{M}_{ur}^{-1}\delta_{ur}, \quad \mathbf{H}'' = \mathbf{M}_{ur}^{-1}\mathbf{H}_{ur}.\end{aligned}$$

Let $\bar{\mathbf{X}}_1'' = [\bar{\phi}_u, \bar{\theta}_u, \bar{\psi}_u]^\top$ be the reference state vector, and $\mathbf{e}_1'' = \mathbf{X}_1 - \bar{\mathbf{X}}_1''$ be the respective state error vector. The proposed control law that stabilizes the UAV's rotational dynamics is defined as:

$$\begin{aligned}\mathbf{U}'' &= \mathbf{b}''^{-1}[-\mathbf{k}_p''\mathbf{e}_1'' - \mathbf{k}_f''\mathbf{e}_1''^I - \mathbf{k}_d''\dot{\mathbf{e}}_1'' + \ddot{\bar{\mathbf{X}}}_1'' - \mathbf{H}'' - \Phi''\hat{\Theta}''], \\ \dot{\mathbf{e}}_1''^I &= \mathbf{e}_1'',\end{aligned}\quad (3.65)$$

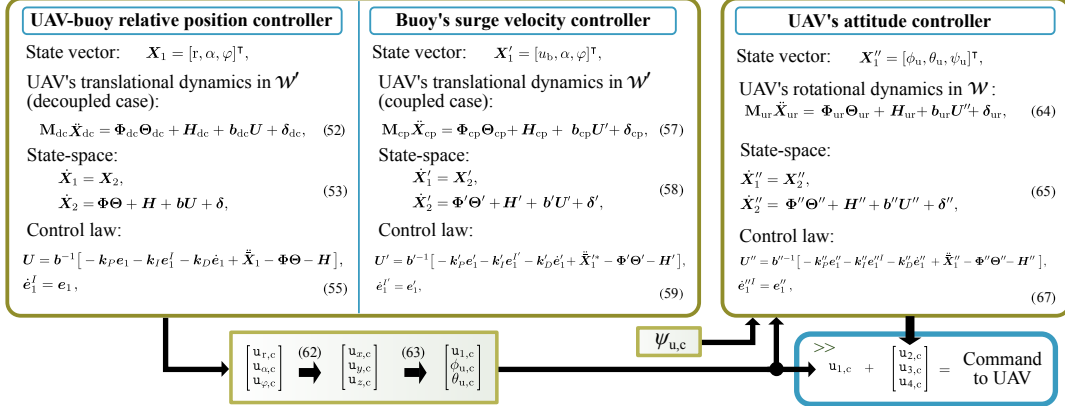


Figure 3.5: Detailed control map of the DSVCS for the tethered UAV–buoy system showing the two outer-loop (position/velocity) controllers and the inner-loop (attitude) controller.

where k''_P , k''_I , and $k''_D \in \mathbb{R}_{>0}^{3 \times 3}$ are controller gains, and $\hat{\Theta}''$ is the estimate of Θ'' . Equivalent outcomes of Theorem 3 are applicable for the UAV's inner-loop controller, and can be proved by replicating the procedure of Theorem 3's proof, which is omitted for brevity.

3.3.3 Parameter Estimation

If the taut-cable condition in (3.30) holds, the following realization of the cable tension is given:

$$T = \begin{cases} ((m_b + a_{11})\dot{u}_b + D'_{b,11}u_b)/(c_{\psi_b}c_{\alpha}c_{\varphi} + s_{\psi_b}c_{\alpha}s_{\varphi}), & \left| \alpha - \frac{\pi}{2} \right| > \epsilon_{\alpha} \\ (m_u\dot{w}_u + m_u g(c_{\phi_u}c_{\theta_u}) - u_1)/(c_{\psi_u}c_{\alpha}c_{\varphi} + s_{\psi_u}c_{\alpha}s_{\varphi}), & \left| \alpha - \frac{\pi}{2} \right| \leq \epsilon_{\alpha}, \end{cases} \quad (3.66)$$

where $\epsilon_{\alpha} \in \mathbb{R}_{\geq 0}$ is a constant that prevents singularity in a small region near $\alpha = \frac{\pi}{2}$. The cable tension expression in (3.66) can be determined from the sum of the first and second rows of the buoy dynamics in (3.21), which shows a direct link with u_b for the first case. The singularity near the vertical cable configuration ($\alpha = \pi/2$) is handled by the UAV dynamics in (3.27) to compute the actual cable tension, T , as in the second case of (3.66). To get an estimate \hat{T} of T , as required by the control laws of the DSVCS, (3.66) should be used. Here, we note that even if \hat{T} is inaccurate, the controller is able to compensate for the tension effect based on its integral term action.

3.4 Simulations

This section presents numerical simulations for validating the 3D tethered UAV–buoy robotic system model, and for evaluating the performance of the

Table 3.1: Three-dimensional tethered UAV–buoy system model parameters

Par.	Value	Unit	Par.	Value	Unit
l_b	0.8	m	m_u	5.0	kg
h_b	0.25	m	J_u	$diag(0.16, 0.16, 0.22)$	$kg\ m^2$
m_b	12.5	kg	$\phi_{u,m}$	$\pi/4$	rad
J_b	$diag(0.07, 0.21, 0.21)$	$kg\ m^2$	$\theta_{u,m}$	$\pi/4$	rad
\mathbf{a}_1	$diag(0.625, 19.6, 12.5)$	kg	l	7	m
\mathbf{a}_2	$diag(0.065, 0.068, 0.105)$	kg	C_u	[0.2, 0.2, 0.2]	-
\mathbf{b}_1	$diag(0, 0, 27.5)$	Ns/m	A_{cs}^u	[0.08, 0.08, 0.1]	m^2
\mathbf{b}_2	$diag(0, 0, 0)$	Ns/m	ρ_w	1000	kg/m^3
g	9.81	m/s^2	ρ_a	1.22	kg/m^3
$\mathbf{r}_{GH,b}$	[0.4; 0; 0]	m	$\mathbf{r}_{GH,u}$	[0; 0; 0]	m

proposed control system in allowing the UAV to manipulate the buoy in the horizontal plane as well as the tether’s elevation and azimuth angles. We validate the system in realistic environmental and operating conditions including wind, waves, and water current. To increase the model’s fidelity, the UAV model includes the motor and propeller dynamics, and the controller uses non-exact feedback signals (e.g. noise and bias), which simulate measured outputs based on state-of-the-art navigation sensors [31].

3.4.1 Simulation Settings

The validation of the proposed system is carried in the MATLAB Simulink[®] simulation environment. The system includes a medium-sized quadrotor UAV tethered to a small buoy having a boat-like shape to enable self-alignment along the pulling direction. The parameters of the system are included in Table 3.1. The UAV’s thrust-to-weight ratio is considered about 2.5, which gives a maximum thrust of about 120 N. The motor’s model is considered a low-pass filter of the first-order with a time constant $\tau_m = 0.05$ s. The buoy’s immersed volume, wetted area, and skin friction coefficients are calculated as per [31]. Additionally, the added mass and damping are calculated based on the strip theory for surface vessels at low oscillating frequencies with their values presented in Table 3.1 [67].

We note that a Cartesian-based PID controller was tested in [31] to check whether it can control the tethered system, however, it resulted in unsatisfactory performance relative to instability. Hence, in this work, only the DSVCS is implemented and validated. The controller gains are set to $\mathbf{k}_P = diag(45, 9.6, 9.6)$, $\mathbf{k}_I = diag(9, 5.6, 5.6)$, $\mathbf{k}_D = diag(19.5, 1.6, 1.6)$, $\mathbf{k}'_P = diag(25, 9.6, 9.6)$, $\mathbf{k}'_I = diag(12, 5.6, 5.6)$, $\mathbf{k}'_D = diag(0, 1.6, 1.6)$, $\mathbf{k}''_P = diag(10, 10, 30)$, $\mathbf{k}''_I = diag(0.2, 0.2, 0.6)$, $\mathbf{k}''_D = diag(5, 5, 15)$.

The controller is fed with conditioned virtual measurements of the required feedback signals. We assume that the UAV is equipped with a Global Positioning System / Inertial Navigation System (GPS/INS) system and a stereo camera, which enable measuring the UAV's pose and its position in the spherical frame. Before being used by the controller in simulation, we apply sensor characteristics (bias, linearity, resolution, and accuracy, noise), on the actual signals that correspond to typical state-of-the-art sensing technologies. For an entity (\bullet), we denote the mean absolute value of its estimation error by $\text{mav}(\tilde{\bullet})$, and we set $\text{mav}(\tilde{\phi}_u) = \text{mav}(\tilde{\theta}_u) = \text{mav}(\tilde{\psi}_u) = 0.5^\circ$, $\text{mav}(\tilde{x}_u) = \text{mav}(\tilde{y}_u) = \text{mav}(\tilde{z}_u) = 0.02 \text{ m}$, $\text{mav}(\tilde{\psi}_b) = 5^\circ$, $\text{mav}(\tilde{\alpha}) = \text{mav}(\tilde{\varphi}) = 0.16^\circ$, and $\text{mav}(\tilde{r}) = 0.02 \text{ m}$. Subsequently, we use (3.9) and (3.10) to obtain the buoy's translational states.

3.4.2 Simulation Scenarios

The designed controller's performance and the fidelity of the derived system model are validated with different simulation scenarios. We first simulate two cases that include constant water current, wind gust, and moderate waves. To further validate the system in additional conditions, we offer extended simulation scenarios (in Section 3.4.4) by varying the water current direction, the wave components in the water environment, and other system parameters such as buoy size and cable length.

The first two cases, C1 and C2, both include a wind gust of $\mathbf{U}_{\text{wd}} = [-5, 0, 0] \text{ m s}^{-1}$ and a water current component with $U_1 = 1 \text{ m s}^{-1}$ and $\psi_1 = \pi \text{ rad}$. The two scenarios include:

- C1: water current and wind gust only.
- C2: water current, wind gust, and moderate waves with one wave components ($N = 1$), such that: $A_1 = 0.75 \text{ m}$, $\psi_{1,w} = 0$, $T_1 = 5.7 \text{ s}$, and $\sigma_1 = \pi$.

In both cases, the UAV is initially hovering around the buoy, it then positions itself in a location that is suitable to start pulling the buoy at mean elevation, \bar{z}_u , of 5.0 m , which corresponds to a mean elevation angle of $\bar{\alpha}_0 = 45^\circ$. In the first stage, the UAV pulls the buoy until it reaches a surge velocity of $\bar{u}_b = 5 \text{ m s}^{-1}$. In the second stage, the UAV gradually steers the buoy by commanding an azimuth angle $\bar{\varphi}$ that increases from 0 to 360° , thus completing a full rotation. The resulting profile is shown in Fig. 3.7.

The buoy has an initial velocity that matches its surrounding water as calculated via (3.12) and (3.14); by referring to Assumption 3.39, we assume that buoy stays tangent to the water surface, such that:

$$\begin{aligned}\phi_b &= \phi_w, \\ \theta_b &= \theta_w,\end{aligned}\tag{3.67}$$

while its yaw dynamics are governed by (3.17).

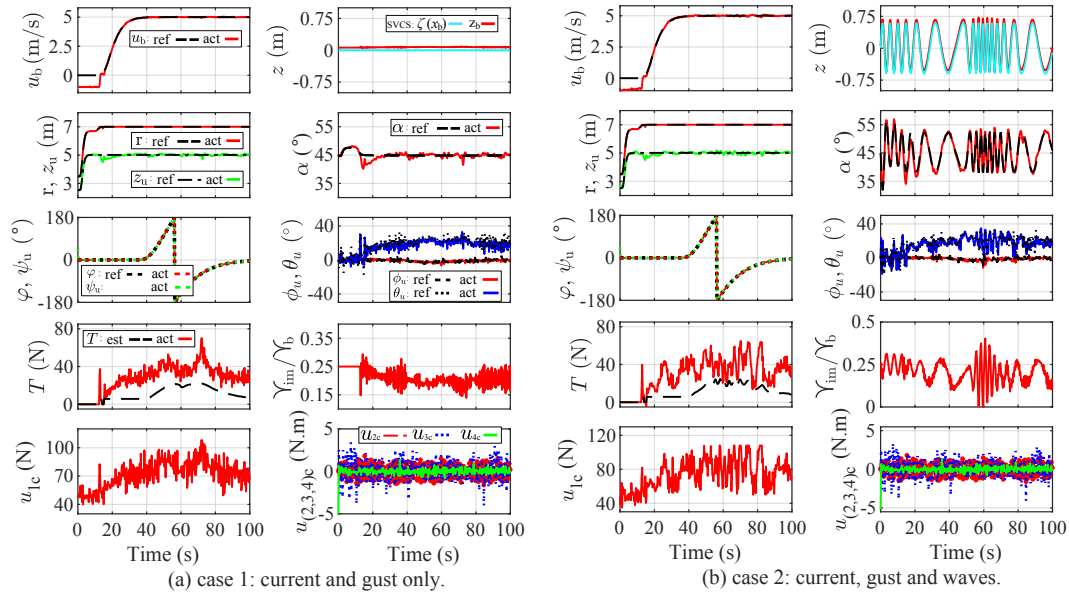


Figure 3.6: States and control inputs of the directional surge velocity control system (DSVCS) for simulation scenarios C1 (a) and C2 (b).

3.4.3 Simulation Results and Discussion

The simulation results for C1 and C2 are shown in Fig. 3.6a (left) and Fig. 3.6b (right), respectively. Stable and accurate performance is noticed in both cases, where the UAV is able to successfully drive the buoy at the desired velocity (u_b), while maintaining good tracking of the reference elevation (α) and azimuth (φ) angles. As seen in the (r) subplot, the UAV first positions itself near the pulling location, it then enters the pulling state by making the cable taut during the entire duration of the buoy's manipulation between [12, 100]s. Since there are no ambient waves in C1, there is no apparent fluctuation in the buoy's vertical position (z_b), immersed volume (γ_b), and elevation angle (α); contrarily, the waves in C2 induce fluctuations in these three states. It is noted that the UAV keeps a level flight in both cases, as seen in the (z_u) subplot in Fig. 3.6a-b. The UAV is commanded to remain oriented forward during the entire maneuver, which induces large pitch angles (θ_u) to counteract disturbances, the roll angle (ϕ_u) remains near-constant, and the yaw angle (ψ_u) smoothly follows the azimuth angle of the cable (φ) in order to complete a full rotation during the course of the motion trajectory. The full rotation in azimuth (φ) causes the buoy to encounter the waves in different directions, where the frequency of encounter is low during the initial phase ([20, 40] s) due to following seas, and when the buoy turns around ([50, 60] s) the encounter frequency becomes high due to head seas. The cable tension, seen in the (T) subplot, increases with velocity when the frequency of encounter with waves is high, and when the rate of change of the azimuth angle

is high. The proposed controller can compensate for the estimation error in T through its integral term and quick performance. Finally, we note that the thrust and torques inputs ($\mathbf{u}_{(1,2,3,4)c}$) are bounded and stable.

Even though the DSVCS achieves good tracking of the desired surge and azimuth states, it does not necessarily translate into identical trajectories in the world frame. This is demonstrated in Fig. 3.7, which shows the planar trajectory of the buoy for cases C1 and C2. It is seen that the two trajectories do not coincide, even though the results of Fig. 3.6 showed good tracking with no steady-state errors. It is also noticed that the final direction of the trajectory is not exactly at zero as seen in the ‘ xy ’ subplot of Fig. 3.7, even though the final azimuth angle is zero. This can be understood by observing the non-zero sway velocity, v_b , in the ‘Vel.’ subplot of Fig. 3.7. We also notice that the sway velocity of the buoy, v_b , can be as large as its surge velocity, u_b , which results in a larger absolute surface velocity of the buoy, V . Finally, we note that the buoy is prone to having a nonzero side-slip angle, β_b , as seen in Fig. 3.7, given that it does not have any constraint on its planar dynamics. At the start of the simulation ($< 17s$) when the cable is slack, the side-slip angle points to a backward motion ($\beta_b = 180^\circ$) since $u_b \approx -1 \text{ m s}^{-1}$ and $v_b \approx 0 \text{ m s}^{-1}$ due to the initial water current velocity.

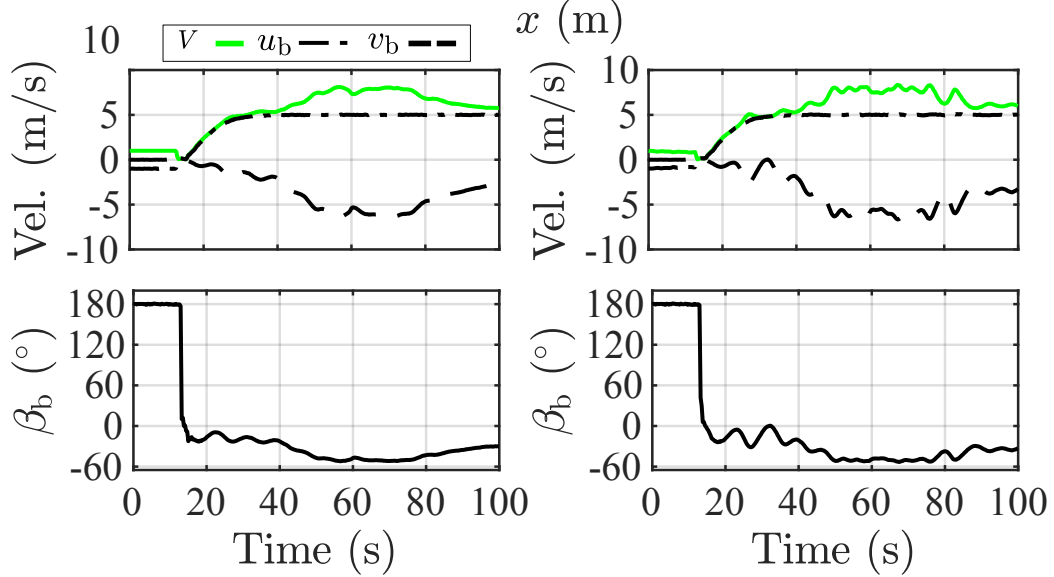
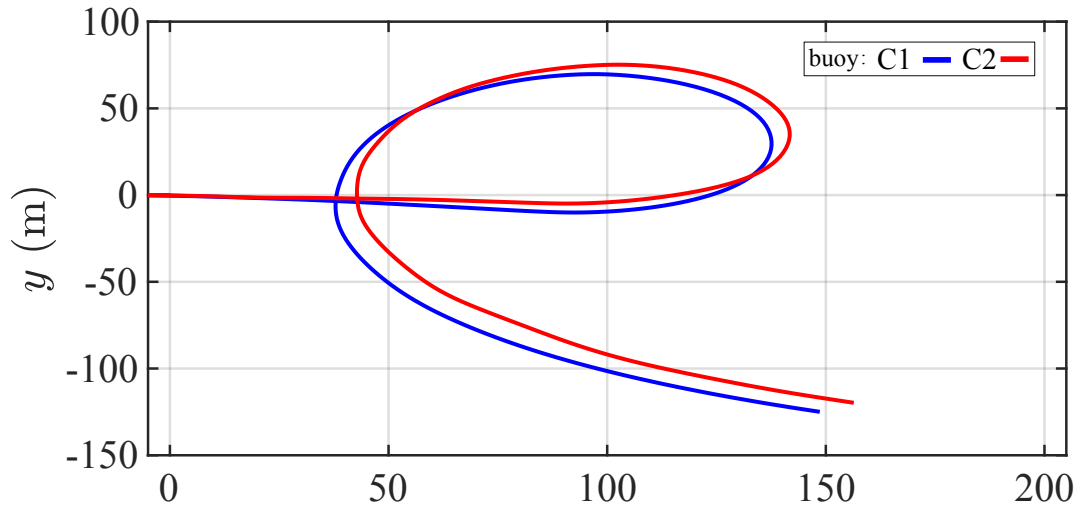
As mentioned in Chapter 3.3, the above demonstration motivates the need for a higher level path planner in the event that the buoy is required to track trajectories in the inertial frame. That said, the obtained simulation results demonstrate the effectiveness of the DSVCS in controlling the buoy’s surge velocity, u_b , and the direction of the pulling force by manipulating the azimuth (φ) and elevation (α) angles. This provides the system with the ability to move forward and steer as a locomotive, which also lays the foundation for designing a path planner in the inertial frame.

3.4.4 Extended Simulations

To further validate the system performance in different conditions, we offer additional simulation scenarios by varying the water current direction, the wave components in the water environment, and well as changing the system parameters (buoy size and tether length). The physical system index is referred to as S_i where $i = \{1, 2\}$, and the environment index is referred to as E_j where $j = \{1, \dots, 7\}$. Hence, we introduce the following system variations:

(a) System size variants (buoy and cable):

- S1. Same system presented in Section 3.4.1, except for the cable length: $l = 10 \text{ m}$ (S1), and $m_b = 16 \text{ kg}$.
- S2. Same system as (S1) except for cable length: $l = 15 \text{ m}$ (S2).



(a) case 1: current and gust only.

(b) case 2: current, gust, and waves.

Figure 3.7: The buoy's location (top), velocity (middle), and side-slip angle (bottom) during the locomotion task in C1 (a) and C2 (b).

(b) Wave environment variants:

E1. No waves, no current.

E2. No waves, with water current $U_c = 1 \text{ m s}^{-1}$, $\psi_c = -\pi/2 \text{ rad}$.

E3. Same current as E2, with two wave components ($N = 2$): $A = [0.135, 0.305] \text{ m}$, $T = [3, 4] \text{ s}$, $\psi_w = [0, 0] \text{ rad}$, and $\sigma = [0, 0] \text{ rad}$.

E4. No waves, with water current $U_c = 1 \text{ m s}^{-1}$, $\psi_c = \pi/2 \text{ rad}$.

E5. Same current as E4, with two wave components ($N = 2$): $A = [0.305, 0.6] \text{ m}$, $T = [4, 5] \text{ s}$, $\psi_w = [0, 0] \text{ rad}$, and $\sigma = [0, 0] \text{ rad}$.

E6. No waves, with water current $U_c = 1 \text{ m s}^{-1}$, $\psi_c = 0 \text{ rad}$.

E7. Same current as E6, with a single high-amplitude wave components ($N = 1$): $A = [1.65] \text{ m}$, $T = [8] \text{ s}$, $\psi_w = [0] \text{ rad}$, and $\sigma = [0] \text{ rad}$.

The UAV is commanded to drive the buoy at $\bar{u}_b = 3 \text{ m s}^{-1}$ while making three turns by going from $\bar{\varphi} = 0$ to $\bar{\varphi} = 6\pi$ with a constant slope.

Fig. 3.8 shows the resulting trajectories of the simulation in environments E1 through E7, where system S1 is adopted with E1 through E5, and S2 is adopted with E6 and E7. From E1, we notice that without wave and current disturbances, the system can follow a circular trajectory by simply commanding a constant velocity and azimuth angle with a constant ramp. If water current exists, the buoy drifts towards the direction of the current as seen in E2, E4, and E6. When the system is deployed in wavy seas, additional drift may occur due to the waves' Stokes-drift effect, which increases with wave height, as seen by comparing the additional drift between E3 (lowest deviation with respect to the current-only case), E5, and E7 (highest deviation with respect to the current-only case), whereas the oscillating component of the waves' velocity tends to cancel out near the end. Finally, we see that the system is able to deal with relatively large waves and reject their respective disturbances, as observed in E7.

In all cases, tracking of the controlled states is achieved with small steady-state errors, which is practical for applications that do not require precision motion control. One room for improvement can be achieving better trajectory tracking in the world frame, which is sub-optimal due to the fact that the DSVCS has no direct effect on the buoy's sway velocity, and thus it cannot trace repetitive trajectories in the world frame. Again, if tracking trajectories in the world frame is desired for certain applications, that can be achieved by employing a high level path planner by means of commanding appropriate combinations of u_b and φ , which is not within the scope of this current work.

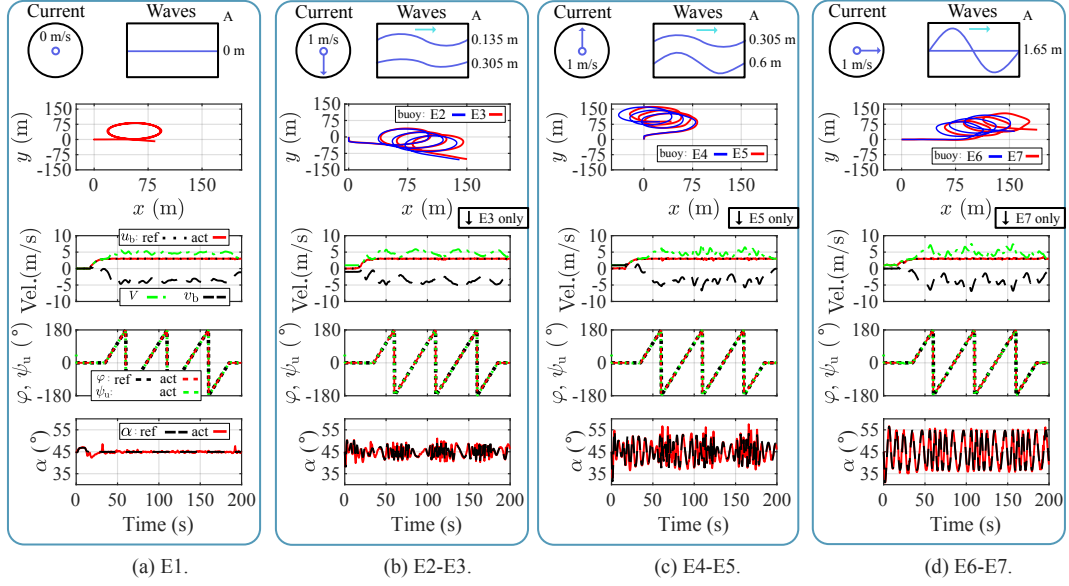


Figure 3.8: Trajectory and main states tracking for variants of the tethered UAV–buoy system in different environments: (a) E1, (b) E2 and E3, (c) E4 and E5, (d) E6 and E7. In all case, $l = 10$ m except in E6 and E7 where $l = 15$ m.

3.5 Conclusion

This chapter extended the 2D marine locomotive UAV problem, realized as a tethered quadrotor UAV–buoy system, to the 3D space to enable the UAV to steer the buoy in the horizontal plane, and not being restricted to forward and backward motion only [93]. The buoy and the UAV were considered as 6-DOF rigid bodies, and their dynamics were derived with a proper definition of all effective forces and moments on the system. A detailed derivation of the coupled system dynamics was included to serve as a reference for the rigorous mathematical formulation and modeling of the heterogeneous multi-physics robotic system under consideration, which features high order, nonlinear, and coupled dynamics. Finally, the control problem for applying a tensile force via the tether with specific magnitude and direction was solved through the proposed DSVCS, which was realized by controlling the buoy’s surge velocity and the tether’s orientation.

Future work naturally flows into designing a path planner to control the buoy’s location in the world frame, and studying the effects of the tether and the buoy’s tilt dynamics on the system.

CHAPTER 4

A PRACTICAL GUIDELINE FOR DESIGNING AND TUNING ADAPTIVE BACKSTEPPING CONTROLLERS FOR A CLASS OF SECOND-ORDER SYSTEMS BASED ON PID SIMILARITY

4.1 INTRODUCTION

In this chapter, we motivate and introduce the backstepping-based Integrated Control System (ICS) used in the control of the tethered UAV–buoy system, and we show the advantages of employing such a controller.

4.1.1 Backstepping-PID Similarity

A similarity in the formulation of backstepping-based control laws with the PID control law has been established. For instance, backstepping control with integral action was used to design a robust adaptive PID controller for linear systems in [56]. The previous formulation was later modified in [94] by incorporating the integral action within the first backstepping error, and it was shown that the regulation poles can be placed at any stable location. Later, various forms of the backstepping method with integral action were applied in [95] to a class of nonlinear second-order systems, where the resulting control law was formulated as a feedback PID plus feedforward terms, and the characteristics of each form were compared, albeit without making the connection of the designs to the resultant closed-loop pole locations and overall system performance. Later works on this

subject were mostly application-based. For instance, a backstepping-based PID controller was designed in [96] for a quadrotor UAV system, where the form proposed by [94] was generalized to include nonlinear terms. In addition, the developed backstepping analysis tools were employed to design a robust PID controller with guaranteed performance in [97], yet it was only applied to the linearized version of a quadrotor’s inner-loop controller. Last but not least, an explicit PID-based tuning guideline of backstepping controllers was proposed for a class of second order nonlinear systems in [72].

4.1.2 Motivation

The motivation of this work lies in the need for control systems that are easy to design and deploy based on PID control laws, yet advanced enough to deal with uncertainties and disturbances. Even though some prior works have established a link between the backstepping design method and PID control, the similarity remains underappreciated and has not been fully exploited, as evidenced by the fact that the general implicit form of backstepping control design still dominates newly published works in the related literature. On the other hand, a select few robotic applications benefited from the PID representation of the backstepping method to achieve easier controller tuning with guaranteed closed-loop system stability [31, 54, 72, 85, 96, 97].

4.1.3 Contribution

This work formulates the similarity between the gains of PID and adaptive backstepping controllers in a form that may enable easier and wider adoption of this nonlinear control method. Here, we emphasize that the scope of this work focuses on establishing an original formulation of the link between existing control techniques, not on providing a new or superior control system design. We present a compact formulation of the adaptive backstepping control law, which consists of two components: PID feedback and feedforward model compensation, for a class of second-order nonlinear systems, which cover a wide spectrum of real-life practical systems [65]. The practicality of this work lies in the unique link that it establishes between backstepping controller gains and the PID gains, whereby tuning the adaptive backstepping gains can be achieved by standard PID tuning rules (e.g., Ziegler–Nichols). We also make available on GitHub the relevant scripts that are needed to compute the gains and plot the various relationships between the backstepping and PID gains. Furthermore, the hereby presented formulation provides insight into the contents of the virtual backstepping commands, which allows for a more prudent employment of filters, thus reducing excessive time-delays. Moreover, we note that a manual tuning process may prove useful and superior since it is based on intuitive understanding of the control law, whereas optimization-based techniques typically require complex or

non-intuitive efforts, as well as knowledge of exact system parameters, which is not always feasible [62].

This work offers a unique representation of the similarity between the PID gains structure and the adaptive backstepping gains of a class of nonlinear second-order systems subject to uncertainties, and features multiple advantages. First, we represent the similarity to the PID gains with the minimal number of parameters (three) as compared to the representations in [94, 96] that use four parameters. Second, we use a minimum number of dependencies between the adaptive backstepping gains and the PID gains (one multiplication and one summation) as compared to [56, 94], which use up to two multiplications and two summations. Third, this work offers a unique and explicit analysis of the implications on this similarity in terms of gains selection and stability, powered by clear and easy-to-follow guidelines in the form of tables, charts, and algorithm. Fourth, we provide a survey on the formulations of the backstepping-PID structure similarity, which makes it easier to choose the suitable method and directly apply it to the systems at hand. These differences are substantiated in a comparative table that is later provided to compare this work to similar works in the literature (Table 4.1).

4.1.4 Structure

This chapter starts with defining the problem under consideration in Section 4.2. A baseline controller is presented in Section 4.3, followed by its reformulation into the proposed form in Section 4.4, which also provides the tuning guidelines after establishing the similarity link to PID control. The proposed methodology is validated in Section 4.5 on a quadrotor UAV system in numerical simulations as well as in experimentation. Finally, the work is summarized in Section 4.6 and suggestions on future work are provided.

4.2 Problem Formulation

In this section, we introduce the general form of second-order systems, which are amongst the most popular and practical system models, and that may benefit from the nonlinear control law in the PID-feedback plus feedforward-model-compensation form presented in this work.

4.2.1 Preliminaries

Notations that facilitate the formulations developed in this paper are first provided here. The set of positive-real numbers $\{x \in \mathbb{R} \mid x > 0\}$, and the set of non-negative real numbers $\{x \in \mathbb{R} \mid x \geq 0\}$, are respectively denoted by $\mathbb{R}_{>0}$ and $\mathbb{R}_{\geq 0}$. The mean absolute value of a signal (\cdot) is denoted by $MAV(\cdot)$, and its

L_2 norm is denoted by $\|\cdot\|$. Also, for an angle (\bullet) , the cosine, sine, and tangent functions are denoted by c_\bullet , s_\bullet , and t_\bullet , respectively.

In addition, for an estimate \hat{x} of $x \in [x_{\min}, x_{\max}]$ and a signal (\cdot) , let the projection function, $\text{Proj}_{\hat{x}}(\cdot)$, be defined as [74]:

$$\text{Proj}_{\hat{x}}(\cdot) = \begin{cases} 0 & \text{if } \hat{x} = x_{\min} \text{ and } \cdot < 0 \\ 0 & \text{if } \hat{x} = x_{\max} \text{ and } \cdot > 0 \\ \cdot & \text{otherwise.} \end{cases} \quad (4.1)$$

4.2.2 Problem Formulation

Let $\mathbf{X}_1 = [x_1, \dots, x_n]^\top$ and $\mathbf{X}_2 = [\dot{x}_1, \dots, \dot{x}_n]^\top$ represent the state vectors of a second-order nonlinear time-varying system, and let $\mathbf{U} = [u_1, \dots, u_n]^\top \in \mathbb{R}^n$ represent the control input vector, where $n \in \mathbb{N}^+$. The system is represented as:

$$\begin{aligned} \dot{\mathbf{X}}_1 &= \mathbf{X}_2, \\ \dot{\mathbf{X}}_2 &= \mathbf{g}(\bar{\mathbf{X}}, t)\mathbf{U} + \boldsymbol{\phi}(\bar{\mathbf{X}})\boldsymbol{\theta}(t) + \mathbf{f}(\bar{\mathbf{X}}) + \boldsymbol{\Delta}(t), \end{aligned} \quad (4.2)$$

where $\bar{\mathbf{X}} = \{\mathbf{X}_1, \mathbf{X}_2\}$; $\mathbf{g} \in \mathbb{R}^{n \times n}$ is the input-multiplied unknown nonlinear functions; $\boldsymbol{\theta} \in \mathbb{R}^l$ represents the vector of unknown linear system parameters with $l \in \mathbb{N}^+$; $\boldsymbol{\phi} \in \mathbb{R}^{n \times l}$ is the known nonlinear regressors matrix; $\mathbf{f} \in \mathbb{R}^n$ is the vector of known nonlinear functions; and vector $\boldsymbol{\Delta} = [\Delta_1, \dots, \Delta_n]^\top \in \mathbb{R}^n$ represents the external disturbances and unmodeled system nonlinearities. The controller objective is described by following the reference command $\mathbf{X}_{1d}(t) = [x_{1d}, \dots, x_{nd}]^\top$.

Assumption 14. *It is assumed that \mathbf{X}_2 is bounded, which implies that the functions \mathbf{g} , $\boldsymbol{\phi}$, and \mathbf{f} are bounded along with their first- and second-order derivatives with respect to \mathbf{X}_2 .*

Assumption 15. *$\mathbf{X}_{1d}(t)$ is smooth and bounded up to the second order.*

Assumption 16. *The unmodeled nonlinearities and external disturbances term in $\boldsymbol{\Delta}$, and the parametric uncertainties in $\boldsymbol{\theta}$ and \mathbf{g} are restricted to bounded sets such that:*

$$\begin{aligned} \|\boldsymbol{\Delta}\| &\leq \bar{\Delta}, \\ g_{ij,\min} &\leq g_{ij} \leq g_{ij,\max}, \quad i, j = 1, \dots, n, \\ \theta_{i,\min} &\leq \theta_i \leq \theta_{i,\max}, \quad i = 1, \dots, l, \end{aligned} \quad (4.3)$$

where $\bar{\Delta}$ is a positive constant, and θ_i and g_{ij} represent the elements of $\boldsymbol{\theta}$ and \mathbf{g} , respectively.

With Assumptions 14, 15 and 16, we ensure that the states and signals in the considered system are bounded.

Remark 7. *As a case-study for validating the hereby proposed algorithm, we consider multicopter UAVs, which can be modelled (after proper reduction) as second-order time-varying systems that fit the model presented in (4.2). This work offers a contribution for this type of systems, without claiming generalization to other system types.*

4.3 Adaptive Backstepping Control System Design

In this chapter, we design an adaptive backstepping control law for the second-order nonlinear time-varying system defined in Section 4.2.2 in two steps with virtual inputs. The control law incorporates a model-based parameter estimation scheme and a robust feedback component to improve its transient response, steady-state accuracy, and robustness. This section serves as an overview of the adaptive backstepping control system design process leading to the similarity with PID control laws.

4.3.1 Parameter Estimation Law

The achievement of desirable tracking performance depends on the knowledge of the system parameters, which cannot be guaranteed using the nominal values of $\boldsymbol{\theta}(t)$ and $\mathbf{g}(\bar{\mathbf{X}}, t)$. Hence, the control law should employ their estimates, $\hat{\boldsymbol{\theta}}(t)$ and $\hat{\mathbf{g}}(t)$, instead of their unknown (true) values. The parameter estimation algorithm relies on a recursive least squares (RLS) adaptation law with projection mapping to provide a model-based estimation of the system's unknown parameters, which tend to be faster and more accurate than error-based estimates. Since this is outside the scope of this paper, readers are referred to [98, 99] for details on the RLS estimator design. The parameter estimation vector includes $\boldsymbol{\theta}$ and the linear combination of unknown parameters in the term $\mathbf{g}(\bar{\mathbf{X}}, t)\mathbf{U}$. Note that $\hat{\mathbf{g}}$ is conditioned to remain invertible.

4.3.2 Controller Design

In this section, and for the sake of notation simplicity and compactness, we do not explicitly express the in-brackets vector dependencies, that we use \mathbf{g} instead of $\mathbf{g}(\mathbf{x}, t)$. Hence, the error dynamics vector is given by:

$$\mathbf{e}_1 = \mathbf{X}_1 - \mathbf{X}_{1d}. \quad (4.4)$$

Two steps are required in the backstepping control design process for second-order systems. In the first step, we propose a candidate Lyapunov function, \mathcal{V}_1 ,

based on the error dynamics, and its derivative $\dot{\mathcal{V}}_1$, to be:

$$\begin{aligned}\mathcal{V}_1 &= \frac{1}{2} \mathbf{e}_1^\top \mathbf{e}_1, \\ \dot{\mathcal{V}}_1 &= \mathbf{e}_1^\top \dot{\mathbf{e}}_1 = \mathbf{e}_1^\top (\dot{\mathbf{X}}_1 - \dot{\mathbf{X}}_{1d}),\end{aligned}\tag{4.5}$$

where \mathbf{e}_1 can be stabilized by employing a virtual control input, $\boldsymbol{\alpha} \in \mathbb{R}^n$, that substitutes $\dot{\mathbf{X}}_1$, as follows:

$$\boldsymbol{\alpha} = \dot{\mathbf{X}}_{1d} - \mathbf{k}_1 \mathbf{e}_1,\tag{4.6}$$

where $\mathbf{k}_1 \in \mathbb{R}_{>0}^{n \times n}$ is a diagonal matrix. In step two of the backstepping controller design, we define the second error term relating the state vector derivative to the virtual input as follows:

$$\mathbf{e}_2 = \mathbf{X}_2 - \boldsymbol{\alpha}.\tag{4.7}$$

The error dynamics are obtained by differentiating (4.7) and substituting $\dot{\mathbf{X}}_2$ from (4.2):

$$\dot{\mathbf{e}}_2 = \mathbf{f} + \boldsymbol{\phi} \boldsymbol{\theta} + \mathbf{g} \mathbf{U} + \boldsymbol{\Delta} - \dot{\boldsymbol{\alpha}},\tag{4.8}$$

which explicitly includes the control input vector \mathbf{U} . However, due to the variable system parameters and the presence of unknown external disturbances, (4.8) is expressed in the following form:

$$\dot{\mathbf{e}}_2 = \mathbf{f} + \boldsymbol{\phi} \hat{\boldsymbol{\theta}} + \hat{\mathbf{g}} \mathbf{U} + \mathbf{d} - \dot{\boldsymbol{\alpha}},\tag{4.9}$$

where $\mathbf{d} = [d_1, \dots, d_n]^\top$ lumps the errors in modeling and estimation with the external disturbances. The term \mathbf{d} is split into two components: one for low-frequency signals, \mathbf{d}_c , and the other for high-frequency signals, $\tilde{\mathbf{d}}^*$ [98]:

$$\mathbf{d} := \mathbf{d}_c + \tilde{\mathbf{d}}^* = -\tilde{\mathbf{g}} \mathbf{U} - \boldsymbol{\phi} \tilde{\boldsymbol{\theta}} + \boldsymbol{\Delta},\tag{4.10}$$

where $\tilde{\mathbf{g}} = \hat{\mathbf{g}} - \mathbf{g}$ and $\tilde{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}} - \boldsymbol{\theta}$.

Remark 8. *The system disturbances and uncertainties are bounded as per Assumption 16, such that $\|\mathbf{d}\| \leq \bar{d}$, with $\bar{d} \in \mathbb{R}_{\geq 0}$. Additionally, vector $\boldsymbol{\Delta}$ includes both disturbances and unmodeled nonlinearities, thus it can be split as per (4.10) into low-frequency and high-frequency terms: the low-frequency terms can be compensated for via an integrator, or they can be included in a parameter estimation scheme if they were structured; and high-frequency terms are handled by a robust control component [74] as shown next.*

Finally, the proposed nominal control law possesses the form:

$$\begin{aligned}\mathbf{U} &= \hat{\mathbf{g}}^{-1} (\dot{\boldsymbol{\alpha}} - \boldsymbol{\phi} \hat{\boldsymbol{\theta}} - \mathbf{f} - \mathbf{e}_1 - \mathbf{k}_2 \mathbf{e}_2 + \mathbf{u}_r - \hat{\mathbf{d}}_c), \\ \mathbf{u}_r &= -\frac{1}{4\epsilon} h^2 \mathbf{e}_2, \quad \dot{\hat{\mathbf{d}}}_c = \text{Proj}_{\hat{\mathbf{d}}_c} (\gamma \mathbf{e}_2),\end{aligned}\tag{4.11}$$

where $\epsilon \in \mathbb{R}_{>0}$ is a design parameter that influences the error attenuation level, steady-state accuracy, and transient response of the tracking error; and h is any smooth function that guaranties the following [61]:

$$h \geq \|\phi\| \|\boldsymbol{\theta}_{\max} - \boldsymbol{\theta}_{\min}\| + \|\mathbf{U}\| \|\mathbf{g}_{\max} - \mathbf{g}_{\min}\| + \|\bar{\Delta}\|, \quad (4.12)$$

$\text{Proj}_{\hat{\mathbf{d}}_c}(\cdot) = [\text{Proj}_{\hat{\mathbf{d}}_{c1}}(\cdot), \dots, \text{Proj}_{\hat{\mathbf{d}}_{cn}}(\cdot)]^\top$, and $\boldsymbol{\gamma} \in \mathbb{R}_{\geq 0}^{n \times n}$ is a diagonal matrix that adjusts the adaptation rate. Note that $\text{Proj}_{\hat{\mathbf{d}}_c}(\cdot)$ limits $\hat{\mathbf{d}}_c$ to the set $\Omega_{\mathbf{d}_c} \triangleq \{\hat{\mathbf{d}}_c : -\bar{d} \leq \hat{\mathbf{d}}_c \leq \bar{d}\}$.

Assumption 17. *It is safe to assume that $\dot{\mathbf{d}}_c = 0$ given that \mathbf{d}_c includes low-frequency signals only.*

Theorem 5. *Consider the system model described in (4.2) that is subject to unknown disturbances, and whose fidelity is affected by unmodeled nonlinearities and parametric uncertainties. Also, let Assumptions 14, 15, 16, and 17 hold true. The stability of this system and the boundedness of its signals are guaranteed by the control law (4.11), with $\boldsymbol{\alpha}$ defined in (4.6) and \mathbf{e}_2 defined in (4.7), for some gains \mathbf{k}_1 and \mathbf{k}_2 [72].*

Proof. We start by defining the augmented candidate Lyapunov function, \mathcal{V}_2 , differentiating it once with respect to time, then substituting \mathbf{d} from (4.10) and \mathbf{U} from (4.11) into (4.9) to obtain:

$$\begin{aligned} \mathcal{V}_2 &= \frac{1}{2} \mathbf{e}_1^\top \mathbf{e}_1 + \frac{1}{2} \mathbf{e}_2^\top \mathbf{e}_2 + \frac{1}{2} \tilde{\mathbf{d}}_c^\top \boldsymbol{\gamma}^{-1} \tilde{\mathbf{d}}_c, \\ \dot{\mathcal{V}}_2 &= -\mathbf{e}_1^\top \mathbf{k}_1 \mathbf{e}_1 - \mathbf{e}_2^\top \mathbf{k}_2 \mathbf{e}_2 + \tilde{\mathbf{d}}_c^\top \boldsymbol{\gamma}^{-1} \dot{\hat{\mathbf{d}}}_c \\ &\quad + \mathbf{e}_2^\top (\mathbf{d}_c - \hat{\mathbf{d}}_c + \tilde{\mathbf{d}}^* + \mathbf{u}_r), \end{aligned} \quad (4.13)$$

where $\tilde{\mathbf{d}}_c = \hat{\mathbf{d}}_c - \mathbf{d}_c$, and $\dot{\mathbf{d}}_c = 0$ per Assumption 17. By substituting $\dot{\hat{\mathbf{d}}}_c$ from (4.11) in (4.13), we get: $\dot{\mathcal{V}}_2 = -\mathbf{e}_1^\top \mathbf{k}_1 \mathbf{e}_1 - \mathbf{e}_2^\top \mathbf{k}_2 \mathbf{e}_2 + \mathbf{e}_2^\top (\tilde{\mathbf{d}}^* + \mathbf{u}_r)$. In addition, two robust performance conditions are to be satisfied by the robust feedback term, \mathbf{u}_r , such that [100]:

$$\begin{aligned} \mathbf{e}_2^\top \mathbf{u}_r &\leq 0, \\ \mathbf{e}_2^\top (\mathbf{u}_r - \tilde{\mathbf{d}}_c + \tilde{\mathbf{d}}^*) &\leq \epsilon, \end{aligned} \quad (4.14)$$

which is achieved by choosing \mathbf{u}_r as described in (4.11), and robust performance under the prescribed conditions can be guaranteed [100]. This results in $\dot{\mathcal{V}}_2$ being negative semi-definite, thus stability in Lyapunov's sense is concluded. It can also be shown that \mathcal{V}_2 is lower bounded and that $\dot{\mathcal{V}}_2$ is uniformly continuous by Assumptions 14 and 16; therefore, by Barbalat's lemma, the asymptotic convergence of \mathcal{V}_2 to zero is obtained [101]. \square

4.4 From Backstepping to PID Form

In this section, we formulate the backstepping control law in a PID feedback plus feedforward form that has two degrees-of-freedom (2-DOF), then we present a visualization of the obtained relations with ample discussion and practical tips to design and tune the controller gains, which justifies the adoption of the proposed form of the control law. Finally, we relate the backstepping gains to the location of the closed-loop system poles.

4.4.1 Expressing the Backstepping Control Law in PID Form

First, we seek to reformulate and simplify the control law while maintaining its practical results. In practice, a large-enough positive constant, $\mathbf{k}_{2,0}$, can produce robust performance that is similar to that of the nonlinear robust gain, $\frac{1}{4\epsilon}h^2$, in (4.11), such that:

$$\mathbf{k}_{2,0,ii} > n_r \frac{1}{4\epsilon} h^2, \quad i = 1, \dots, n, \quad (4.15)$$

where $n_r > 1$ is a constant multiplier. This causes the effect of \mathbf{u}_r to be similar to that of the linear term $-\mathbf{k}_2 \mathbf{e}_2$. Thus, by choosing the elements of matrix \mathbf{k}_2 to be greater than some positive constants ($\mathbf{k}_2 \succ_{\mathbb{R}_{\geq 0}^{n \times n}} \mathbf{k}_{2,0}$), robust performance can be guaranteed within a large-enough and practical operating range [74, 102]. With this modification, the control law (4.11) is approximated as $\mathbf{U} \approx \mathbf{U}'$, where \mathbf{U}' approximates the effect of the nonlinear robust term, \mathbf{u}_r , by the high gain feedback term, $-\mathbf{k}_2 \mathbf{e}_2$, and can be reformulated by substituting (4.4), (4.6), and (4.7) into (4.11):

$$\begin{aligned} \mathbf{U}' &= \hat{\mathbf{g}}^{-1} [(\ddot{\mathbf{X}}_{1d} - \mathbf{k}_1 \dot{\mathbf{e}}_1) - \phi \hat{\boldsymbol{\theta}} - \mathbf{f} - \mathbf{e}_1 - \mathbf{k}_2 (\dot{\mathbf{X}}_1 - (\dot{\mathbf{X}}_{1d} - \mathbf{k}_1 \mathbf{e}_1)) - \hat{\mathbf{d}}_c] \\ &= \hat{\mathbf{g}}^{-1} [-(\mathbf{k}_1 \mathbf{k}_2 + \mathbf{I}_n) \mathbf{e}_1 - (\mathbf{k}_1 + \mathbf{k}_2) \dot{\mathbf{e}}_1 - \hat{\mathbf{d}}_c - \phi \hat{\boldsymbol{\theta}} - \mathbf{f} + \ddot{\mathbf{X}}_{1d}], \\ \hat{\mathbf{d}}_c &= \text{Proj}_{\hat{\mathbf{d}}_c} (\gamma [\dot{\mathbf{X}}_1 - (\dot{\mathbf{X}}_{1d} - \mathbf{k}_1 \mathbf{e}_1)]) \\ &= \text{Proj}_{\hat{\mathbf{d}}_c} (\gamma (\dot{\mathbf{e}}_1 + \mathbf{k}_1 \mathbf{e}_1)), \end{aligned} \quad (4.16)$$

where $\dot{\mathbf{e}}_1 = \dot{\mathbf{X}}_1 - \dot{\mathbf{X}}_{1d}$ and $\mathbf{I}_n \in \mathbb{R}^{n \times n}$ is the identity matrix. Thus, (4.16) is now expressed as a 2-DOF controller with a feedforward model compensation component and a feedback PID component, designated as the Integrated Control System (ICS):

$$\begin{aligned} \mathbf{U}' &= \hat{\mathbf{g}}^{-1} [-\mathbf{k}_P \mathbf{e}_1 - \mathbf{k}_I \mathbf{e}_1^I - \mathbf{k}_D \dot{\mathbf{e}}_1 - \phi \hat{\boldsymbol{\theta}} - \mathbf{f} + \ddot{\mathbf{X}}_{1d}], \\ \dot{\mathbf{e}}_1^I &= \text{Proj}_{\mathbf{e}_1^I} (\mathbf{e}_1 + \mathbf{k}_1^{-1} \dot{\mathbf{e}}_1), \end{aligned} \quad (4.17)$$

where $\mathbf{e}_1^I = \int \dot{\mathbf{e}}_1^I dt$ and the bounds of the projection function vary accordingly as per (4.1) by a factor $(\gamma \mathbf{k}_1)^{-1}$, and

$$\begin{aligned}\mathbf{k}_P &= \mathbf{k}_1 \mathbf{k}_2 + \mathbf{I}_n, \\ \mathbf{k}_I &= \gamma \mathbf{k}_1, \\ \mathbf{k}_D &= \mathbf{k}_1 + \mathbf{k}_2.\end{aligned}\tag{4.18}$$

The control law (4.17), with its six terms, can now be seen as a combination of an error-based PID feedback component with gains \mathbf{k}_P , \mathbf{k}_I , and \mathbf{k}_D that are included in the first three terms, and a feedforward component based on model compensation desired accelerations that are included in the last three terms.

When operating in the nominal region, it is safe to assume that saturation of the integral term does not occur, which permits bypassing the projection function representation, so that after integrating the term $\mathbf{k}_1^{-1} \dot{\mathbf{e}}_1$, the control law in (4.17) and the control gains in (4.18) undergo the following modifications:

$$\begin{aligned}\dot{\mathbf{e}}_1^I &= \mathbf{e}_1, \\ \mathbf{k}_P &= \mathbf{k}_1 \mathbf{k}_2 + \gamma + \mathbf{I}_n.\end{aligned}\tag{4.19}$$

Let the diagonal elements of an arbitrary single row of matrices \mathbf{k}_P , \mathbf{k}_I , \mathbf{k}_D , \mathbf{k}_1 , \mathbf{k}_2 , and γ be represented by k_P , k_I , k_D , k_1 , k_2 , and γ , respectively. By solving the following third-order equation, which is derived from the definition of \mathbf{k}_P in (4.19) and that of \mathbf{k}_I and \mathbf{k}_D in (4.18), for some desired values of k_P , k_I , and k_D , we can determine the set of equivalent k_1 , k_2 , and γ gains as follows:

$$\begin{aligned}k_1^3 + (-k_D)k_1^2 + (k_P - 1)k_1 - k_I &= 0, \\ k_2 &:= k_D - k_1, \\ \gamma &:= k_I/k_1,\end{aligned}\tag{4.20}$$

where the solution is valid only when k_1 and k_2 are positive and real. To examine the behavior of (4.20), we present a sample solution with respect to k_P for $k_D = 5$ and $k_I = 1$ in Fig. 4.1(a), and with respect to k_D for $k_P = 5$ and $k_I = 1$ in Fig. 4.1(b). Note that the solutions for k_1 (red), k_2 (green), and γ (blue) in each subplot is divided into three subsets, based on their change of direction at the inflection points, as indicated by their line style (Subset #1-solid, Subset #2-dashed, and Subset #3-dotted) to simplify the analysis and arrive at practical recommendations:

1. Subset #1 (solid lines) has a parabola-like shape for k_1 and k_2 in Fig. 4.1(a), and a hyperbola-like shape in Fig. 4.1(b), which result in a wide range of variation for these two gains and thus offers more leverage in tuning them to meet the system's stability and performance requirements. On the other hand, the value and variation of γ are minimal (near zero) as compared to Subsets #2 and #3.

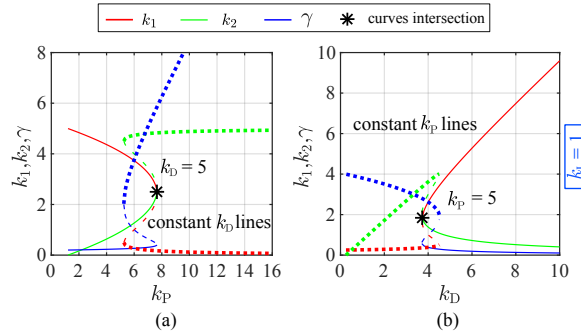


Figure 4.1: Coupled backstepping gains, k_1 , k_2 , and γ , for specific k_P , k_D , and k_I values: (a) at constant k_D and (b) at constant k_P . Three solution subsets exist: Subset #1 - solid lines, Subset #2 - dashed lines, and Subset #3 - dotted lines.

2. Subset #2 (dashed lines) corresponds to the smallest domain of the solution space and features large variation of all three parameters (k_1 , k_2 , and γ). The strong dependencies amongst the parameters, combined with their large variations in this relatively small subset, make it difficult and impractical to interpret the relationship between the backstepping and PID gains within Subset #2. In fact, any formulated relationship is of a limited merit since it encompasses a very small domain of coupled k_P and k_D values.
3. Subset #3 (dotted lines) is characterized by large γ values and a narrow tuning range (nearly constant) for both k_1 (near zero) and k_2 as compared to Subsets #1 and #2, which limits the ability to meet the stability and performance requirements of the system.

Based on the above analysis of Fig. 4.1, it is prudent to only consider the first subset of the solution of (4.20) given its low dependency on γ , high dependency on k_1 and k_2 , and the relatively larger tuning range to achieve the desired system stability and performance requirements. The recommendation to adopt Subset #1 is further justified in Section 4.4.4.

Remark 9. *The differentiated term \dot{e}_1 typically requires filtration in order to reduce the signal's noise level. However, \dot{e}_1 is implicitly embedded within $\dot{\alpha}$ and e_2 of the control law (4.11), whereas the PID-like form of the control law in (4.17) explicitly shows it, which makes it easier to design an appropriate filter with minimal time delay and information loss that may arise from excessive filtration of $\dot{\alpha}$ and e_2 [103].*

Remark 10. *By inspecting (4.18), (4.19), and (4.20), we see that k_1 and k_2 can be used interchangeably. This is achieved by solving for k_2 first instead of k_1 as*

was done in (4.20):

$$\begin{aligned} k_2^3 + (-k_D)k_2^2 + (k_P - 1)k_2 - k_I &= 0, \\ k_1 &:= k_D - k_2, \\ \gamma &:= k_I/(k_D - k_1). \end{aligned} \tag{4.21}$$

Note that the results in (4.21) are identical to those in (4.20), with the only exception being that k_1 and k_2 values have been switched. This can be proven by substituting the interchanged values of k_1 and k_2 into the definition k_D in (4.18) and the definition k_P in (4.19), where the same exact control law in (4.17) can be obtained by setting:

$$k_I := \gamma(k_D - k_1). \tag{4.22}$$

Remark 11. As seen in Fig. 4.1, k_1 is larger than k_2 in the gains solution subset of interest (#1), in which k_2 can have very small values. However, based on Remark 10, k_2 can be the larger gain in the $\{k_1, k_2\}$ solution set, thus the condition for large enough k_2 gain to overtake the effect of the robust gain ($\frac{1}{4\epsilon}h^2$) remains valid.

Remark 12. The gain k_2 's upper bound is governed with respect to the system's sensitivity to noise, and its robust effect is bounded by the maximum control input, thus, the condition in (4.15) has a practical limitation.

4.4.2 Comparison to Other Formulations

To help differentiate this work and to validate the proposed PID-like adaptive backstepping control law, Table 4.1 is provided to summarize previous works that are found in the literature dealing with the backstepping-PID gains similarity. We note that in order to get a clear comparison and arrive at a better interpretation of the results, the notations of the original formulations were mathematically manipulated to show that the various efforts lead to similar relationships, albeit with minor differences. Note that in the PID gains column, the gains k_i , $i = \{1, 2, 3\}$ refer to the Lyapunov function gains, the gains c_e and c_i relate to the integral action within the control law, and the other parameters relate to the system model.

In this work, the similarity formulation to the PID gains is simpler in the sense that it is established based on three parameters only as shown in Table 4.1, whereas the integral backstepping representations in [96] use four parameters (k_1, k_2, c_e , and c_i) and two additional model-dependent constants (a_0 and a_1) in [94], respectively. Furthermore, the number of dependencies between the adaptive backstepping gains and the PID gains in this work is also minimal, where we use only one multiplication and one summation as compared to [56, 94], which use up to two multiplications and two summations. Therefore, an advantage of the formulation presented in this work over other representations lies in the fact that

Table 4.1: Backstepping-PID relationships in the literature

Reference	Control method	System type	Control law	PID gains
[56]	Backstepping with integral action	Minimal phase linear system: $G(s) = \frac{Y(s)}{U(s)} = \frac{B(s)}{A(s)}$ $= \frac{s+b_0}{s^2+a_1s+a_0}$	$U(s) = \frac{k_D s + k_I / s + k_P}{B(s)} E(s) + \frac{A(s)}{B(s)} Y_d(s)$	$k_P = k_1 k_2 + k_3(k_1 + k_2) + 2 - a_0,$ $k_I = k_1 k_2 k_3 + k_1 + k_3,$ $k_D = k_1 + k_2 + k_3 - a_1.$
[94]	Backstepping	Second-order system: $\ddot{x} = b_0 u(t) - a_1 \dot{x} - a_0 x$	$u = \frac{1}{b_0} (-k_P e - k_I \int edt - k_D \dot{e} + a_0 x_d + a_1 \dot{x}_d + \ddot{x}_d)$	$k_P = k_1 k_2 + \frac{c_i}{c_e} (k_1 + k_2) + c_e^2 - a_0,$ $k_I = k_1 k_2 \frac{c_i}{c_e} + c_e c_i,$ $k_D = k_1 + k_2 + \frac{c_i}{c_e} - a_1.$
[95]	Integral Backstepping	Nonlinear mechanical systems of degree 2: $\dot{x}_1 = f_1(x_1) + g_1(x_1)x_2$ $\dot{x}_2 = f_2(\bar{x}) + g_2(\bar{x})u + b$	$u = g_2^{-1} (-k_P e - k_I \int edt - k_D [x_2 - g_1^{-1}(\dot{x}_{1d} - f_1)] + \alpha_1^{x_1} [f_1 + g_1 x_2] + \alpha_1^t - f_2),$ $\alpha_1 = g_1^{-1} (-c_i \int edt - k_1 e - f_1 + \dot{x}_{1d})$	$k_P = k_2 g_1^{-1} k_1 + g_1^{-1} c_i + g_1^T,$ $k_I = k_2 g_1^{-1} c_i,$ $k_D = k_2.$
[96]	Integral Backstepping	Quadrotor rotational subsystem: $\dot{x}_1 = x_2$ $\dot{x}_2 = f + gu$	$u = g^{-1} (-k_P e - k_I \int edt - k_D \dot{e} + \ddot{y}_d - f)$	$k_P = k_1 k_2 + \frac{c_i}{c_e} (k_1 + k_2) + c_e^2,$ $k_I = k_1 k_2 \frac{c_i}{c_e} + c_e c_i,$ $k_D = k_1 + k_2 + \frac{c_i}{c_e}.$
[97]	Nonlinear Backstepping	Quadrotor rotational subsystem: $\dot{x}_1 = x_2$ $\dot{x}_2 = f + gu$	$u = g^{-1} (-k_P e - k_I \int edt - k_D \dot{e} + \ddot{x}_{1d})$	$k_P = (k_1 k_2 + k_3) / mg,$ $k_I = (k_1 k_3) / mg,$ $k_D = (m k_1 + k_2) / mg.$
This work	Integral Backstepping	A class of nonlinear systems of degree 2: $\dot{x}_1 = x_2$ $\dot{x}_2 = f(\bar{x}) + \phi(\bar{x})\theta(t) + g(\bar{x}, t)u + \Delta(t)$	$u = g^{-1} (-k_P e - k_I \int edt - k_D \dot{e} + \ddot{x}_{1d} - f - \phi\hat{\theta})$	$k_P = k_1 k_2 + \gamma + 1,$ $k_I = \gamma k_1,$ $k_D = k_1 + k_2.$

it provides a simpler method to design and tune backstepping controllers based on PID laws and tuning rules, which makes it more practical and easier to adopt in real-life applications.

4.4.3 Implementation of the Backstepping-PID Gains Relationship

An implementation of (4.20) for four preset values of k_I ($k_I=0,1,2,4$) is shown in Fig. 4.2, where Fig. 4.2(a) on the left shows the backstepping gains, k_1 and k_2 , as a function of k_P with a constant k_D condition, and Fig. 4.2(b) on the right shows the gains as a function of k_D with a constant k_P condition. For every streamline marking a constant k_D or k_P , the black asterisks mark the intersections of the k_1 and k_2 curves as applicable, which correspond to the maximum k_P value in Fig. 4.2(a) and the minimum k_D value in Fig. 4.2(b). Note that γ is not plotted on the same figure in order to maintain a clear and legible representation of

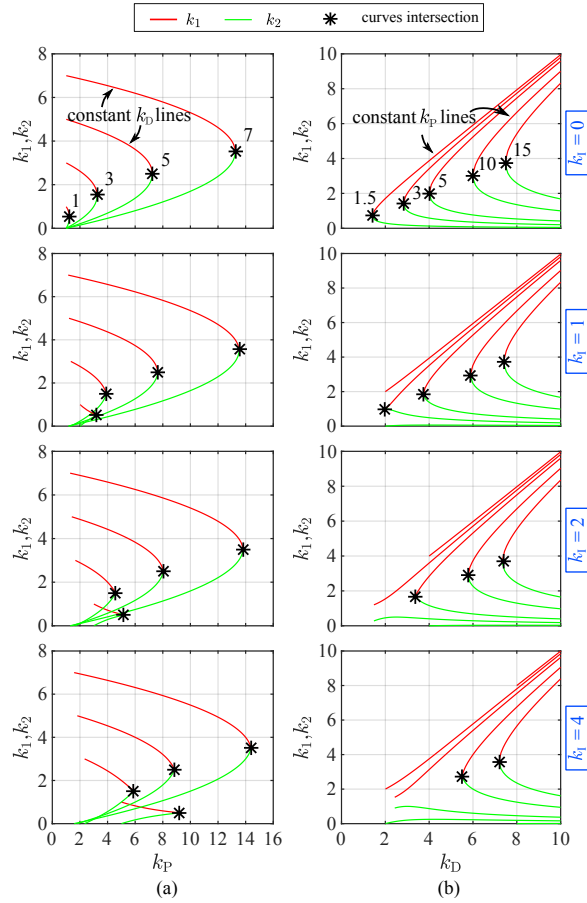


Figure 4.2: Coupled backstepping gains, k_1 and k_2 , as a function of k_P and k_D at constant k_I values ($k_I=0,1,2,4$ corresponding to four rows). The left column in (a) shows the backstepping gains at constant k_D and k_I , and the right column in (b) shows the backstepping gains at constant k_P and k_I .

the analysis results, noting that it can be directly calculated from (4.20) and is perceived as a scaled version of the inverse of k_1 for each corresponding plot.

The significance of the above results lies in making the design and tuning process of the backstepping-based ICS gains easier by allowing the application of well-established PID tuning rules. Additionally, by excluding the unfeasible PID gains that result in non-positive or non-real k_1 and k_2 roots, the tuning interval is tightened. Furthermore, the formulation provides insight into both the traditional stability perspective as well as the performance perspective of the adaptive backstepping control law.

Tuning each one of the gains k_1 and k_2 has a dual effect on both k_P and k_D , as illustrated in Fig. 4.2, yet it does not influence the system performance directly as typically expected from tuning PID gains, which can make the tuning process cumbersome and not straightforward. For instance, increasing any of the backstepping gains k_1 and k_2 will also result in an increase in both k_P and k_D simultaneously, and similarly, tuning γ affects both k_P and k_I .

Remark 13. *To provide a more detailed analysis, we introduce the following notes to guide the tuning process based on the results obtained in Fig. 4.2:*

- *For a given value of k_D , the k_P value that attains the desired performance and stability margin can be obtained from Fig. 4.2(a).*
- *For a given value of k_P , the k_D value that attains the desired performance and stability margin can be obtained from Fig. 4.2(b).*
- *As the value of k_1 increases, the shape of the solutions starts to deviate from the uniform parabola and hyperbola contours.*
- *Since k_1 and k_2 must be positive per Lyapunov stability requirements, a streamline stops when one of its two branches reaches the abscissa.*
- *The dual point (asterisk) can be the optimal solution in terms of stability, for some preset values of k_P or k_D .*
- *k_D linearly relates to the sum of k_1 and k_2 , which manifests in Fig. 4.2(b) as all solution sets (k_1 and k_2) are below the identity line (unity slope), and their sum is equal to it ($k_1 + k_2 = 1$). As such, one can promptly compute the value of k_2 in Fig. 4.2(b) for any given k_1 based on (4.20).*
- *While other solutions besides the ones presented in Fig. 4.2 may exist (as per Fig. 4.1), they are disregarded due to their irrelevance to this work.*

4.4.4 Simplified Backstepping-PID Relationships

If (4.20) has a weak dependence on γ , then by referring to (4.19) and (4.20), the following inequality holds:

$$k_1 k_2 + 1 \gg k_I/k_1, \quad (4.23)$$

and (4.20) can be simplified into the following second-order form:

$$\begin{aligned} k_{1,2}^2 + (-k_D)k_{1,2} + (k_P - \gamma - 1) &= 0, \\ k_1 &\cong \max\{k_{1,2}\}, \\ k_2 &\cong \min\{k_{1,2}\}, \\ \gamma &\cong k_I/k_1, \end{aligned} \quad (4.24)$$

such that: $k_{1,2} > 0$.

As a rule of thumb, the condition in (4.23) holds if:

$$k_P > 10 k_I/k_D.$$

Hence, the general solution of the second order polynomial, $k_{1,2}$, is given by:

$$k_{1,2} = \frac{k_D}{2} \pm \sqrt{\left(\frac{k_D}{2}\right)^2 - (k_P - \gamma - 1)}. \quad (4.25)$$

Note that k_1 and k_2 are directly obtained from solving the polynomial in (4.24), unlike in (4.20) where only k_1 is obtained. To further analyze (4.24) and determine the behavior of $k_{1,2}$ when k_P and k_D vary, the following analysis based on conic representation is introduced.

The general second-order equation of conics is of the form:

$$\kappa_1 x_1^2 + \kappa_2 x_1 x_2 + \kappa_3 x_2^2 + \kappa_4 x_1 + \kappa_5 x_2 + \kappa_6 = 0. \quad (4.26)$$

where x_1 and $x_2 \in \mathbb{R}$ are the variables of the equation, and $\kappa_i \in \mathbb{R}$, $i = \{1, \dots, 6\}$ are constants [104]. If $\kappa_2 \neq 0$, then the principle axes of the conic incur a counterclockwise rotation ϑ with respect to x_1 , such that:

$$\cot 2\vartheta = \frac{\kappa_1 - \kappa_3}{\kappa_2}. \quad (4.27)$$

Conic Representation with Constant k_D

If (4.24) is solved for specific k_D values, as in Fig. 4.2(a), we obtain:

$$\begin{aligned} x_1 &:= k_{1,2}, & x_2 &:= k_P - \gamma, & \kappa_1 &:= 1, & \kappa_2 &:= 0, \\ \kappa_3 &:= 0, & \kappa_4 &:= -k_D, & \kappa_5 &:= 1, & \kappa_6 &:= -1. \end{aligned}$$

Noticing that $\kappa_1 \neq 0$ and $\kappa_1\kappa_3 = 0$, we conclude that (4.24) becomes a horizontal parabola that relates $k_{1,2}$ to k_P in the form:

$$\begin{aligned} x_1 - \left(\left(\frac{k_D}{2} \right)^2 + 1 \right) &= - \left(x_2 - \frac{k_D}{2} \right)^2, \\ V_P &= \left\{ \left(\frac{k_D}{2} \right)^2 + 1, \frac{k_D}{2} \right\}, \end{aligned} \quad (4.28)$$

where V_P is the parabola's vertex.

Conic Representation with Constant k_P

Similarly, (4.24) can be solved for preset k_P values, as in Fig. 4.2(b), to relate $k_{1,2}$ to k_D , that is:

$$\begin{aligned} x_1 &:= k_{1,2}, \quad x_2 := k_D, \quad \kappa_1 := 1, \quad \kappa_2 := -1, \\ \kappa_3 &:= 0, \quad \kappa_4 := 0, \quad \kappa_5 := 0, \quad \kappa_6 := k_P - \gamma - 1. \end{aligned}$$

Noticing that $\kappa_2 \neq 0$, the rotation angle of the conic axes is calculated from (4.27) to be $\vartheta_h = \frac{\pi}{8}$. We can then substitute $x_1 = x'_1 c_{\vartheta_h} - x'_2 s_{\vartheta_h}$ and $x_2 = x'_1 s_{\vartheta_h} + x'_2 c_{\vartheta_h}$ in (4.24), where x'_1 and x'_2 are the principle axes of the rotated conic, to obtain:

$$\begin{aligned} \frac{x_1'^2}{a^2} - \frac{x_2'^2}{b^2} &= 1, \\ V_D &= \left\{ a c_{\vartheta_h}, a s_{\vartheta_h} \right\}, \end{aligned} \quad (4.29)$$

which represent a rotated horizontal hyperbola, with V_D being the hyperbola's vertex, $a = \sqrt{\frac{2(k_P - \gamma - 1)}{\sqrt{2} - 1}}$, and $b = \sqrt{\frac{2(k_P - \gamma - 1)}{\sqrt{2} + 1}}$.

However, the function of $k_{1,2}$ with respect to k_D is interpreted easier when it has the same form as (4.24), since we seek solutions for a single value of k_D one at a time. The double roots occur at V'_D such that:

$$V'_D = \left\{ 2\sqrt{k_P - \gamma - 1}, \sqrt{k_P - \gamma - 1} \right\}. \quad (4.30)$$

Interpretation

The resultant plots of the parabola and hyperbola formulations in (4.28) and (4.29) are shown in Fig. 4.3, which can be used to promptly and clearly interpret the relationship between the backstepping and the PID gains, while mainly benefiting from the vertices' locations. For instance, starting from an initial combination of k_1 and k_2 , one can easily find the corresponding k_P , k_D , and k_I gains using Fig. 4.3; at the same time, one can navigate along the curves to arrive at the desired PID gains while keeping an eye on the resulting k_1 and k_2 gains.

Table 4.2: PD Gains Tuning Intervals and Effect on Backstepping Gains

	$(k_P - \gamma)_{\min}$	$(k_P - \gamma)_{\max}$	$k_{D,\min}$	$k_{D,\max}$	
Constant k_D	1	$(\frac{k_D}{2})^2 + 1$	-	-	\rightarrow
Constant k_P	-	-	$2\sqrt{k_P - \gamma - 1}$	∞	\rightarrow

	$k_{1,\min}$	$k_{1,\max}$	$k_{2,\min}$	$k_{2,\max}$
Constant k_D	$\frac{k_D}{2}$	k_D	0	$\frac{k_D}{2}$
Constant k_P	$\sqrt{k_P - \gamma - 1}$	∞	0	$\sqrt{k_P - \gamma - 1}$

Note that the feasible k_1 and k_2 solutions are marked in red and green colors, respectively, while the unfeasible region is marked in grey.

To maximize the benefits of Fig. 4.3, we provide an explicit interpretation of its results:

- For some preset k_D , the maximum $(k_P - \gamma)$ value can be deduced from (4.28) as:

$$(k_P - \gamma)_{\max} = \left(\frac{k_D}{2}\right)^2 + 1, \quad (4.31)$$

which corresponds to the last real solution (two identical roots) of (4.28):

$$k_{1,2} = \frac{k_D}{2}.$$

- For some preset $(k_P - \gamma)$, the minimum k_D value can be approximated from (4.30) as:

$$k_{D,\min} = 2\sqrt{k_P - \gamma - 1}, \quad (4.32)$$

which corresponds to the first real solution (two identical roots) of (4.24):

$$k_{1,2} = \sqrt{k_P - \gamma - 1}.$$

- If it is desired to increase k_P while maintaining a constant k_D , k_1 should be decreased while k_2 should be increased.
- If it is desired to increase k_D while maintaining a constant k_P , k_1 should be increased while k_2 should be decreased.

A summary of stable tuning intervals of the PID gains as well as the backstepping gains intervals is presented in Table 4.2, which serves as a practical guide for the tuning process concerning the gains and stability bounds. In addition, a summary of the influence of tuning the PID gains on the backstepping gains, as well as their effect on the system performance is presented in Table 4.3.

Table 4.3: Effects of Tuning PID Gains [2] and their Influence on the Backstepping Gains

Gain variation	k_1	k_2	γ	Speed	Stability	Overshoot
k_P increases	-	+	+	+	-	+
k_I increases	-	+	+	-	+	+
k_D increases	+	-	-	+	+	-

Remark 14. *Unlike the PID gains, a single backstepping gain variation, i.e. k_1 , k_2 , or γ , does not result in a unified effect on the system performance as per Table 4.3, which can be verified by referring to (4.18) and (4.19).*

By combining the results of Fig. 4.3 and Table 4.3, and starting from an initial system performance, we can specify the desired performance modification, i.e. speed, stability, or overshoot, then decide on the PID gains tuning direction based on Table 4.3, then by referring to Fig. 4.3, the effect on the backstepping gains can be directly assessed.

Given the above discussion, and by perceiving the seemingly complex control law in (4.11) as a PID feedback plus feedforward terms as in (4.17), the ambiguity can be clarified by a direct visual interpretation that can be obtained from Fig. 4.3, thus reducing any complexity concerns that are typically associated with adopting nonlinear control laws.

4.4.5 From Backstepping to Pole Placement

After substituting (4.17) and (4.19) in (4.8), the resulting error dynamics of the system become:

$$E_1(s^3 + s^2(k_1 + k_2) + s(k_1k_2 + \gamma + 1) + (\gamma k_1)) = D, \quad (4.33)$$

where s is the Laplace operator, and E_1 and D are the Laplace transforms of e_1 and Δ , respectively. In the PID representation, (4.33) becomes:

$$E_1(s^3 + s^2k_D + sk_P + k_I) = D. \quad (4.34)$$

The error dynamics are expressed in the poles form to show the relationship between the PID gains and the locations of the closed-loop poles in the s -plane:

$$\begin{aligned} E_1((s + p_1)(s + p_2)(s + p_3)) &= D, \\ E_1\left(s^3 + s^2(p_1 + p_2 + p_3) + s(p_1p_2 + p_3(p_1 + p_2)) \right. \\ &\quad \left. + (p_1p_2p_3)\right) = D, \end{aligned} \quad (4.35)$$

where p_1 and $p_2 \in \mathbb{C}$, and $p_3 \in \mathbb{R}_{>0}$ are the roots of (4.35), such that they always have positive real parts $\in \mathbb{R}_{>0}$ for a stable system performance. With a basic analysis that compares (4.34) to (4.35), the required PID gains for a specific roots location in the s -plane are obtained as follows:

$$\begin{aligned} k_P &= p_1 p_2 + p_3(p_1 + p_2), \\ k_I &= p_1 p_2 p_3, \\ k_D &= p_1 + p_2 + p_3. \end{aligned} \tag{4.36}$$

The resulting PID gains can then be substituted in (4.18) and (4.20) to obtain the equivalent backstepping gains, if feasible. Furthermore, a visual relationship can be established between the system error dynamics roots to the backstepping gains as follows:

$$\begin{aligned} k_1 k_2 + \gamma + 1 &\equiv p_1 p_2 + p_3(p_1 + p_2), \\ \gamma k_1 &\equiv p_1 p_2 p_3, \\ k_1 + k_2 &\equiv p_1 + p_2 + p_3. \end{aligned} \tag{4.37}$$

Note that although a simple analytical solution cannot be attained for this problem, some similarity still exists between the backstepping gains and the error dynamics roots. For instance, k_P and k_I include multiplication of the backstepping gains, and k_D includes summation of the gains. In addition, one can notice the stark similarity captured in Table 4.1 between the formulations in [56,94] and that in [96], relative to the s -plane poles in (4.36). This observation stipulates that the backstepping gains have a close manifestation as the system closed-loop poles, when specifically formulated to lend themselves for such analysis.

To provide a clear visual aid that illustrates the above compelling finding, Fig. 4.4 shows an implementation of the system of equations in (4.36) for the case when $k_I = 0$, such that the root locus of the error dynamics is drawn in Fig. 4.4(a) for some constant k_D values, and in Fig. 4.4(b) for some constant k_P values. Note that the colored (red, green, and blue) asterisks and stars in Fig. 4.4 correspond to the exact same gains of their counterparts in Fig. 4.3. For instance, the solution sets at the stars that mark distant values of k_1 and k_2 in Fig. 4.3 correspond to more-damped closed-loop system poles, but with one pole being closer to the imaginary axis in most cases as per Fig. 4.4. Also, the solution sets at the asterisks that mark dual (equal) values of k_1 and k_2 in Fig. 4.3 correspond to less-damped closed-loop system poles, with their real-parts lying in between the real-part of the sets marked by the stars as per Fig. 4.4. Hence, it can be stated that moving towards the dual solution of k_1 and k_2 in Fig.4.3 (at the asterisks) during the gains tuning process tends to reduce the damping ratio of the resultant s -plane poles.

The unique combination of Fig. 4.3 and Fig. 4.4 allows for understanding and visually expressing the relationship between backstepping gains, the PID gains, and the poles of the closed-loop system.

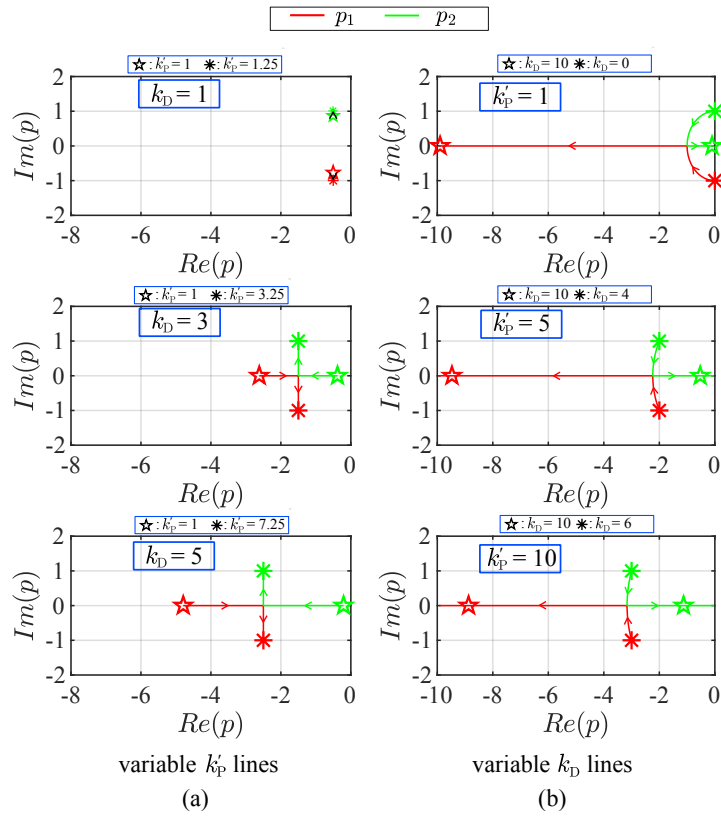


Figure 4.4: Coupled pole locations, p_1 and p_2 , for specific k_p and k_D values at constant k_D in (a) on the left, and at constant $k'_p = k_p - \gamma$ in (b) on the right. The stars and asterisks mark the same set of the corresponding gains in Fig. 4.3.

Algorithm 2 Backstepping Gains Tuning Using PID Formulation

Input: k_P, k_I, k_D **Output:** k_1, k_2, γ *Initialization :*Calculate k_1, k_2, γ from (4.20)**while** *stability and performance criteria not satisfied* **do**

- 1: **if** $(k_1 k_2 + 1 \gg k_I/k_1)$ **then**
 - 2: Tune $(k_P - \gamma)$ using Table 4.3 and Fig. 4.3(c)
 - 3: Tune k_D using Table 4.3 and Fig. 4.3(d)
 - 4: Tune k_I using Table 4.3
 - 5: Find the equivalent k_1 and k_2 from Fig. 4.3 or (4.24)
 - 6: Calculate γ from (4.24)
 - 7: **else**
 - 8: Tune k_P using Table 4.3 and Fig. 4.2(a)
 - 9: Tune k_D using Table 4.3 and Fig. 4.2(b)
 - 10: Tune k_I using Table 4.3 and Fig. 4.2
 - 11: Find the equivalent k_1, k_2 and γ from Fig. 4.2 and (4.20)
 - 12: **end if**
 - 13: **return** k_1, k_2, γ
-

4.4.6 Tuning Procedure

After deriving the necessary equations, expressing them in the proper form, and providing visual aids and practical implementation tips, we consolidate the work in an algorithm that can be used to tune the gains of a backstepping-based controller based on the findings of this paper, which is presented in Algorithm 2.

Note that if a primitive PID controller is designed as a baseline to get a first assessment of the system performance, the obtained PID gains can be plugged into Algorithm 2 for initialization. It is also possible to calculate the set of PID gains that satisfy the design specifications given by the pole locations in (4.36). Finally, the fine-tuning of the gains can be performed to strike a balance between the performance or stability of the system by referring to Fig. 4.2 and Fig. 4.3.

4.5 Simulation and Experimental Validation

Among various possible applications, a quadrotor UAV platform with variable payload is adopted to validate the hereby proposed ICS, where a motion control system is required to achieve good tracking performance of desired trajectories in the presence of disturbances and changing operating conditions. Collectively achieving the aforementioned objectives is faced with several challenges including

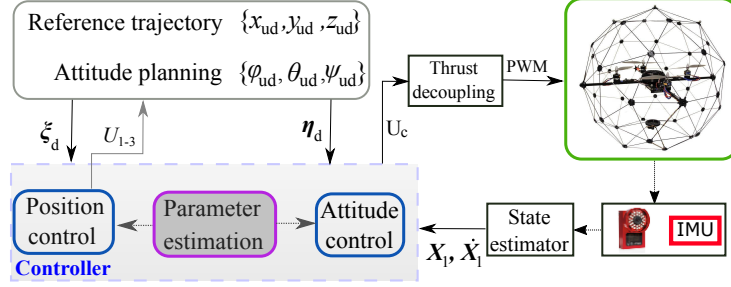


Figure 4.5: Control system architecture for a quadrotor UAV.

design complexity, internal and external disturbances, high frequency noise, and applicability to various quadrotor platforms. To deal with the challenges while achieving the performance objectives, the controller structure shown in Fig. 4.5 is adopted, which features an inner-loop for attitude control and an outer-loop for position control, thus the proposed ICS has the advantage of reducing the complexity of the design and tuning process.

Nonetheless, the main purpose of this validation section is to show the tuning process for a complex robotic system using the methodology presented in this work, and not necessarily to show the superiority of the control law. The proposed algorithm is validated via numerical simulations on a quadrotor UAV's dynamic model of high-fidelity, and experimentally on the Quanser QBall-2 quadrotor platform [105].

4.5.1 Dynamic Model

Consider the system illustrated in Fig. 4.6, where the inertial and body frames of reference are \mathcal{W} and \mathcal{B} , respectively. The latter is fixed to the quadrotor's centroid, \mathcal{O}_B . The quadrotor's mass and inertia tensor relative to \mathcal{O}_B are respectively $m_u \in \mathbb{R}_{>0}$ and $\mathbf{J}_u \in \mathbb{R}^{3 \times 3}$. The quadrotor's center of mass (CoM) coordinates in \mathcal{B} are denoted by $\mathbf{r}_{\text{com}} = [x_{\text{com}}, y_{\text{com}}, z_{\text{com}}]^\top \in \mathbb{R}^3$, its translational position in \mathcal{W} is $\boldsymbol{\xi} = [x_u, y_u, z_u]^\top$, and its Euler angles vector for roll, pitch, and yaw motions is $\boldsymbol{\eta} = [\phi_u, \theta_u, \psi_u]^\top$.

The motor and propeller dynamics are superficially modelled as a reduced first-order transfer function, but they are not included in the controller design process due to their relatively fast dynamics as compared to those of the quadrotor's chassis. The quadrotor's dynamics are represented by the following model:

$$\begin{aligned} m_u \ddot{\boldsymbol{\xi}} &= -m_u \mathbf{G} + \mathbf{R}_t \mathbf{F}_B + \Delta_\xi, \\ \mathbf{J}_u \mathbf{R}_r \ddot{\boldsymbol{\eta}} &= \boldsymbol{\tau}_B + \mathbf{r}_{\text{com}} \times \mathbf{F}_B + \Delta_\eta, \end{aligned} \quad (4.38)$$

where $\mathbf{R}_{1,u}$ and $\mathbf{R}_{2,u}$ are the translational and rotational transformation matrices between \mathcal{W} and \mathcal{B} [106]; $\mathbf{G} = [0, 0, g]^\top$ is the gravitational acceleration vector and g is the gravity constant; $\mathbf{F}_B = [0, 0, F_t]^\top$ and $\boldsymbol{\tau}_B = [T_1, T_2, T_3]^\top \in \mathbb{R}^3$ are the total

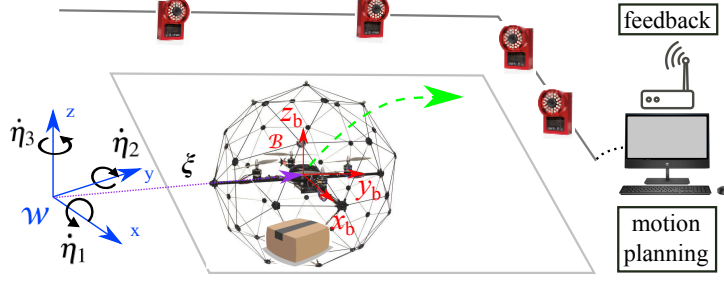


Figure 4.6: Quadrotor system setup with an indoor motion capture system.

force and torque vectors of the rotors in \mathcal{B} ; the relationship between the motor speeds and the net thrust and torques is defined as $\mathbf{U} = \mathbf{\Gamma}\mathbf{\Omega}^2$, and explicitly expressed as:

$$\begin{bmatrix} F_t \\ T_1 \\ T_2 \\ T_3 \end{bmatrix} = \begin{bmatrix} K_t & K_t & K_t & K_t \\ 0 & 0 & dK_t & -K_t L \\ K_t L & -K_t L & 0 & 0 \\ K_Q & K_Q & -K_Q & -K_Q \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix}, \quad (4.39)$$

where K_t and K_Q are respectively the rotor constants of thrust and torque, L is the quadrotor's arm length, ω_i is the angular speed of the i^{th} rotor with $i \in [1, 2, 3, 4]$, and $\mathbf{\Omega}$ is the vector of the four rotors' angular speeds. The control inputs, ω_i , are calculated from the above equations using matrix inversion and square root operations.

The terms, Δ_ξ and $\Delta_\eta \in \mathbb{R}^3$, respectively denote the external disturbances and unmodeled nonlinearities (not explicitly shown for brevity purposes), and they are assumed to be uniformly bounded. For more details on the quadrotor modeling, readers are referred to [68, 106].

4.5.2 Problem Formulation

The dynamics of the quadrotor are expressed in the form of (4.2) by choosing:

$$\begin{aligned} \mathbf{X}_1 &= [x_u, y_u, z_u, \phi_u, \theta_u, \psi_u]^\top, \quad \mathbf{X}_2 = [\dot{x}_u, \dot{y}_u, \dot{z}_u, \dot{\phi}_u, \dot{\theta}_u, \dot{\psi}_u]^\top, \\ \mathbf{g} &= [m_u^{-1} \mathbf{I}_3, \mathbf{O}_3; \mathbf{O}_3, (\mathbf{J}_u \mathbf{R}_r)^{-1}], \\ \mathbf{H}_u &= [0, 0, -g, 0, 0, 0]^\top, \quad \Phi_u = [\mathbf{O}_{3 \times 2}; \Phi_{u,2}], \\ \Delta_u &= [\Delta_\xi; \Delta_\eta], \quad \mathbf{U} = [\mathbf{F}_W; \boldsymbol{\tau}_B], \quad \Theta_u = [x_{\text{com}}, y_{\text{com}}]^\top, \end{aligned} \quad (4.40)$$

where the vectors $\boldsymbol{\theta}$, \mathbf{g} , and Δ are the unknown entities in the system, and $\mathbf{O}_3 \in \mathbb{R}^{3 \times 3}$ is the null matrix. Acrobatic maneuvers are not considered in this work, and the singular point of $\psi_u = \frac{\pm\pi}{2}$ is not to be reached, which implies that

\mathbf{g} remains Lipschitz continuous. The matrix $\phi_2 \in \mathbb{R}^{3 \times 2}$ is defined as:

$$\Phi_{u,2} = \begin{bmatrix} 0 & -F_t c_{\theta_u} / J_{u,xx,0} \\ F_t / J_{u,yy,0} & -F_t t_{\phi_u} s_{\theta_u} / J_{u,xx,0} \\ 0 & F_t s_{\theta_u} / (c_{\phi_u} J_{u,xx,0}) \end{bmatrix},$$

where $J_{u,xx,0}$ and $J_{u,yy,0}$ are the nominal values of $J_{u,xx}$ and $J_{u,yy}$, respectively. Choosing these nominal values is realistic and practical since the quadrotor's inertia variation is typically a small fraction of its nominal value, which simplifies the formulation of the adaptation law. The control input vector for the translational subsystem is calculated as follows:

$$\mathbf{F}_W = \begin{bmatrix} s_{\theta_u} c_{\psi_u} + s_{\phi_u} c_{\theta_u} s_{\psi_u} \\ s_{\theta_u} s_{\psi_u} - s_{\phi_u} c_{\theta_u} c_{\psi_u} \\ c_{\phi_u} c_{\theta_u} \end{bmatrix} F_t. \quad (4.41)$$

4.5.3 Control Inputs and Reference Signals

The aim of the controller is to achieve trajectory tracking by bringing the state vector, \mathbf{X}_1 , to a desired state vector, $\mathbf{X}_{1d} = [\boldsymbol{\xi}_d; \boldsymbol{\eta}_d] = [x_{u,d}, y_{u,d}, z_{u,d}, \phi_{u,d}, \theta_{u,d}, \psi_{u,d}]^\top$, by generating an appropriate control input vector, $\mathbf{U}_c = [F_{tc}, T_{1c}, T_{2c}, T_{3c}]^\top$, denoted by the subscript $(\bullet)_c$. The motor's rotational speed can be modelled as:

$$\omega_i = \mathcal{G}_m(s) \omega_{i,c}, \quad i \in [1, 2, 3, 4], \quad (4.42)$$

where $\mathcal{G}_m(s)$ is the motor's transfer function, and $\omega_{i,c}$ is the command rotational speed that is proportional to the input pulse width modulated (PWM) signal of each motor. $\omega_{i,c}$ can be obtained from the inverse of the following equation: $\mathbf{U} = \Gamma \boldsymbol{\Omega}_c^2$. In practice, the motor dynamics are much faster than those of the quadrotor, thus $\mathcal{G}_m(s)$ can be set to one.

By dividing the quadrotor's motion control system into two subsystems, namely an inner-loop for attitude control (Euler angles) and an outer-loop for position control, the complexity of the problem is considerably reduced. The position controller generates the control signal, $[\mathbf{u}_x, \mathbf{u}_y, \mathbf{u}_z]$, from which the total thrust command is calculated as:

$$F_{tc} = \sqrt{\mathbf{u}_x^2 + \mathbf{u}_y^2 + \mathbf{u}_z^2}, \quad (4.43)$$

and the desired attitude angles, $\phi_{u,d}$ and $\theta_{u,d}$, are calculated from (4.41) as follows:

$$\begin{aligned} \phi_{u,d} &= \arctan((\mathbf{u}_x s_{\psi_u} - \mathbf{u}_y c_{\psi_u}) / \mathbf{u}_z), \\ \theta_{u,d} &= \arcsin((\mathbf{u}_x c_{\psi_u} + \mathbf{u}_y s_{\psi_u}) / F_{tc}). \end{aligned} \quad (4.44)$$

The inner-loop commands in (4.44) are smoothed and bounded by the function:

$$\begin{aligned}\phi'_{u,d} &= \phi_{u,\max} \tanh(\phi_{u,d}/\phi_{u,\max}), \\ \theta'_{u,d} &= \theta_{u,\max} \tanh(\theta_{u,d}/\theta_{u,\max}),\end{aligned}\tag{4.45}$$

where $\phi_{u,\max}$ and $\theta_{u,\max} \in (0, \frac{\pi}{2})$ are the attitude angles' upper limit. As a result, \mathbf{X}_{1d} is smooth and bounded. This completes the definition of the desired state vector, \mathbf{X}_{1d} , to be tracked by \mathbf{X}_1 .

Note that the control law in (4.17) and its modification in (4.19) were implemented with the projection function being realised as an integrator, with upper and lower saturation bounds calculated as $\bar{d}/(\gamma k_1)$ based on Remark 8.

4.5.4 Simulation Model Elements

Simulation Setup

The MATLAB/Simulink[®] environment is used for validating the proposed control design and tuning scheme, where a high-fidelity nonlinear dynamic model powers the flight mission. The model is augmented to include feedback delays, force and torque disturbances, and induced sensor noise. Estimates of the actual measurement signals are used in the feedback process, where a Kalman filter generates smooth signals from raw sensory data, and RLS algorithm provides estimates of the parameters m_u , $J_{u,xx}$, $J_{u,yy}$, $J_{u,zz}$, x_{com} , and y_{com} . An external wind gust is generated via the method described in [107] to disturb the quadrotor with a magnitude of 5 m/s, which results in a drag force that acts on the quadrotor's centroid.

Quadrotor Specifications

The Quanser QBall-2 platform [105] is used as a baseline for the quadrotor's specifications. We set: $m_u = 1.76$ kg, $J_{u,xx} = J_{u,yy} = 0.03$ kg m², $J_{u,zz} = 0.04$ kg m², $K_t = 13$ N, $K_Q = 0.4$ N m, and $L = 0.2$ m. Upon adding a payload, the quadrotor's CoM and moment of inertia (MoI) vary as per [108], with their main effect manifesting in the moment arm shift due to pure thrust as expressed in (4.40) by the term $(\mathbf{r}_{\text{com}} \times \mathbf{F}_{\mathcal{B}})$. The thrust and torques feedback signals are estimated via a battery drain model due to the unavailability of a motor speed sensor [109].

4.5.5 Testing Scenario

The controller testing scenario consists of the following actions: the quadrotor first takes-off from the ground while picking up a payload, it executes an ∞ -shaped maneuver three times, and then performs landing on top of its starting position. The payload has a mass $m_p = 0.2$ kg and is placed beneath the original CoM of the quadrotor at coordinates $\mathbf{r}_p = [x_p, y_p, z_p] = [0.05, 0.05, -0.1]$ m in \mathcal{B} .

We set $J_{xx,u,0} = J_{xx,u}$, and $J_{yy,u,0} = J_{u,yy}$, and a wind gust of magnitude 5 m/s is activated at $t = 11$ s.

4.5.6 Tuning Process

The tuning process described in Section 4.4.6 is performed for the outer loop's lateral motion subsystems (x_u and y_u components) to serve as a guideline for utilizing the proposed tuning method. Tuning is based on the PID gains, since it is intended in this work to move away from the abstract backstepping gains, and the process is phased as follows:

1. The x_u and y_u subsystems' gains are initially set to $k_{P,x,y} = 1.1$, $k_{D,x,y} = 1.1$, and $k_{I,x,y} = 0$. The results are shown in Fig. 4.7(a), where a sluggish response with undesirable overshoot and steady-state error are obtained.
2. We attempt to improve the overshoot performance and increase the tuning range of $k_{P,x,y}$, as we notice from Fig. 4.3(c) that for $k_D \approx 1$, k_P is limited to a small set near unity. The specified objectives are attained by increasing $k_{D,x,y}$ as per Table 4.3. We choose $k_{D,x,y} = 3$. The results are shown in Fig. 4.7(b), where we notice a significant reduction in the tracking error and overshoot. But the system's transient response is still slow in the initial time interval $t = [10, 19]$ s.
3. Next, we attempt to increase the speed of the system, which by referring to Table 4.3 can be achieved by increasing $k_{P,x,y}$. By referring to Fig. 4.3(c), we choose $k_{P,x,y} = 3$. The results are shown in Fig. 4.7(c), where we indeed observe an improved speed of response. But the system still has a steady-state error that is noticed at the time intervals $[0, 10]$ s and $[70, 80]$ s.
4. Next, we increase the integral gain to eliminate the steady-state error, which is due to the initial drift of the quadrotor that is induced by the presence of the eccentric payload. We gradually increase $k_{I,x,y}$ until we find an optimal value, while maintaining $k'_{P,x,y} := k_{P,x,y} - \gamma_{x,y}$ constant to reduce unwanted effects on the previously attained performance. We set $k_{I,x,y} = 4.2$. The results are shown in Fig. 4.7(d), where we notice that the initial drift in position has been promptly compensated for. Now, we can proceed to the final stage of fine-tuning the gains.
5. By inspecting Fig. 4.3(b), we see that the overshoot can be improved at the expense of reducing $k_{2,x,y}$. Thus, we proceed by slightly increasing $k_{D,x,y}$ and setting it to $k_{D,x,y} = 4$. The results are shown in Fig. 4.7(e), where we notice that the overshoot has been reduced (e.g. $t = \{19, 39, 59\}$ s). At this stage, the performance is deemed satisfactory and the tuning process is terminated.

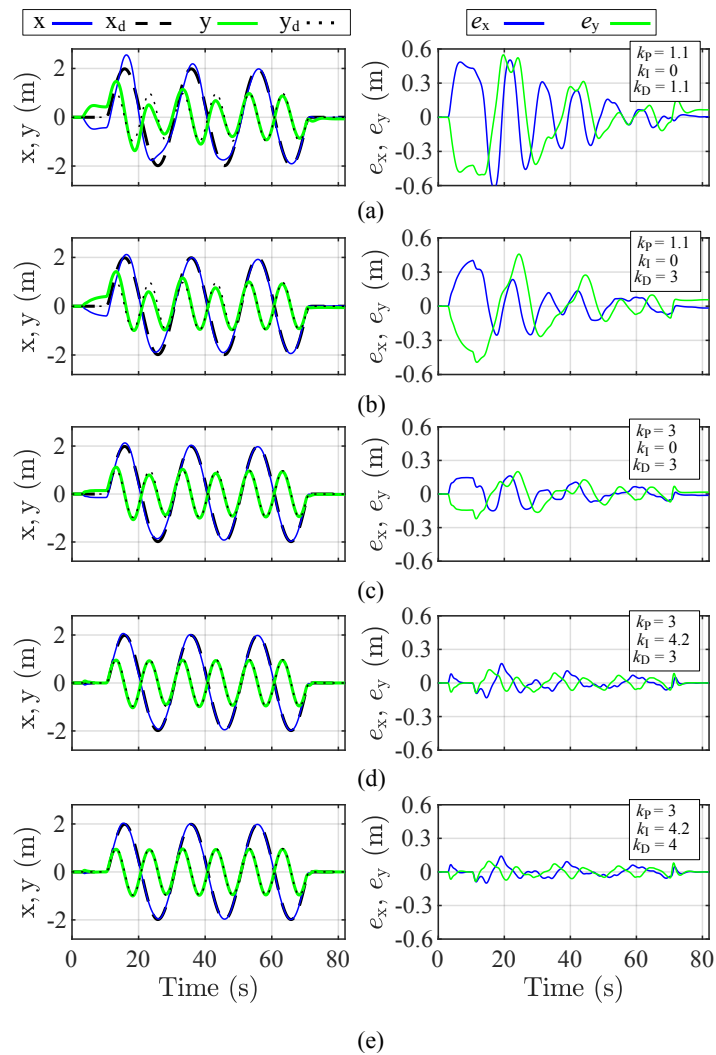


Figure 4.7: Desired trajectory tracking of the x and y position states (left) and their corresponding tracking errors (right) of the ICS-controlled quadrotor in each step of the tuning process executed via numerical simulations.

Table 4.4: Tuning steps for the x and y subsystems of the quadrotor’s adaptive backstepping controller starting from the selected PID gains.

	k_P	k_I	k_D	k_1	k_2	γ	MAV $\{ e_{x_u} + e_{y_u} \}$ (cm)
1.	1.1	0	1.1	1	0.1	0	37
2.	1	0	3	2.97	0.03	0	25
3.	3	0	3	2	1	0	12
4.	3	4.2	3	2.8	0.2	1.5	7
5.	3	4.2	4	3.8	0.2	1.1	6

The tuned PID gains and their corresponding backstepping counterparts, which are obtained in each step of the tuning process, are shown in Table 4.4 along with the respective mean absolute error of the tracking signals (MAE). From the table, it is clearly noticed that predicting the combinations of k_1 , k_2 , and γ that yield the desired performance improvement is not a trivial task and may prove to be very challenging to be determined by control system engineers. This last result shows the power of exploiting the backstepping and PID gains relationship to speed-up the manual tuning process. On the other hand, when a PID controller is employed for controlling a second-order system of the form shown in (4.2), the relationships shown in Fig. 4.3 can serve as an indicator of the stability margins of the system.

4.5.7 Simulation Results

The final controller gains are set to $\mathbf{k}_1 = \text{diag}(3.8, 3.8, 2.6, 30, 30, 4)$, $\mathbf{k}_2 = \text{diag}(0.2, 0.2, 0.4, 0.3, 0.3, 1)$, and $\boldsymbol{\gamma} = \text{diag}(1.1, 1.1, 1, 0.4, 0.4, 1)$ by following the proposed tuning process in conjunction with the performance results obtained in Fig. 4.7. The quadrotor planar trajectory is shown in the ‘ xy ’-subplot of Fig. 4.8 (top), where the UAV makes three laps in an ∞ -shape trajectory in the presence of wind. The quadrotor keeps oriented towards the direction of the trajectory while maintaining a level flight as seen in the ‘heading’ and ‘ z_u ’ subplots of Fig. 4.8, respectively. Finally, the sinusoidal x - and y -signals that produce the ∞ -shape trajectory are shown in the ‘ x_u ’ and ‘ y_u ’ subplots of Fig. 4.8 (bottom right). Based on a mean absolute error $MAV\{[e_{x_u}, e_{y_u}, e_{z_u}]\}$ of [3.0, 2.7, 1.2] cm in position and $MAV\{e_{\psi_u}\} = 1.5^\circ$ in heading angle, the ICS’ overall performance is deemed acceptable.

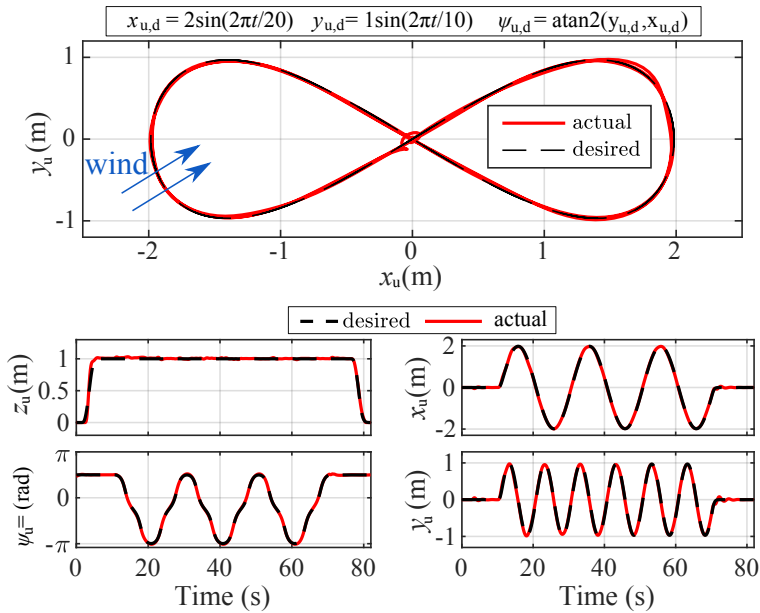


Figure 4.8: Trajectory (top) and tracking states (bottom) of the quadrotor in numerical simulations after fine-tuning the ICS per the hereby proposed Algorithm 2.

4.5.8 Experimental Validation

We first note that since tuning of controllers typically produces gains that yield sub-optimal performance, the design process is started in the simulation environment to prevent physical damage to the quadrotor system. After arriving at satisfactory simulation results, the experimental validation phase kicks off by following the same tuning procedure, which further validates the proposed algorithm. In the experimental validation, the ICS is tested on the Quanser QBall-2 platform [105], with specifications that are similar to those presented in Section 4.5.4. The flight experiment is conducted inside an indoor testing facility, where a motion capture system, depicted in Fig. 4.6, provides feedback for the closed-loop control system.

Experimental Scenario

The quadrotor is loaded with a payload of $m_p = 0.2\text{ kg}$, placed at coordinates $\mathbf{r}_p = [0, 0, -0.1]\text{ m}$ in \mathcal{B} , which is beneath its original CoM as shown in Fig. 4.9. The flight maneuver starts by a vertical take-off, followed by a sideways motion in the x-y plane during which the carried payload is released midway, and ends by landing on the ground, as depicted at the top of Fig. 4.10.

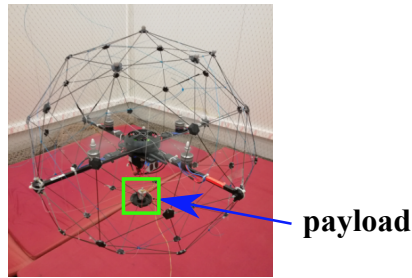


Figure 4.9: Demonstration of the quadrotor platform (Quanser QBall-2) lifting a payload inside the UAV testing laboratory.

Controller Tuning

Given the expected inaccuracies between the simulation model and the physical quadrotor platform, fine-tuning of the obtained gains from the numerical simulations is performed through multiple flight tests using the same proposed tuning guidelines of Algorithm 2. We refrain from showing the detailed tuning step to avoid redundancy. The resulting controller gains that were deemed adequate in terms of performance and robustness are $\mathbf{k}_P = \text{diag}(2.7, 2.7, 7.2, 32.2, 32.2, 6)$, $\mathbf{k}_I = \text{diag}(1.3, 1.3, 4.1, 5.6, 5.6, 5.2)$, and $\mathbf{k}_D = \text{diag}(3.7, 3.7, 4, 16.2, 16.2, 4)$, which give the following backstepping gains: $\mathbf{k}_1 = \text{diag}(3.3, 3.3, 1.96, 14, 14, 2)$, $\mathbf{k}_2 = \text{diag}(0.4, 0.4, 2.1, 2.2, 2.2, 2)$, and $\boldsymbol{\gamma} = \text{diag}(0.4, 0.4, 2.1, 0.4, 0.4, 1)$. Note that the ratio of change of the PID gains, as compared to those in simulation, is smaller than that of the backstepping gains. Since the PID gains better represent the controller's performance, they should be considered as the starting point for evaluating the transition between simulation and experimentation.

Experimental Results

The final tracking performance of the quadrotor is shown at the bottom of Fig. 4.10, which captures its position in the z -, x -, and y -directions. After adequate fine-tuning, the ICS achieves stable and accurate tracking performance, with a mean absolute error $MAV\{[e_{x_u}, e_{y_u}, e_{z_u}]\}$ of $[1.3, 2.7, 1.8]$ cm in position. The recorded error in position-hold is minimal in all three directions, even during lifting the payload off the ground and while dropping it. The quadrotor recovery after the sudden drop of the payload (at $t = 18$ s, marked by the purple strips at the bottom of Fig. 4.10), is fairly quick, and the quadrotor successfully and accurately completes its maneuver. The experimental results demonstrate the efficacy of the adaptive backstepping-based ICS and the practicality of the proposed tuning algorithm based on the established PID similarity.

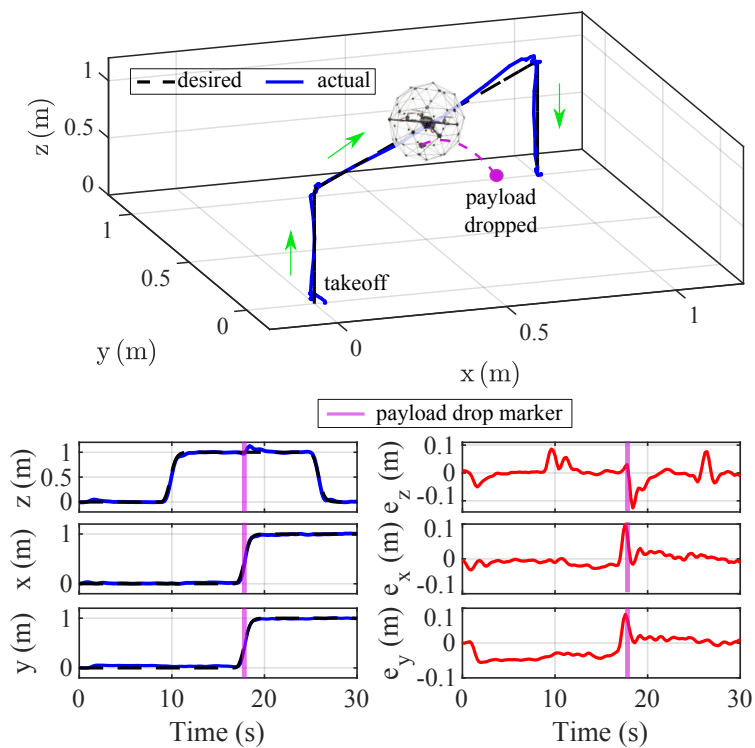


Figure 4.10: 3D trajectory tracking of the ICS-controlled quadrotor in the payload-drop experiment. The experiment is depicted at the top, and the x_u , y_u , and z_u positions and tracking errors are shown at the bottom.

4.6 Conclusion

In this chapter, we leveraged the rather underutilized similarity between backstepping and PID control laws for a class of second-order systems, and developed it to formulate a specific relationship between the backstepping and PID gains in the form of a third-order polynomial, which can be further simplified to second-order under certain conditions. The obtained similarity provides deeper insight into the tuning of backstepping gains and allows for a better understanding of the backstepping control law, which can be represented as the sum of a feedback PID component and a feedforward model compensation component. We provided visual and mathematical tools that facilitate the design and tuning of the backstepping gains in a similar fashion as tuning PID controllers, which benefits from the legacy of well-established PID tuning rules, while maintaining the original stability perspective of the backstepping gains. This work contributes to the field of control systems by promoting the adoption of nonlinear Lyapunov-based integral backstepping controllers, which is achieved by facilitating certain aspects of the control law design and tuning, and bridging the gap between complexity and practicality. The proposed control scheme and tuning algorithm were validated numerically on a quadrotor UAV dynamic model of high-fidelity, and experimentally on a quadrotor platform (Quanser QBall-2), to showcase the ease and utility of the proposed tuning process.

Future work entails the extension of the proposed method to other nonlinear control designs, in addition to backstepping, and to consider higher-order classes of nonlinear systems.

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS

This dissertation presented a novel heterogeneous robotic system including a quadrotor UAV and a floating buoy/USV linked via a tether, named as tethered UAV–buoy system. The system’s multi-physics dynamics were modeled in detail for both the coupled and decoupled cases, along with the design of a surge velocity control system for the simplified 2D planar case, and the full scale 3D case with the addition of the buoy’s steering ability. Detailed analysis was provided on the system’s constraints, attained steady-state conditions, working limits, and operational modes. A state machine was designed to manage the switching between the system’s UAV–buoy relative position control law and the buoy’s surge velocity control law. The system was validated in numerical simulations using a specifically designed open-source MATLAB/Simulink model that is coupled to a Virtual Reality Modelling Language (VRML) simulator, in addition to the development of an open-source ROS-gazebo package that allows the free space validation and manipulation of the proposed robotic system.

In addition, an integrated control and estimation system was introduced with application to quadrotor UAVs to provide robust and adaptive performance, which is suitable for the marine locomotive UAV problem. The integrated control law combines an adaptive backstepping controller design with a normalized recursive least squares estimation algorithm. The resultant backstepping design was formulated as a two-degrees-of-freedom controller: a feedback component in a PID-like structure and a feedforward component that provides model compensation. The ICS design uncovers a special relationship between the backstepping and PID gains in the form of a third-order polynomial, which can be further simplified to second-order under certain conditions. The proposed control and estimation scheme was validated in numerical simulation via a high-fidelity quadrotor model, where the obtained results demonstrated good tracking performance while coping with parametric uncertainties, fast and accurate parameter

estimation, and adequate disturbance rejection. The ICS was also tested experimentally on the Quanser QBall-2 quadrotor platform.

5.1 Research Implications

The proposed tethered UAV–buoy system serves the field of marine robotics by paving the way in front of a wide variety of novel marine applications for multirotor UAVs, where their high speed and maneuverability, as well as their ease of deployment and wide field of vision, give them a superior advantage. It best suits applications that require remote and fast manipulation with minimal water surface disruption.

This dissertation contributes to the field of control systems by promoting the adoption of nonlinear Lyapunov-based integral backstepping controllers, which is achieved by facilitating certain aspects of the control law design and tuning and bridging the gap between complexity and practicality. The obtained similarity presented in Chapter 4 between the backstepping and PID gains provides deeper insight into the tuning of backstepping gains and allows for a better understanding of the backstepping control law, which can be represented as the sum of a feedback PID component and a feedforward model compensation component.

5.2 Recommendations for Future Work

Future work may focus on designing a higher level path planner for the tethered UAV–buoy system to allow tracking trajectories in the inertial frame. Energy minimization techniques can be employed to improve the performance of the path planner under different wave and current conditions. Additionally, the buoy’s stability and shape effects can be studied by introducing additional constraints to the system, and including additional analysis on the effect of the tether on the system. Future work also entails building a prototype of the proposed system and performing experimental validation of the proposed control system. The rope linking mechanism can be also an interesting aspect of future work, where the locomotive UAV can be equipped with a hook holding a sensory package, and once catching the floating buoy can provide the buoy states measurements required to calculate the system’s spherical states without the need for visual sensing from the UAV. Last but not least, future work may include the extension of the proposed ICS algorithm to other nonlinear control design techniques, in addition to backstepping, and to higher-order classes of nonlinear systems.

LIST OF PUBLICATIONS

Journal Articles

A. Kourani and N. Daher, "Marine locomotion: A tethered UAV-buoy system with surge velocity control," *Robotics and Autonomous Systems*, Vol. 145, 2021, 103858, ISSN 0921-8890, <https://doi.org/10.1016/j.robot.2021.103858>.

A. Kourani and Daher. N., "A practical guideline for designing and tuning adaptive backstepping controllers for a class of second-order systems based on PID similarity," *International Journal of Dynamics and Control*, Manuscript ID No. IJDY-D-21-00459R1, (Accepted).

A. Kourani and N. Daher, "Three-dimensional modeling and control of a tethered UAV-buoy system," *Nonlinear Dynamics*, Manuscript ID No. NODY-D-21-02981 (In review).

Conference Papers

A. Kourani and N. Daher, "A tethered quadrotor UAV-buoy system for marine locomotion," *IEEE International Conference on Robotics and Automation (ICRA)*, China, 2021, pp. 59-65.

A. Kourani and N. Daher, "Leveraging PID gain selection towards adaptive backstepping control for a class of second-order systems," *2021 American Control Conference (ACC)*, USA, 2021, pp. 1174-1179.

A. Kourani and Daher. N., "Bidirectional manipulation of a buoy with a tethered quadrotor UAV," *2021 IEEE International Multidisciplinary Conference on Engineering Technology (IMCET)*, Lebanon, 2021, pp. 101-106.

A. Kourani, K. Kassem, and Daher. N., "Coping with quadcopter payload variation via adaptive robust control," *2018 IEEE International Multidisciplinary Conference on Engineering Technology (IMCET)*, Lebanon, 2018, pp. 1-6.

A. Kourani and M. Liermann, "Wave simulation of Lebanese near-shore region with SWAN and unstructured grid," *3rd Asian Wave and Tidal Energy Conference*, Singapore, 2016, pp. 300-303.

APPENDIX A

BUOY INERTIA AND DAMPING MATRICES

The buoy's inertia matrix in the world frame \mathcal{W} , \mathbf{M}_b , is symmetric (i.e., $M_{ij} = M_{ji}$, for $i \neq j$), and has its non-zero elements explicitly given as:

$$\begin{aligned}
M_{b,11} &= m_b + a_{11}(c_{\theta_b}c_{\psi_b} - s_{\phi_b}s_{\theta_b}s_{\psi_b})^2 + a_{22}(c_{\phi_b}s_{\psi_b})^2 + a_{33}(s_{\theta_b}c_{\psi_b} + s_{\phi_b}c_{\theta_b}s_{\psi_b})^2, \\
M_{b,22} &= m_b + a_{11}(c_{\theta_b}s_{\psi_b} + s_{\phi_b}s_{\theta_b}c_{\psi_b})^2 + a_{22}(c_{\phi_b}c_{\psi_b})^2 + a_{33}(s_{\theta_b}s_{\psi_b} - s_{\phi_b}c_{\theta_b}c_{\psi_b})^2, \\
M_{b,33} &= m_b + a_{11}(c_{\phi_b}s_{\theta_b})^2 + a_{22}(s_{\phi_b})^2 + a_{33}(c_{\phi_b}c_{\theta_b})^2, \\
M_{b,12} &= a_{11}(c_{\theta_b}s_{\psi_b} + s_{\phi_b}s_{\theta_b}c_{\psi_b})(c_{\theta_b}c_{\psi_b} - s_{\phi_b}s_{\theta_b}s_{\psi_b}) - a_{22}(c_{\phi_b}^2s_{\psi_b}c_{\psi_b}) \\
&\quad + a_{33}(s_{\theta_b}c_{\psi_b} + s_{\phi_b}c_{\theta_b}s_{\psi_b})(s_{\theta_b}s_{\psi_b} - s_{\phi_b}c_{\theta_b}c_{\psi_b}), \\
M_{b,13} &= -a_{11}(c_{\phi_b}s_{\theta_b})(c_{\theta_b}c_{\psi_b} - s_{\phi_b}s_{\theta_b}s_{\psi_b}) - a_{22}(c_{\phi_b}s_{\psi_b}s_{\phi_b}) \\
&\quad + a_{33}(c_{\phi_b}c_{\theta_b})(s_{\theta_b}c_{\psi_b} + s_{\phi_b}c_{\theta_b}s_{\psi_b}), \\
M_{b,23} &= a_{11}(c_{\phi_b}s_{\theta_b})(c_{\theta_b}s_{\psi_b} + s_{\phi_b}s_{\theta_b}c_{\psi_b}) + a_{22}(c_{\phi_b}c_{\psi_b}s_{\phi_b}) \\
&\quad + a_{33}(c_{\phi_b}c_{\theta_b})(s_{\theta_b}s_{\psi_b} - s_{\phi_b}c_{\theta_b}c_{\psi_b}), \\
M_{b,44} &= (J_{b,xx} + a_{44}) + [(J_{b,zz} + a_{66}) - (J_{b,xx} + a_{44})]s_{\theta_b}^2, \\
M_{b,55} &= (J_{b,yy} + a_{55}) + [(J_{b,xx} + a_{44})s_{\theta_b}^2 + (J_{b,zz} + a_{66})c_{\theta_b}^2]t_{\phi_b}^2, \\
M_{b,66} &= (J_{b,zz} + a_{66}) + [(J_{b,xx} + a_{44}) - (J_{b,zz} + a_{66})]s_{\theta_b}^2/c_{\phi_b}^2, \\
M_{b,45} &= [(J_{b,xx} + a_{44}) - (J_{b,zz} + a_{66})]s_{\theta_b}c_{\theta_b}t_{\phi_b}, \\
M_{b,46} &= [(J_{b,zz} + a_{66}) - (J_{b,xx} + a_{44})]s_{\theta_b}c_{\theta_b}/c_{\phi_b}, \\
M_{b,56} &= -[(J_{b,xx} + a_{44})s_{\theta_b}^2 + (J_{b,zz} + a_{66})c_{\theta_b}^2]t_{\phi_b}/c_{\phi_b}. \tag{A.1}
\end{aligned}$$

Similarly, its damping matrix in \mathcal{W} , \mathbf{D}_b , is symmetric (i.e., $D_{ij} = D_{ji}$, for $i \neq j$), and has its non-zero elements explicitly given as:

$$\begin{aligned}
D_{b,11} &= b_{11}(c_{\theta_b}c_{\psi_b} - s_{\phi_b}s_{\theta_b}s_{\psi_b})^2 + b_{22}(c_{\phi_b}s_{\psi_b})^2 + b_{33}(s_{\theta_b}c_{\psi_b} + s_{\phi_b}c_{\theta_b}s_{\psi_b})^2, \\
D_{b,22} &= b_{11}(c_{\theta_b}s_{\psi_b} + s_{\phi_b}s_{\theta_b}c_{\psi_b})^2 + b_{22}(c_{\phi_b}c_{\psi_b})^2 + b_{33}(s_{\theta_b}s_{\psi_b} - s_{\phi_b}c_{\theta_b}c_{\psi_b})^2, \\
D_{b,33} &= b_{11}(c_{\phi_b}s_{\theta_b})^2 + b_{22}(s_{\phi_b})^2 + b_{33}(c_{\phi_b}c_{\theta_b})^2,
\end{aligned}$$

$$\begin{aligned}
D_{b,12} &= b_{11}(c_{\theta_b} s_{\psi_b} + s_{\phi_b} s_{\theta_b} c_{\psi_b})(c_{\theta_b} c_{\psi_b} - s_{\phi_b} s_{\theta_b} s_{\psi_b}) + b_{22}(-c_{\phi_b}^2 s_{\psi_b} c_{\psi_b}) \\
&\quad + b_{33}(s_{\theta_b} c_{\psi_b} + s_{\phi_b} c_{\theta_b} s_{\psi_b})(s_{\theta_b} s_{\psi_b} - s_{\phi_b} c_{\theta_b} c_{\psi_b}), \\
D_{b,13} &= b_{11}(-c_{\phi_b} s_{\theta_b})(c_{\theta_b} c_{\psi_b} - s_{\phi_b} s_{\theta_b} s_{\psi_b}) + b_{22}(-c_{\phi_b} s_{\psi_b} s_{\phi_b}) \\
&\quad + b_{33}(c_{\phi_b} c_{\theta_b})(s_{\theta_b} c_{\psi_b} + s_{\phi_b} c_{\theta_b} s_{\psi_b}), \\
D_{b,23} &= b_{11}(c_{\phi_b} s_{\theta_b})(c_{\theta_b} s_{\psi_b} + s_{\phi_b} s_{\theta_b} c_{\psi_b}) + b_{22}(c_{\phi_b} c_{\psi_b} s_{\phi_b}) \\
&\quad + b_{33}(c_{\phi_b} c_{\theta_b})(s_{\theta_b} s_{\psi_b} - s_{\phi_b} c_{\theta_b} c_{\psi_b}), \\
D_{b,44} &= b_{44} + [b_{66} - b_{44}] s_{\theta_b}^2, \\
D_{b,55} &= b_{55} + [b_{44} s_{\theta_b}^2 + b_{66} c_{\theta_b}^2] t_{\phi_b}^2, \\
D_{b,66} &= b_{66} + [b_{44} - b_{66}] s_{\theta_b}^2 / c_{\phi_b}^2, \\
D_{b,45} &= [b_{44} - b_{66}] s_{\theta_b} c_{\theta_b} t_{\phi_b}, \\
D_{b,46} &= [b_{66} - b_{44}] s_{\theta_b} c_{\theta_b} / c_{\phi_b}, \\
D_{b,56} &= -[b_{44} s_{\theta_b}^2 + b_{66} c_{\theta_b}^2] t_{\phi_b} / c_{\phi_b}.
\end{aligned} \tag{A.2}$$

APPENDIX B

UAV INERTIA MATRIX

The UAV's inertial matrix in \mathcal{W} is symmetric, with elements explicitly expressed as:

$$\begin{aligned} M_{u,11} &= m_u, & M_{u,22} &= m_u, & M_{u,33} &= m_u, \\ M_{u,12} &= 0, & M_{u,13} &= 0, & M_{u,23} &= 0, \\ M_{u,44} &= J_{u,xx} + [J_{u,zz} - J_{u,xx}]s_{\theta_u}^2, \\ M_{u,55} &= J_{u,yy} + [J_{u,xx}s_{\theta_u}^2 + J_{u,zz}c_{\theta_u}^2]t_{\phi_u}^2, \\ M_{u,66} &= J_{u,zz} + [J_{u,xx} - J_{u,zz}]s_{\theta_u}^2/c_{\phi_u}^2, \\ M_{u,45} &= [J_{u,xx} - J_{u,zz}]s_{\theta_u}c_{\theta_u}t_{\phi_u}, \\ M_{u,46} &= [J_{u,zz} - J_{u,xx}]s_{\theta_u}c_{\theta_u}/c_{\phi_u}, \\ M_{u,56} &= -[J_{u,xx}s_{\theta_u}^2 + J_{u,zz}c_{\theta_u}^2]t_{\phi_u}/c_{\phi_u}. \end{aligned} \tag{B.1}$$

APPENDIX C

SPHERICAL COORDINATES DETAILS

The spherical coordinates transformation to Cartesian coordinates, \mathbf{R}_{S2C} , is made by two ordered rotations, \mathbf{R}_α then $\mathbf{R}_{\nu\varphi}$, where:

$$\mathbf{R}_\alpha = \begin{bmatrix} c_\alpha & -s_\alpha & 0 \\ 0 & 0 & 1 \\ s_\alpha & c_\alpha & 0 \end{bmatrix}, \quad (\text{C.1})$$

and:

$$\mathbf{R}_{\nu\varphi} = \begin{bmatrix} c_\varphi & -s_\varphi & 0 \\ s_\varphi & c_\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (\text{C.2})$$

By differentiating \mathbf{r} with respect to time, we find the derivatives of the spherical unit vectors to be:

$$\begin{aligned} \dot{\hat{\mathbf{e}}}_r &= \dot{\alpha}\hat{\mathbf{e}}_\alpha + c_\alpha\dot{\varphi}\hat{\mathbf{e}}_\varphi, \\ \dot{\hat{\mathbf{e}}}_\alpha &= -\dot{\alpha}\hat{\mathbf{e}}_r - s_\alpha\dot{\varphi}\hat{\mathbf{e}}_\varphi, \\ \dot{\hat{\mathbf{e}}}_\varphi &= -c_\alpha\dot{\varphi}\hat{\mathbf{e}}_r + s_\alpha\dot{\varphi}\hat{\mathbf{e}}_\alpha, \end{aligned} \quad (\text{C.3})$$

From the definition of the spherical coordinates system and from (C.3), the position, velocity, and acceleration vectors of \mathcal{O}_u are given by:

$$\mathbf{r} = r\hat{\mathbf{e}}_r, \quad (\text{C.4a})$$

$$\dot{\mathbf{r}} = \dot{r}\hat{\mathbf{e}}_r + r\dot{\alpha}\hat{\mathbf{e}}_\alpha + rc_\alpha\dot{\varphi}\hat{\mathbf{e}}_\varphi, \quad (\text{C.4b})$$

$$\ddot{\mathbf{r}} = (\ddot{r} - r\dot{\alpha}^2 - rc_\alpha^2\dot{\varphi}^2)\hat{\mathbf{e}}_r + (r\ddot{\alpha} + 2\dot{r}\dot{\alpha} + rs_\alpha c_\alpha\dot{\varphi}^2)\hat{\mathbf{e}}_\alpha + (r\ddot{\varphi}c_\alpha + 2\dot{r}\dot{\varphi}c_\alpha - 2r\dot{\alpha}\dot{\varphi}s_\alpha)\hat{\mathbf{e}}_\varphi. \quad (\text{C.4c})$$

The components of the UAV's thrust vector in the spherical frame are explic-

itly realized from (3.26) as:

$$\begin{aligned}
\mathbf{u}_r &= \mathbf{u}_1 \left[c_\alpha c_\varphi (s_{\theta_u} c_{\psi_u} + s_{\phi_u} c_{\theta_u} s_{\psi_u}) + c_\alpha s_\varphi (s_{\theta_u} s_{\psi_u} - s_{\phi_u} c_{\theta_u} c_{\psi_u}) + s_\alpha (c_{\phi_u} c_{\theta_u}) \right], \\
\mathbf{u}_\alpha &= \mathbf{u}_1 \left[-s_\alpha c_\varphi (s_{\theta_u} c_{\psi_u} + s_{\phi_u} c_{\theta_u} s_{\psi_u}) - s_\alpha s_\varphi (s_{\theta_u} s_{\psi_u} - s_{\phi_u} c_{\theta_u} c_{\psi_u}) + c_\alpha (c_{\phi_u} c_{\theta_u}) \right], \\
\mathbf{u}_\varphi &= \mathbf{u}_1 \left[-s_\varphi (s_{\theta_u} c_{\psi_u} + s_{\phi_u} c_{\theta_u} s_{\psi_u}) + c_\varphi (s_{\theta_u} s_{\psi_u} - s_{\phi_u} c_{\theta_u} c_{\psi_u}) \right].
\end{aligned} \tag{C.5}$$

With the taut-cable condition in (3.30), we have $r = l$, and the spherical coordinates of the UAV with respect to the buoy's center of mass, \mathcal{O}_b , reduced to:

$$\mathbf{r}_u = \mathbf{r}_b + \mathbf{R}_{S2C} [l, 0, 0]^\top, \tag{C.6}$$

which expands to:

$$\begin{aligned}
x_u &= x_b + l c_\alpha c_\varphi, \\
y_u &= y_b + l c_\alpha s_\varphi, \\
z_u &= z_b + l s_\alpha,
\end{aligned} \tag{C.7}$$

and similarly, its velocity can be obtained as:

$$\dot{\mathbf{r}}_u = \dot{\mathbf{r}}_b + \mathbf{R}_{S2C} \dot{\mathbf{r}}, \tag{C.8}$$

which expands to:

$$\begin{aligned}
\dot{x}_u &= \dot{x}_b + (-l s_\alpha c_\varphi) \dot{\alpha} + (-l c_\alpha s_\varphi) \dot{\varphi}, \\
\dot{y}_u &= \dot{y}_b + (-l s_\alpha s_\varphi) \dot{\alpha} + (l c_\alpha c_\varphi) \dot{\varphi}, \\
\dot{z}_u &= \dot{z}_b + (l c_\alpha) \dot{\alpha}.
\end{aligned} \tag{C.9}$$

APPENDIX D

EULER-LAGRANGE FORMULATION

The Kinetic energy of the tethered UAV–buoy system, \mathcal{K} , includes the rotational and translational energies of the UAV and that of the buoy while neglecting the cable’s energy, and is explicitly described as:

$$\begin{aligned}
 \mathcal{K} &= \frac{1}{2} \left\{ \begin{aligned}
 &\dot{x}_b(M_{b,11}\dot{x}_b + M_{b,12}\dot{y}_b + M_{b,13}\dot{z}_b) \\
 &+ \dot{y}_b(M_{b,21}\dot{x}_b + M_{b,22}\dot{y}_b + M_{b,23}\dot{z}_b) \\
 &+ \dot{z}_b(M_{b,31}\dot{x}_b + M_{b,32}\dot{y}_b + M_{b,33}\dot{z}_b) \\
 &+ \dot{\phi}_b(M_{b,44}\dot{\phi}_b + M_{b,45}\dot{\theta}_b + M_{b,46}\dot{\psi}_b) \\
 &+ \dot{\theta}_b(M_{b,54}\dot{\phi}_b + M_{b,55}\dot{\theta}_b + M_{b,56}\dot{\psi}_b) \\
 &+ \dot{\psi}_b(M_{b,64}\dot{\phi}_b + M_{b,65}\dot{\theta}_b + M_{b,66}\dot{\psi}_b) \\
 &+ m_u(\dot{x}_u^2 + \dot{y}_u^2 + \dot{z}_u^2) \\
 &+ \dot{\phi}_u(M_{u,44}\dot{\phi}_u + M_{u,45}\dot{\theta}_u + M_{u,46}\dot{\psi}_u) \\
 &+ \dot{\theta}_u(M_{u,54}\dot{\phi}_u + M_{u,55}\dot{\theta}_u + M_{u,56}\dot{\psi}_u) \\
 &+ \dot{\psi}_u(M_{u,64}\dot{\phi}_u + M_{u,65}\dot{\theta}_u + M_{u,66}\dot{\psi}_u) \end{aligned} \right\} \\
 &= \frac{1}{2} \left\{ \begin{aligned}
 &\dot{x}_b [(M_{b,11} + m_u)\dot{x}_b + M_{b,12}\dot{y}_b + M_{b,13}\dot{z}_b] \\
 &+ \dot{y}_b [M_{b,21}\dot{x}_b + (M_{b,22} + m_u)\dot{y}_b + M_{b,23}\dot{z}_b] \\
 &+ \dot{z}_b [M_{b,31}\dot{x}_b + M_{b,32}\dot{y}_b + (M_{b,33} + m_u)\dot{z}_b] \\
 &+ m_u [(l\dot{\alpha})^2 + (lc_\alpha\dot{\varphi})^2 \\
 &- 2(ls_\alpha c_\varphi\dot{\alpha}\dot{x}_b) - 2(lc_\alpha s_\varphi\dot{\varphi}\dot{x}_b) \\
 &- 2(ls_\alpha s_\varphi\dot{\alpha}\dot{y}_b) + 2(lc_\alpha c_\varphi\dot{\varphi}\dot{y}_b) + 2(lc_\alpha\dot{\alpha}\dot{z}_b)] \\
 &+ \dot{\phi}_b(M_{b,44}\dot{\phi}_b + M_{b,45}\dot{\theta}_b + M_{b,46}\dot{\psi}_b) \\
 &+ \dot{\theta}_b(M_{b,54}\dot{\phi}_b + M_{b,55}\dot{\theta}_b + M_{b,56}\dot{\psi}_b) \\
 &+ \dot{\psi}_b(M_{b,64}\dot{\phi}_b + M_{b,65}\dot{\theta}_b + M_{b,66}\dot{\psi}_b)
 \end{aligned} \right\}
 \end{aligned}$$

$$\begin{aligned}
& + \dot{\phi}_u(M_{u,44}\dot{\phi}_u + M_{u,45}\dot{\theta}_u + M_{u,46}\dot{\psi}_u) \\
& + \dot{\theta}_u(M_{u,54}\dot{\phi}_u + M_{u,55}\dot{\theta}_u + M_{u,56}\dot{\psi}_u) \\
& + \dot{\psi}_u(M_{u,64}\dot{\phi}_u + M_{u,65}\dot{\theta}_u + M_{u,66}\dot{\psi}_u) \}. \tag{D.1}
\end{aligned}$$

The kinetic energy is differentiated with respect to each state rate then with respect to time. Only the buoy's translational states and tether's orientation states are of interest, since the UAV and the buoy's rotational states feature no dependencies on other states, which can be seen by inspection of the expanded \mathcal{K} . Their corresponding results are expressed as:

$$\begin{aligned}
\frac{d}{dt} \left(\frac{\partial \mathcal{K}}{\partial \dot{x}_b} \right) &= (M_{b,11} + m_u)\ddot{x}_b + M_{b,12}\ddot{y}_b + M_{b,13}\ddot{z}_b \\
&\quad + m_ul(-s_\alpha c_\varphi \ddot{\alpha} - c_\alpha c_\varphi \dot{\alpha}^2 - c_\alpha s_\varphi \ddot{\varphi} - c_\alpha c_\varphi \dot{\varphi}^2 + 2s_\alpha s_\varphi \dot{\alpha} \dot{\varphi}), \\
\frac{d}{dt} \left(\frac{\partial \mathcal{K}}{\partial \dot{y}_b} \right) &= M_{b,21}\ddot{x}_b + (M_{b,22} + m_u)\ddot{y}_b + M_{b,23}\ddot{z}_b \\
&\quad + m_ul(-s_\alpha s_\varphi \ddot{\alpha} - c_\alpha s_\varphi \dot{\alpha}^2 + c_\alpha c_\varphi \ddot{\varphi} - c_\alpha s_\varphi \dot{\varphi}^2 - 2s_\alpha c_\varphi \dot{\alpha} \dot{\varphi}), \\
\frac{d}{dt} \left(\frac{\partial \mathcal{K}}{\partial \dot{z}_b} \right) &= M_{b,31}\ddot{x}_b + M_{b,32}\ddot{y}_b + (M_{b,33} + m_u)\ddot{z}_b \\
&\quad + m_ul(c_\alpha \ddot{\alpha} - s_\alpha \dot{\alpha}^2), \\
\frac{d}{dt} \left(\frac{\partial \mathcal{K}}{\partial \dot{\alpha}} \right) &= m_u[l^2\ddot{\alpha} - ls_\alpha(c_\varphi \ddot{x}_b + s_\varphi \ddot{y}_b) + lc_\alpha \ddot{z}_b \\
&\quad + l(-c_\alpha c_\varphi \dot{\alpha} \dot{x}_b + s_\alpha s_\varphi \dot{\alpha} \dot{x}_b - c_\alpha s_\varphi \dot{\alpha} \dot{y}_b - s_\alpha c_\varphi \dot{\alpha} \dot{y}_b - s_\alpha \dot{\alpha} \dot{z}_b)], \\
\frac{d}{dt} \left(\frac{\partial \mathcal{K}}{\partial \dot{\varphi}} \right) &= m_u[l^2(c_\alpha^2 \ddot{\varphi} - 2s_\alpha c_\alpha \dot{\alpha} \dot{\varphi}) + lc_\alpha(-s_\varphi \ddot{x}_b + c_\varphi \ddot{y}_b) \\
&\quad + l(s_\alpha s_\varphi \dot{\alpha} \dot{x}_b - c_\alpha c_\varphi \dot{\alpha} \dot{x}_b - s_\alpha c_\varphi \dot{\alpha} \dot{y}_b - c_\alpha s_\varphi \dot{\alpha} \dot{y}_b)]. \tag{D.2}
\end{aligned}$$

The differentiation results of the UAV and buoy's rotational states are omitted as they result in similar components to the rotational dynamics already introduced in (3.27) and (3.21).

The kinetic energy of the system is differentiated with respect to the states, $(\frac{\partial \mathcal{K}}{\partial \eta})$, where the resulting nonzero elements are obtained as:

$$\begin{aligned}
\frac{\partial \mathcal{K}}{\partial \alpha} &= m_ul(-ls_\alpha c_\alpha \dot{\varphi}^2 - c_\alpha c_\varphi \dot{\alpha} \dot{x}_b + s_\alpha s_\varphi \dot{\alpha} \dot{x}_b - c_\alpha s_\varphi \dot{\alpha} \dot{y}_b - s_\alpha c_\varphi \dot{\alpha} \dot{y}_b - s_\alpha \dot{\alpha} \dot{z}_b), \\
\frac{\partial \mathcal{K}}{\partial \varphi} &= m_ul(s_\alpha s_\varphi \dot{\alpha} \dot{x}_b - c_\alpha c_\varphi \dot{\alpha} \dot{x}_b - s_\alpha c_\varphi \dot{\alpha} \dot{y}_b - c_\alpha s_\varphi \dot{\alpha} \dot{y}_b). \tag{D.3}
\end{aligned}$$

Notice that most elements of $\frac{\partial \mathcal{K}}{\partial \eta}$ cancels out with elements of $\frac{d}{dt} \left(\frac{\partial \mathcal{K}}{\partial \dot{\eta}} \right)$.

The system's potential energy was given in (3.38), and its external forces and moments vector can be formulated based on (3.22) and (3.2.6) as:

$$\begin{aligned}
\tau_1 &= \mathbf{u}_1(s_{\theta_u}c_{\psi_u} + s_{\phi_u}c_{\theta_u}s_{\psi_u}), \\
\tau_2 &= \mathbf{u}_1(s_{\theta_u}s_{\psi_u} - s_{\phi_u}c_{\theta_u}c_{\psi_u}), \\
\tau_3 &= \mathbf{u}_1c_{\phi_u}c_{\theta_u} + \rho_w g \Upsilon_{\text{im}}, \\
\tau_4 &= \mathbf{u}_1l[-s_{\alpha}c_{\varphi}(s_{\theta_u}c_{\psi_u} + s_{\phi_u}c_{\theta_u}s_{\psi_u}) - s_{\alpha}s_{\varphi}(s_{\theta_u}s_{\psi_u} - s_{\phi_u}c_{\theta_u}c_{\psi_u}) + c_{\alpha}(c_{\phi_u}c_{\theta_u})], \\
\tau_5 &= \mathbf{u}_1l[-s_{\varphi}(s_{\theta_u}c_{\psi_u} + s_{\phi_u}c_{\theta_u}s_{\psi_u}) + c_{\varphi}(s_{\theta_u}s_{\psi_u} - s_{\phi_u}c_{\theta_u}c_{\psi_u})],
\end{aligned} \tag{D.4}$$

Additionally, we have: $\boldsymbol{\tau}_{6-8} = \boldsymbol{\tau}_{2,u}$ and $\boldsymbol{\tau}_{9-11} = \boldsymbol{\tau}_{2,b}$. The potential energy's derivative with respect the states rates then to time is obtained as:

$$\frac{d}{dt} \left(\frac{\partial \mathcal{U}}{\partial \dot{\boldsymbol{\eta}}} \right) = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]^\top, \tag{D.5}$$

and its first derivative with respect to states is obtained as:

$$\frac{\partial \mathcal{U}}{\partial \boldsymbol{\eta}} = [0, 0, (m_b + m_u)g, m_u g l c_{\alpha}, 0, 0, 0, 0, 0, 0, 0]^\top. \tag{D.6}$$

APPENDIX E

ROS-GAZEBO SIMULATIONS

For further validation of the tethered UAV–buoy system, a customized model was developed in the Robot Operating System (ROS) Gazebo environment. Given that this tool is open-source and widely used by the robotics community worldwide, having a ROS implementation of the hereby proposed system will facilitate the accessibility and sustainability of this research topic.

E.1 Terminology

ROS uses its own terminology. The following definitions are provided for the reader’s reference:

node: a node is a process that performs computation. It is an executable program that runs inside of the application.

topic: topics are named buses over which nodes exchange messages.

publisher: a ROS publisher is a ROS node that publishes a specific type of ROS messages over a given ROS topic.

subscriber: a node that subscribes to the topic so that it receives the messages whenever any message is published to the topic.

launch file: it is an XML file that provides a convenient way to launch multiple nodes and a master, as well as other initialization requirements such as setting parameters.

spawn: the action of initializing a robot in the 3D Gazebo world.

E.2 Robot Model

Following standard descriptions of robots in ROS, the tethered UAV–buoy system is formed as a chain of joints and links, as illustrated by Fig. E.1. The model tree is detailed in Table E.1. We start with the buoy object, which is linked to the tether via a universal joint, where the tether is modeled as a chain of n rigid

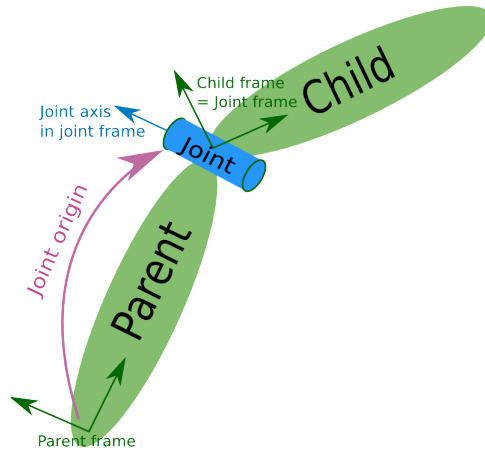


Figure E.1: Sample structure of the links and joint architecture in ROS [1].

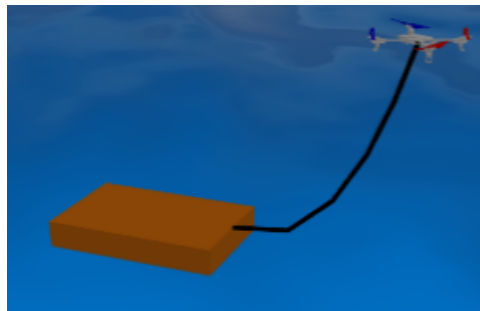


Figure E.2: Sample robot model for the tethered UAV–buoy system.

cylinders that are interconnected by means of revolute joints, then comes the UAV body that is also connected to the tether via a universal joint. Finally, the rotors are connected to the UAV body via continuous rotary joints. Note that each universal joint is modeled as one continuous joint in the yaw orientation followed by a revolute joint in the pitch orientation. We also note that sensory equipment is not included in the tree. This structure increases the system’s flexibility and modularity. For example, different types of buoy links or UAV links can be used without affecting the integrity of the system model. Note that the discretized number of the tether elements and the inertia definition of small links affect the stability of the model.

A sample robot model is constructed as described in Table E.1, and it is rendered in Fig. E.2. The buoy is considered of a cuboid shape, the tether is discretized into five elements, and the quadrotor UAV is selected to be an AR-drone [110] with increased motor power to deliver more thrust that can propel the system forward. The supplied library makes the design of the buoy and tether properties, including the number of discretized tether elements, easier. The UAV model is based on the RotorS package, which offers a variety of UAV models with

Table E.1: Links and Joints tree of the tethered UAV–buoy robot

Joint name	Joint type	Parent link	Child link
buoy- z	continuous- z	buoy body	buoy pitch link
buoy- y	revolute- y	buoy pitch link	tether element 1
tether joint 1	revolute- y	tether element 1	tether element 2
tether joint 2	revolute- y	tether element 2	tether element 3
...
tether joint n	revolute- y	tether element n	UAV yaw link
UAV- z	continuous- z	UAV yaw link	UAV body
rotor-1 joint	continuous- z	UAV body	rotor-1 link
rotor-2 joint	continuous- z	UAV body	rotor-2 link
rotor-3 joint	continuous- z	UAV body	rotor-3 link
rotor-4 joint	continuous- z	UAV body	rotor-4 link

their proper descriptions.

E.3 Simulation Environment

The simulation environment is based on the Gazebo simulator, which generates physical models of robots in the 3D space. Various ROS packages were used to build the system model, some packages were readily available and others had to be modified or developed. To include the UAV robots and their aerodynamics, we use the *RotorS* package; to include the water medium and the USV dynamics, we use the *VRX* package, and to design the tethered UAV–buoy system with its controller (DSVC), we develop our own package, which is detailed next.

RotorS: The *RotorS* package [111] lays the infrastructure for the flying UAV element in Gazebo, by providing multiple known multirotors with their physical properties. It also includes the sensors needed for UAV flight and sample controllers that allow direct flight in simulation. From this package, we make use of the UAV platforms, the odometry sensors, and the inner-loop attitude and thrust controller.

VRX: The Virtual RobotX (VRX) package [112] was developed in 2019 by the developers of the Gazebo simulator in order to fill the gap in marine robots simulation, where there were no available models for USVs, as compared to UAVs and UGVs. The *VRX* package defines the dynamic water surfaces environment with current and wave plugins, and handles the computation of the hydrodynamic forces that affect any USV object, given a set of predefined hydrodynamics coefficient in the Gazebo simulation environment. Both the *VRX* package and this work’s methodologies follow the marine dynamic modeling presented by Fossen

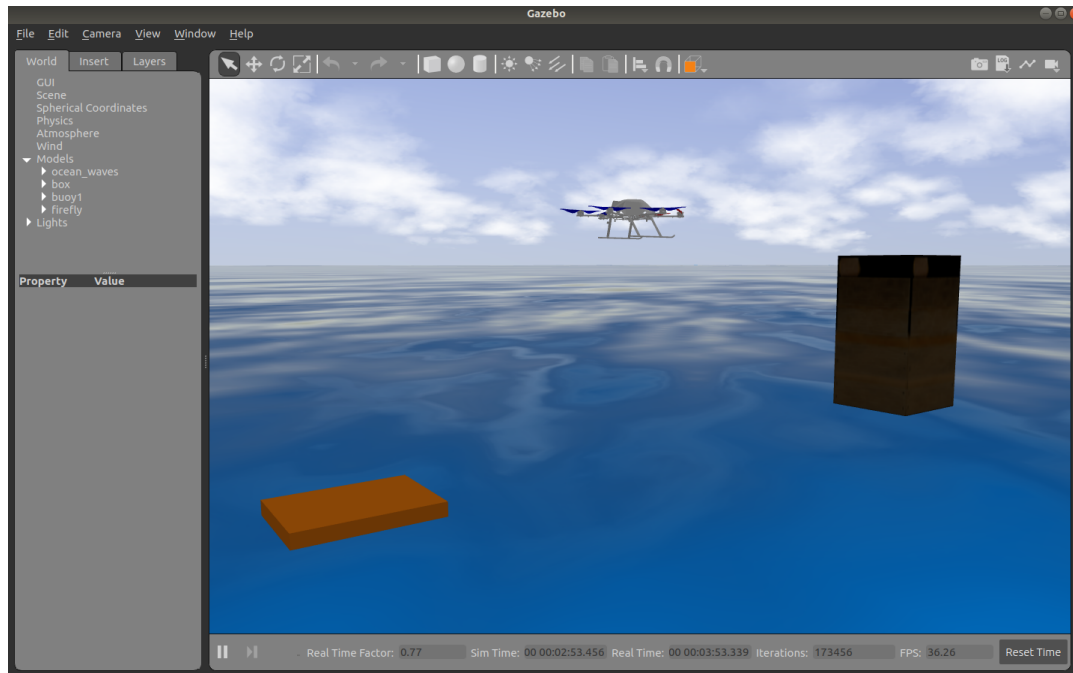


Figure E.3: Gazebo scene with independently spawned buoy (orange) and UAV robots in a water environment with a landing pad (black).

in [67]. The wave plugin allows users to set customized wave profiles, and the integrated Gerstner waves give the simulation environment a realistic sense. The restoring forces are calculated at discrete locations along the vessel hull.

RotorS + VRX: The first task of modeling the tethered UAV–buoy system in Gazebo is to build both packages together, and integrate their elements in the same simulation environment, that is, to deploy a floating buoy robot and a flying UAV robot in the same 3D environment. This is achieved by proper package installation and generating one launch file to include all of the simulation elements. To allow the UAV to properly initialize and start spinning its rotors, it needs to be laid on a fixed ground, otherwise it would sink in the water. For that purpose, a small (fixed) floating landing pad is required to be added to the system model. Note that the UAV link does not need to be equipped with the hydrodynamics plugin. To obtain sensing data for both the UAV and the buoy, each should be equipped with an odometry plugin, which is responsible for publishing the link’s pose and rates. With this data available, the spherical coordinates can be easily calculated by referring to Section 3.2.3. A sample simulation environment, with independently spawned buoy and UAV robots, is shown in Fig. E.3.

tuav-buoy: After establishing the above software infrastructure in Gazebo, the tethered UAV–buoy system is designed using a custom-built package that was specifically developed for this work. First, we spawn the tethered robot

as described in the **Robot Model** section, and we equip it with one *odometry* plugin at the main UAV link and another at the main buoy link. Additionally, the buoy link is equipped with a *hydrodynamics* plugin and proper definition of the hydrodynamic coefficients, \mathbf{D}'_b and $\mathbf{a}_{1,b}$. Next, the UAV is equipped with the designed DSVC system to be enable hovering and pulling actions, and to feed it with the correct commands and measurements of the buoy states, as is described next.

E.4 Implementation

Below is a list of elements and generated files that are required to model the tethered UAV–buoy system in Gazebo, with a brief description of each element’s function:

- Buoy Gazebo dynamics plugin: the VRX `wamv_gazebo_dynamics_plugin.xacro` file is upgraded to have a settable link name instead of the generic "base_link" name, which makes it possible to plug the hydrodynamic plugin on any link of the robot. Also, the hydrodynamic coefficients are modified to match those of the selected buoy model.
- buoy physical model file (.xacro): it is a cuboid shape buoy model, equipped with a *hydrodynamics* plugin, which can be spawned independently of the UAV robot. We note that selecting the buoy shape is not arbitrary; for instance, a buoy with a square-shaped vertical cross-section is not feasible.
- physical tethered UAV–buoy model file (.xacro): it is the main robot definition file that includes the UAV, the tether, and the buoy links in addition to their joints. It can be split into multiple .xacro files for convenience. It also includes the specific system parameters/properties, such as the buoy mass and tether length. Here, care should be taken with the choice and naming of the robot base link and its effect on both the *RotorS* and *VRX* packages.
- ocean world file (.xacro): this is the world file that should include the "ocean_waves" model from the *VRX* package, and the `librotors_gazebo_ros_interface_plugin.so` plugin from *RotorS*. Additionally, one can optionally include a landing pad for the UAV and other landmarks or floating objects within the world file.
- spawn ml-uav launch file (.launch): it includes the specific robot parameters and is responsible for spawning the tethered UAV–buoy system model in Gazebo. Note that a set of related files needs to be modified to handle the name change from the basic UAV robot name (*e.g.* AR-drone) to the new tethered UAV–buoy robot name (*e.g.* ml-uav).

- topics name (.h): in addition to the list of topics defined in `default_topics.h` by *RotorS*, the buoy odometry, spherical coordinates, and the DSVC commands topics need to be defined as well. This is achieved by adding the topics' names to the `default_topics.h` file.
- publishers: the additional publishers in the developed ROS package are the buoy odometry, the DSVC data, the initial spherical position set-point, and spherical way-point publishers.
- subscribers: the developed package adds a subscriber in the controller node to the buoy odometry topic. Also, since recording the spherical coordinates data is of interest, one can optionally subscribe a bag file to the DSVC data topic.
- way-points file (.txt): this is a text file that includes the DSVC control state, plus the reference radial position (m), UAV elevation (m), azimuth angle ($^{\circ}$), and buoy surge velocity (m/s).
- startup hovering example (.cpp): the *RotorS* `hovering_example.cpp` file is upgraded to publish a spherical coordinates set-point data instead of a Cartesian coordinates set-point.
- spherical way-point publisher (.cpp): the *RotorS* `waypoint_publisher_file.cpp` is upgraded to publish a series of spherical coordinates and buoy surge velocity trajectory points, with an indicator of the action type whether it is a relative position motion or buoy velocity manipulation.
- DSV controller (.cpp): the *RotorS* `lee_position_controller.cpp` file is upgraded with the DSV controller function, which is responsible for calculating the desired and actual spherical coordinates of the UAV, the commanded acceleration in the spherical frame, and its transformation into the world frame.
- controller gains file (.yaml): includes the DSVC's gains.
- Controller node: the *RotorS* `lee_position_controller_node.cpp` file is upgraded to use the DSV controller instead of the standard Cartesian position controller, and to subscribe to the spherical setpoint and waypoints publishers instead of their Cartesian counterparts. This node also subscribes to the buoy odometry topic, initializes the DSV controller parameters, and publishes the DSV controller data.
- main launch file (.launch): this file is responsible for initializing all of the ROS nodes, and it allows the users to choose the world file, the robot and its spawn location, in addition to the buoy's link name. The launched nodes are as follows: 1) Gazebo node with the specified world file and robots to

spawn, 2) controller node, 3) initial hovering position publisher node, 4) spherical waypoint publisher node, 5) robot state publisher node, and 6) joint state publisher node.

After listing and defining the elements and files that are required to model the tethered UAV–buoy system in Gazebo, a brief description of the working principle of the developed package is provided. First, the model’s `.urdf` file is prepared to include the robot’s description in terms of joints, links, and system dimensions and parameters. The controller script includes the DSVC control law presented in (3.53) and (3.58), which is called by the controller node, and it is responsible for stabilizing the outer-loop dynamics of the system. The signals required by the controller, namely the waypoints and the odometry messages, are published by other nodes. The controller node subscribes to these published signals and delivers them to the DSVC controller function, which in turn returns the desired inertial acceleration as per (3.60). The predefined *RotorS*’s inner-loop controller handles the remaining computations until delivering the commands to the motors. The launch file should include the followings steps: loading of the tethered UAV–buoy system model with the proper *USV dynamics* and *odometry* plugins, launching the Gazebo simulator with ocean world including the *waves* plugin and a landing pad, spawning the robot such that the buoy link is above the water surface and the UAV link is above the landing pad, launching the controller node, and finally launching the initializer hovering node then the waypoint publisher node.

E.5 Simulation Scenario

Since the developed ROS-gazebo package is meant to serve as a multi-purpose simulation environment, a sample simulation scenario is provided to demonstrate the functionality and fidelity of the package. Note that the package can be upgraded/customized with models, scripts, and plugins to fit a user’s needs.

The proposed simulation scenario considers an AR-drone quadrotor with enhanced thrust abilities, a buoy with a mass of 10 kg, and a tether of length 1.3 m discretized into five elements. At first, the UAV spawns above the landing pad, while being physically connected to a floating buoy via the tether. The UAV then takes off and hovers, while maintaining an initial position in a spherical coordinates frame that is centered at the buoy. The UAV prepares to enter the pulling phase by extending the tether to near its maximum length, then it starts gradually pulling the buoy up to a specified maximum speed. A turning maneuver, which consists of the buoy going straight then turning left, is executed in order to demonstrate the steering ability of the system, while obtaining adequate performance, bounded and practical control signals, and realistic values of the state variables.

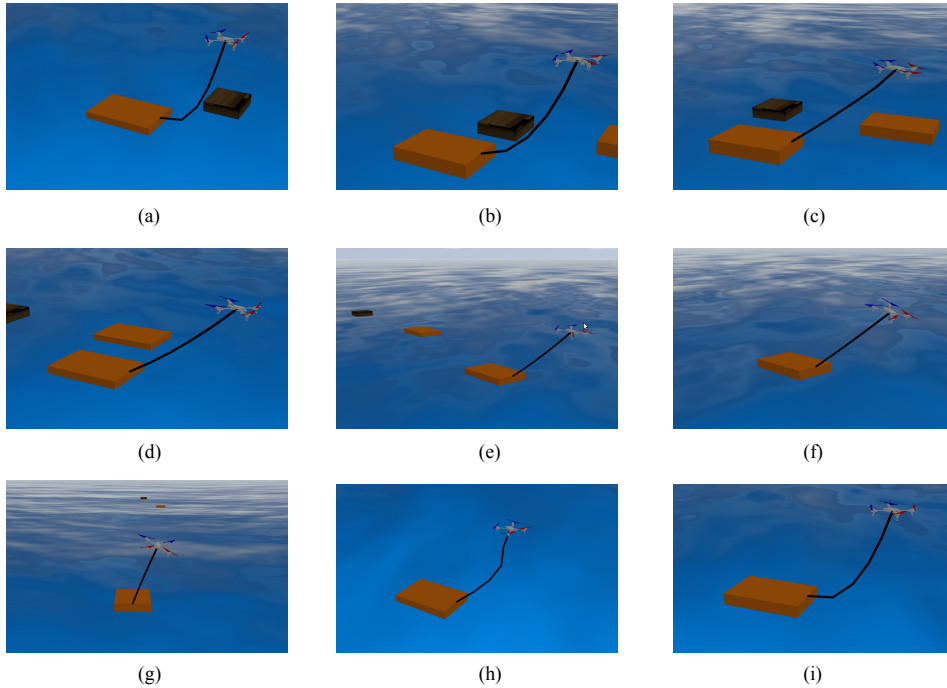


Figure E.4: Gazebo scene for the tethered UAV–buoy system at different moments during the locomotion task.

E.6 Simulation Results

Preliminary simulation results in the ROS-Gazebo environment are furnished to merely demonstrate the functionality of the developed ROS packages and the fidelity of the system’s dynamical model and designed controller. The simulation results of the devised scenario are shown in Fig. E.4. Notice the different stages of the simulation maneuver, starting from (a) initial position hold, (b) tether extension, (c-f) forward pulling (locomotion), (g) steering to the left, and finally (h-i) decoupling and returning to a spherical position hold.

The tracking performance of the system is shown in Fig. E.5. The planar ($x-y$) trajectory of the buoy is shown in the top subplot that captures the buoy’s forward motion in the x -direction at first, then turning left for the remainder of the simulation maneuver. The UAV exhibits good spherical position tracking in r , α , and φ , as well as adequate buoy surge velocity tracking.

The preliminary ROS-based simulation results demonstrate the following:

- The various ROS packages that were specifically developed for the proposed robotic system are functional.
- The UAV–buoy robot is properly defined and it is sensibly influenced by

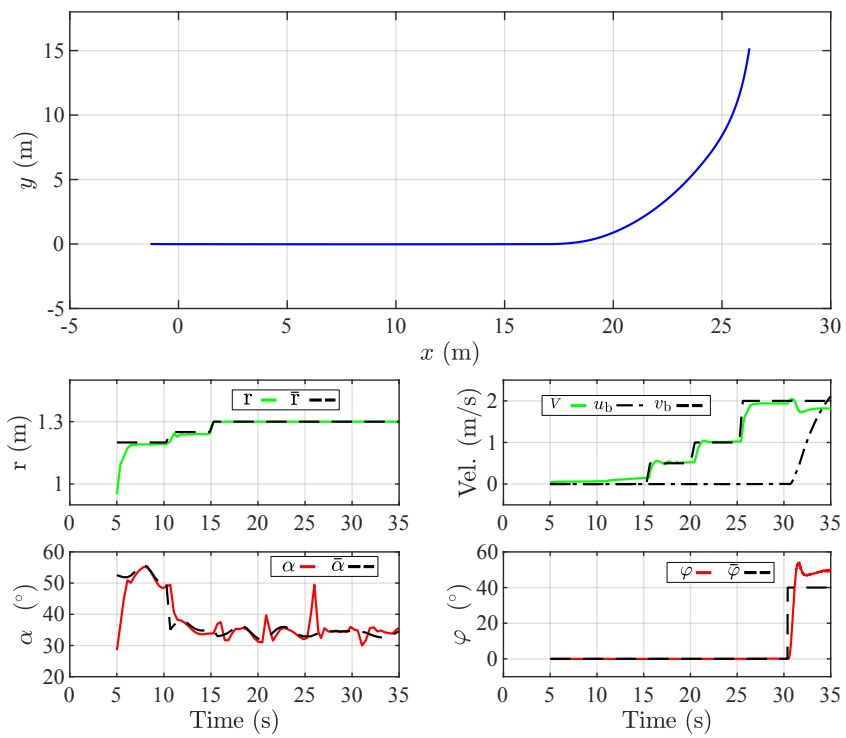


Figure E.5: Gazebo simulation tracking results.

the water environment and tether link forces.

- The proposed control system (DSVC) is implementable using the widely-used *RotorS* package for UAV simulation.
- Similar overall performance is obtained as compared with with the MATLAB/Simulink[®] model given in Section 3.4.3.

E.7 Future Work

Given that the developed *tuav-buoy* ROS package is not meant to be a replicate/replacement of the designed MATLAB/Simulink[®] model, there is room for further development, refinement, and additions to the package. For example, the state machine with its different operating modes can be included, other control system designs can be implemented, additional system dimensions and configurations can be evaluated, and various buoy shapes and sizes of can be tested.

APPENDIX F

REPOSITORIES

We make publicly available implementations of the proposed marine locomotive UAV system and the ICS tuning tools. This is presented by the following open-source GitHub [113] repositories:

1. `AUBVRL/Primitive-ML-UAV`: A MATLAB/Simulink implementation of the system proposed in "A Tethered Quadrotor UAV-Buoy System for Marine Locomotion", in ICRA 2021, by A. Kourani and N. Daher (coupled case only).
2. `AUBVRL/Marine-Loomotive-UAV`: A MATLAB/Simulink implementation of a robotic system consisting of a quadrotor UAV that pulls a floating buoy across water. Velocity manipulation of the buoy is allowed through a tether that links the UAV to the buoy (both coupled and decoupled cases for the 2D model).
3. `AUBVRL/3D-Tethered-UAV-buoy`: A MATLAB/Simulink implementation of the tethered UAV-buoy system (3D model).
4. `AUBVRL/tuav-buoy`: A ROS package for simulating the tethered UAV–buoy system.
5. `AUBVRL/Tune-Backstepping-Like-PID`: Some MATLAB scripts that show the similarity between the backstepping and the PID gains, and help tune a backstepping control law in a similar way to PID tuning.

BIBLIOGRAPHY

- [1] R. Angeli, “joint, ros wiki.” <http://wiki.ros.org/urdf/XML/joint>, 2022. [Online; accessed 6-January-2022].
- [2] K. Astrom and T. Hagglund, *PID Controller: Theory, Design and Tuning*. ISA: The Instrumentation, Systems, and Automation Society, 2 ed., 1995.
- [3] H. Yang, N. Staub, A. Franchi, and D. Lee, “Modeling and control of multiple aerial-ground manipulator system (MAGMaS) with load flexibility,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1–8, Oct 2018.
- [4] M. Krizmancic, B. Arbanas, T. Petrovic, F. Petric, and S. Bogdan, “Cooperative aerial-ground multi-robot system for automated construction tasks,” *IEEE Robotics and Automation Letters*, vol. 5, pp. 798–805, April 2020.
- [5] P. Tripicchio, M. Satler, G. Dabisias, E. Ruffaldi, and C. A. Avizzano, “Towards smart farming and sustainable agriculture with drones,” in *2015 International Conference on Intelligent Environments*, pp. 140–143, 2015.
- [6] amazon, “Amazon Prime Air.” www.amazon.com/Amazon-Prime-Air/b?ie=UTF8&node=8037720011, 2020. [Online; accessed 10-July-2020].
- [7] M. Kovac, “ORCA Hub.” www.offshore-technology.com/news/orca-hub-unveils-offshore-robotic-research-for-the-offshore-sector/, 2019. [Online; accessed 22-February-2020].
- [8] K. Sreenath, T. Lee, and V. Kumar, “Geometric control and differential flatness of a quadrotor UAV with a cable-suspended load,” in *IEEE Conference on Decision and Control*, pp. 2269–2274, 2013.
- [9] M. B. Srikanth, A. Soto, A. A. E. Lavretsky, and J.-J. Slotine, “Controlled manipulation with multiple quadrotors,” in *AIAA Guidance, Navigation, and Control Conference*, p. 6547, 2011.
- [10] R. Naldi, A. Gasparri, and E. Garone, “Cooperative pose stabilization of an aerial vehicle through physical interaction with a team of ground robots,” in

- 2012 IEEE International Conference on Control Applications*, pp. 415–420, Oct 2012.
- [11] N. Staub, D. Bicego, Q. Sablé, V. Arellano, S. Mishra, and A. Franchi, “Towards a flying assistant paradigm: the OTHex,” in *International Conference on Robotics and Automation*, pp. 6997–7002, 2018.
- [12] T. W. Nguyen, L. Catoire, and E. Garone, “Control of a quadrotor and a ground vehicle manipulating an object,” *Automatica*, vol. 105, pp. 384 – 390, 2019.
- [13] S. Lupashin and R. D’Andrea, “Stabilization of a flying vehicle on a taut tether using inertial sensing,” in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2432–2438, Nov 2013.
- [14] M. M. Nicotra, R. Naldi, and E. Garone, “Taut cable control of a tethered UAV,” *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 3190 – 3195, 2014.
- [15] M. M. Nicotra, R. Naldi, and E. Garone, “Nonlinear control of a tethered UAV: The taut cable case,” *Automatica*, vol. 78, pp. 174 – 184, 2017.
- [16] U. Weichenhain, “Cutting costs in infrastructure maintenance with UAVs.” www.rolandberger.com/fr/Publications/Drones-The-future-of-asset-inspection.html, 2019. [Online; accessed 22-February-2020].
- [17] M. Murison, “Aerones creates drone to de-ice and service wind turbines.” internetofbusiness.com/aerones-drone-de-ice-wind-turbines/, 2018. [Online; accessed 23-February-2020].
- [18] N. Mišković, S. Bogdan, đ. Nađ, F. Mandić, M. Orsag, and T. Haus, “Unmanned marsupial sea-air system for object recovery,” in *22nd Mediterranean Conference on Control and Automation*, pp. 740–745, 2014.
- [19] M. F. Ozkan, L. R. G. Carrillo, and S. A. King, “Rescue boat path planning in flooded urban environments,” in *2019 IEEE International Symposium on Measurement and Control in Robotics (ISMCR)*, pp. 221–229, 2019.
- [20] So-Ryeok Oh, K. Pathak, S. K. Agrawal, H. R. Pota, and M. Garratt, “Approaches for a tether-guided landing of an autonomous helicopter,” *IEEE Transactions on Robotics*, vol. 22, no. 3, pp. 536–544, 2006.
- [21] K. A. Talke, M. De Oliveira, and T. Bewley, “Catenary tether shape analysis for a UAV - USV team,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 7803–7809, Oct 2018.

- [22] S. Y. Choi, B. H. Choi, S. Y. Jeong, B. W. Gu, S. J. Yoo, and C. T. Rim, "Tethered aerial robots using contactless power systems for extended mission time and range," in *2014 IEEE Energy Conversion Congress and Exposition (ECCE)*, pp. 912–916, IEEE, 2014.
- [23] M. Tognon, H. A. T. Chávez, E. Gasparin, Q. Sablé, D. Bicego, A. Mallet, M. Lany, G. Santi, B. Revaz, J. Cortés, and A. Franchi, "A truly-redundant aerial manipulator system with application to push-and-slide inspection in industrial plants," *IEEE Robotics and Automation Letters*, vol. 4, pp. 1846–1851, April 2019.
- [24] J. Zhang, J. Xiong, G. Zhang, F. Gu, and Y. He, "Flooding disaster oriented USV and UAV system development and demonstration," in *OCEANS 2016 - Shanghai*, pp. 1–4, April 2016.
- [25] Y. Han and W. Ma, "Automatic monitoring of water pollution based on the combination of UAV and USV," in *2021 IEEE 4th International Conference on Electronic Information and Communication Technology (ICEICT)*, pp. 420–424, Aug 2021.
- [26] M. Saleh, G. Oueidat, I. H. Elhajj, and D. Asmar, "In situ measurement of oil slick thickness," *IEEE Transactions on Instrumentation and Measurement*, vol. 68, no. 7, pp. 2635–2647, 2019.
- [27] J. Dufek and R. Murphy, "Visual pose estimation of usv from uav to assist drowning victims recovery," in *2016 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pp. 147–153, Oct 2016.
- [28] M. F. Ozkan, L. R. G. Carrillo, and S. A. King, "Rescue boat path planning in flooded urban environments," in *2019 IEEE International Symposium on Measurement and Control in Robotics (ISMCR)*, pp. 221–229, 2019.
- [29] A. Gonzalez-Garcia, A. Miranda-Moya, and H. Castañeda, "Robust visual tracking control based on adaptive sliding mode strategy: Quadrotor UAV - catamaran USV heterogeneous system," in *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 666–672, June 2021.
- [30] S. Jung, H. Cho, D. Kim, K. Kim, J.-I. Han, and H. Myung, "Development of algal bloom removal system using unmanned aerial vehicle and surface vehicle," *IEEE Access*, vol. 5, pp. 22166–22176, 2017.
- [31] A. Kourani and N. Daher, "Marine locomotion: A tethered UAV-buoy system with surge velocity control," *Robotics and Autonomous Systems*, vol. 145, p. 103858, 2021.

- [32] N. Wang and C. K. Ahn, “Coordinated trajectory tracking control of a marine aerial-surface heterogeneous system,” *IEEE/ASME Transactions on Mechatronics*, pp. 1–1, 2021.
- [33] C. G. L. Krishna, M. Cao, and R. R. Murphy, “Autonomous observation of multiple USVs from UAV while prioritizing camera tilt and yaw over uav motion,” in *2017 IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*, pp. 141–146, Oct 2017.
- [34] G. Shao, Y. Ma, R. Malekian, X. Yan, and Z. Li, “A novel cooperative platform design for coupled USV–UAV systems,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 9, pp. 4913–4922, 2019.
- [35] Foxtech, “FOXTECH T3500 Plus Tethered Power System.” <https://www.foxtechfpv.com/foxtech-t3500-plus-tethered-power-system.html>, 2020. [Online; accessed 14-May-2021].
- [36] Z. Xu, “Application research of tethered UAV platform in marine emergency communication network,” *Journal of Web Engineering*, vol. 20, no. 2, pp. 491–512, 2021.
- [37] L. Xing, X. Liu, and S. Zhang, “The vibration characteristics analysis of the tethered UAV system,” *Yingyong Lixue Xuebao/Chinese Journal of Applied Mechanics*, vol. 38, no. 1, pp. 106–112, 2021.
- [38] E. Dicembrini, M. Scanavino, F. Dabbene, and G. Guglieri, “Modelling and simulation of a tethered UAS,” in *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 1801–1808, Sep. 2020.
- [39] M. Tognon and A. Franchi, “Dynamics, control, and estimation for aerial robots tethered by cables or bars,” *IEEE Transactions on Robotics*, vol. 33, pp. 834–845, Aug 2017.
- [40] Y. Liu, F. Zhang, P. Huang, and X. Zhang, “Analysis, planning and control for cooperative transportation of tethered multi-rotor UAVs,” *Aerospace Science and Technology*, vol. 113, p. 106673, 2021.
- [41] T. Kominami, H. Paul, R. Miyazaki, B. Sumetheeprasit, R. Ladig, and K. Shimonomura, “Active tethered hook: Heavy load movement using hooks that move actively with micro UAVs and winch system,” in *2021 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 264–269, July 2021.
- [42] C. Papachristos and A. Tzes, “The power-tethered UAV-UGV team: A collaborative strategy for navigation in partially-mapped environments,” in *22nd Mediterranean Conference on Control and Automation*, pp. 1153–1158, 2014.

- [43] G. Shao, Y. Ma, R. Malekian, X. Yan, and Z. Li, “A novel cooperative platform design for coupled USV–UAV systems,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 9, pp. 4913–4922, 2019.
- [44] C. Depcik, T. Cassady, B. Collicott, S. P. Burugupally, X. Li, S. S. Alam, J. R. Arandia, and J. Hobeck, “Comparison of lithium ion batteries, hydrogen fueled combustion engines, and a hydrogen fuel cell in powering a small unmanned aerial vehicle,” *Energy Conversion and Management*, vol. 207, p. 112514, 2020.
- [45] Z. Pan, L. An, and C. Wen, “Recent advances in fuel cells based propulsion systems for unmanned aerial vehicles,” *Applied Energy*, vol. 240, pp. 473–485, 2019.
- [46] F. G. Harmon, A. A. Frank, and J.-J. Chattot, “Conceptual design and simulation of a small hybrid-electric unmanned aerial vehicle,” *Journal of Aircraft*, vol. 43, no. 5, pp. 1490–1498, 2006.
- [47] Foxtech, “GAIA 160 elite pro hybrid drone.” <https://www.foxtechfpv.com/gaia-160-elite-hybrid-drone.html>, 2020. [Online; accessed 13-May-2021].
- [48] Foxtech, “World Record: 175 mins flight time and 100 KM range.” <https://www.foxtechfpv.com/gaia-160-elite-hybrid-drone.html>, 2020. [Online; accessed 13-May-2021].
- [49] Skyfront, “Products: perimeter 8.” <https://skyfront.com/product-list/>, 2021. [Online; accessed 13-May-2021].
- [50] Quaternium, “World best performance hybrid drone.” <https://www.quaternium.com>, 2021. [Online; accessed 13-May-2021].
- [51] W.-D. Chang, R.-C. Hwang, and J.-G. Hsieh, “A self-tuning PID control for a class of nonlinear systems based on the Lyapunov approach,” *Journal of Process Control*, vol. 12, no. 2, pp. 233 – 242, 2002.
- [52] Astrom, *Adaptive Control*. Addison-Wesley, New York, 1995.
- [53] P. H. Chang and J. H. Jung, “A systematic method for gain selection of robust PID control for nonlinear plants of second-order controller canonical form,” *IEEE Transactions on Control Systems Technology*, vol. 17, pp. 473–483, March 2009.
- [54] J. Y. Lee, M. Jin, and P. H. Chang, “Variable PID gain tuning method using backstepping control with time-delay estimation and nonlinear damping,” *IEEE Transactions on Industrial Electronics*, vol. 61, pp. 6975–6985, Dec 2014.

- [55] J. Han, “From PID to active disturbance rejection control,” *IEEE Transactions on Industrial Electronics*, vol. 56, pp. 900–906, March 2009.
- [56] A. Benaskeur and A. Desbiens, “Backstepping-based adaptive PID control,” *IEE Proceedings - Control Theory and Applications*, vol. 149, pp. 54–59(5), January 2002.
- [57] T. Yamaguchi and K. Shimizu, “Asymptotic stabilization by PID control: Stability analysis based on minimum phase and high-gain feedback,” *Electrical Engineering in Japan*, vol. 156, no. 1, pp. 783–791, 2006.
- [58] H. Yu, Z. Guan, T. Chen, and T. Yamamoto, “Design of data-driven PID controllers with adaptive updating rules,” *Automatica*, vol. 121, p. 109185, 2020.
- [59] M. Chen, S. S. Ge, and B. Ren, “Adaptive tracking control of uncertain MIMO nonlinear systems with input constraints,” *Automatica*, vol. 47, no. 3, pp. 452 – 465, 2011.
- [60] J. Yu, P. Shi, and L. Zhao, “Finite-time command filtered backstepping control for a class of nonlinear systems,” *Automatica*, vol. 92, pp. 173 – 180, 2018.
- [61] B. Yao and M. Tomizuka, “Adaptive robust control of SISO nonlinear systems in a semi-strict feedback form,” *Automatica*, vol. 33, no. 5, pp. 893–900, 1997.
- [62] M. Mohd Basri, A. Husain, and K. Danapalasingam, “Enhanced backstepping controller design with application to autonomous quadrotor unmanned aerial vehicle,” *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 79, no. 2, pp. 295–321, 2015.
- [63] F. Muttin, “Umbilical deployment modeling for tethered UAV detecting oil pollution from ship,” *Applied Ocean Research*, vol. 33, no. 4, pp. 332–343, 2011.
- [64] H. V. Abeywickrama, B. A. Jayawickrama, Y. He, and E. Dutkiewicz, “Comprehensive energy consumption model for unmanned aerial vehicles, based on empirical studies of battery performance,” *IEEE Access*, vol. 6, pp. 58383–58394, 2018.
- [65] Y. Pan, G. Liu, and K. D. Kumar, “Robust stability analysis of asymptotic second-order sliding mode control system using Lyapunov function,” in *The 2010 IEEE International Conference on Information and Automation*, pp. 313–318, June 2010.

- [66] O. M. Faltinsen, *Sea Loads on Ships and Offshore Structures*. Cambridge University Press, 1990.
- [67] T. I. Fossen, *Guidance and Control of Ocean Vehicles*. WILEY, 1995.
- [68] J. Zhang, D. Gu, Z. Ren, and B. Wen, “Robust trajectory tracking controller for quadrotor helicopter based on a novel composite control scheme,” *Aerospace Science and Technology*, vol. 85, pp. 199–215, 2019.
- [69] G. Fridsma, “A systematic study of the rough-water performance of planing boats,” tech. rep., Davidson Laboratory Report 1275, Stevens Institute of Technology, 1969.
- [70] D. J. Kim, S. Y. Kim, Y. J. You, K. P. Rhee, S. H. Kim, and Y. G. Kim, “Design of high-speed planing hulls for the improvement of resistance and seakeeping performance,” *International Journal of Naval Architecture and Ocean Engineering*, vol. 5, no. 1, pp. 161 – 177, 2013.
- [71] M. Tognon and A. Franchi, *Theory and Applications for Control of Aerial Robots in Physical Interaction Through Tethers*. Springer, 2020.
- [72] A. Kourani and N. Daher, “Leveraging PID gain selection towards adaptive backstepping control for a class of second-order systems,” in *2021 American Control Conference (ACC)*, pp. 1174–1179, May 2021.
- [73] E. Tal and S. Karaman, “Accurate tracking of aggressive quadrotor trajectories using incremental nonlinear dynamic inversion and differential flatness,” in *2018 IEEE Conference on Decision and Control (CDC)*, pp. 4282–4288, 2018.
- [74] C. Hu, B. Yao, and Q. Wang, “Integrated direct/indirect adaptive robust contouring control of a biaxial gantry with accurate parameter estimations,” *Automatica*, vol. 46, no. 4, pp. 701 – 707, 2010.
- [75] E. V. Lewis, *Principles of Naval Architecture (Second Revision), Volume II - Resistance, Propulsion and Vibration*. Society of Naval Architects and Marine Engineers (SNAME), 1988.
- [76] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons, Ltd, 2011.
- [77] C. Aoun, N. Daher, and E. Shamma, “An energy optimal path-planning scheme for quadcopters in forests,” in *2019 IEEE 58th Conference on Decision and Control (CDC)*, pp. 8323–8328, 2019.

- [78] L. A. Sandino, D. Santamaria, M. Bejar, A. Viguria, K. Kondak, and A. Ollero, “Tether-guided landing of unmanned helicopters without gps sensors,” in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3096–3101, May 2014.
- [79] J. M. Daly, Y. Ma, and S. L. Waslander, “Coordinated landing of a quadrotor on a skid-steered ground vehicle in the presence of time delays,” *Autonomous Robots*, vol. 38, no. 2, pp. 179–191, 2015.
- [80] A. Hussein, P. Marín-Plaza, D. Martín, A. de la Escalera, and J. M. Armingol, “Autonomous off-road navigation using stereo-vision and laser-rangefinder fusion for outdoor obstacles detection,” in *2016 IEEE Intelligent Vehicles Symposium (IV)*, pp. 104–109, June 2016.
- [81] K. Schauwecker and A. Zell, “On-board dual-stereo-vision for the navigation of an autonomous MAV,” *Journal of Intelligent & Robotic Systems*, vol. 74, pp. 1–16, Apr. 2014.
- [82] M. Pizzoli, C. Forster, and D. Scaramuzza, “Remode: Probabilistic, monocular dense reconstruction in real time,” in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2609–2616, May 2014.
- [83] D. Falanga, E. Mueggler, M. Faessler, and D. Scaramuzza, “Aggressive quadrotor flight through narrow gaps with onboard sensing and computing using active vision,” in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5774–5781, May 2017.
- [84] Y. Xie, A. Savvarisal, A. Tsourdos, D. Zhang, and J. Gu, “Review of hybrid electric powered aircraft, its conceptual design and energy management methodologies,” *Chinese Journal of Aeronautics*, vol. 34, no. 4, pp. 432–450, 2021.
- [85] A. Kourani and N. Daher, “A tethered quadrotor UAV-buoy system for marine locomotion,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 59–65, May 2021.
- [86] S. Dai, T. Lee, and D. S. Bernstein, “Adaptive control of a quadrotor UAV transporting a cable-suspended load with unknown mass,” in *53rd IEEE Conference on Decision and Control*, pp. 6149–6154, 2014.
- [87] SANDWICHAERO, “GS 1000W Hybrid Generator For Drones.” <https://sandwichaero.com/products/gs-1000-hybrid-generator>, 2021. [Online; accessed 16-May-2021].
- [88] ELISTAIR, “SAFE-T 2 tether drone station.” <https://elistair.com/safe-t-tethered-drone-station/>, 2021. [Online; accessed 16-May-2021].

- [89] X. Xiao, J. Dufek, and R. Murphy, “Benchmarking tether-based UAV motion primitives,” in *2019 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, pp. 51–55, Sep. 2019.
- [90] J. Dufek, X. Xiao, and R. Murphy, “Visual pose stabilization of tethered small unmanned aerial system to assist drowning victim recovery,” in *2017 IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*, pp. 116–122, Oct 2017.
- [91] X. Xiao, J. Dufek, and R. R. Murphy, “Autonomous visual assistance for robot operations using a tethered UAV,” in *Field and Service Robotics* (G. Ishigami and K. Yoshida, eds.), (Singapore), pp. 15–29, Springer Singapore, 2021.
- [92] A. Kourani and N. Daher, “Bidirectional manipulation of a buoy with a tethered quadrotor UAV,” in *2021 IEEE 3rd International Multidisciplinary Conference on Engineering Technology (IMCET)*, pp. 101–106, Dec 2021.
- [93] A. Kourani and N. Daher, “Three-dimensional modeling and control of a tethered UAV-buoy system,” *Nonlinear Dynamics (NODY)*, in review, 2021.
- [94] P. Ranger and A. Desbiens, “Improved backstepping-based adaptive PID control,” in *2003 4th International Conference on Control and Automation Proceedings*, pp. 123–127, June 2003.
- [95] R. Skjetne and T. I. Fossen, “On integral control in backstepping: analysis of different techniques,” in *Proceedings of the 2004 American Control Conference*, vol. 2, pp. 1899–1904, June 2004.
- [96] A. A. Mian, M. I. Ahmad, and D. Wang, “Backstepping based PID control strategy for an underactuated aerial robot,” *IFAC Proceedings Volumes*, vol. 41, no. 2, pp. 15636 – 15641, 2008.
- [97] Y. Kartal, P. Kolaric, V. Lopez, A. Dogan, and F. Lewis, “Backstepping approach for design of PID controller with guaranteed performance for micro-air UAV,” *Control Theory and Technology*, vol. 18, pp. 19–33, Feb. 2019.
- [98] B. Yao and R. G. Dontha, “Integrated direct/indirect adaptive robust precision control of linear motor drive systems with accurate parameter estimations,” *IFAC Proceedings Volumes*, vol. 35, no. 2, pp. 587–592, 2002. 2nd IFAC Conference on Mechatronic Systems, Berkeley, CA, USA, 9-11 December.
- [99] M. Krstic, I. Kanellakopoulos, and P. Kokotovic, *Nonlinear and Adaptive Control Design*. Wiley, 1995.

- [100] B. Yao and M. Tomizuka, “Adaptive robust control of MIMO nonlinear systems in semi-strict feedback forms,” *Automatica*, vol. 37, pp. 1305–1321, 2001.
- [101] Y. Zou, “Nonlinear robust adaptive hierarchical sliding mode control approach for quadrotors,” *International Journal of Robust and Nonlinear Control*, vol. 27, pp. 925–941, 2017.
- [102] B. Yao, “Adaptive robust motion control of single-rod hydraulic actuators: Theory and experiments,” *IEEE/ASME Transactions on Mechatronics*, vol. 5, pp. 79–91, March 2000.
- [103] W. Dong, J. A. Farrell, M. M. Polycarpou, V. Djapic, and M. Sharma, “Command filtered adaptive backstepping,” *IEEE Transactions on Control Systems Technology*, vol. 20, pp. 566–580, May 2012.
- [104] G. F. Simmons, *Calculus with Analytic Geometry*. McGraw-Hill, 2 ed., 1996.
- [105] Quanser, “Quanser innovate-educate, www.quanser.com.” www.quanser.com, 2018. [Online; accessed 20-April-2018].
- [106] A. Kourani, K. Kassem, and N. Daher, “Coping with quadcopter payload variation via adaptive robust control,” in *2018 IEEE International Multidisciplinary Conference on Engineering Technology (IMCET)*, pp. 1–6, 2018.
- [107] S. A. Raza and J. Etele, “Autonomous position control analysis of quadrotor flight in urban wind gust conditions,” in *AIAA Guidance, Navigation, and Control Conference*, 2016.
- [108] B. Lotfi and M. Goharimanesh, “Modelling and control of quadrotor maneuvers with variations of center of gravity (COG),” in *IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 1570–1574, 2015.
- [109] M. Dhaybi and N. Daher, “Accurate real-time estimation of the inertia tensor of package delivery quadrotors,” in *2020 American Control Conference (ACC)*, pp. 1520–1525, July 2020.
- [110] Parrot, “Ar drone, www.parrot.com.” <https://www.parrot.com/us/support/documentation/ar-drone>, 2020. [Online; accessed 7-January-2022].
- [111] F. Furrer, M. Burri, M. Achtelik, and R. Siegwart, *Robot Operating System (ROS): The Complete Reference (Volume 1)*, ch. RotorS—A Modular

Gazebo MAV Simulator Framework, pp. 595–625. Cham: Springer International Publishing, 2016.

- [112] B. Bingham, C. Agüero, M. McCarrin, J. Klamo, J. Malia, K. Allen, T. Lum, M. Rawson, and R. Waqar, “Toward maritime robotic simulation in gazebo,” in *Proceedings of MTS/IEEE OCEANS Conference*, (Seattle, WA), October 2019.
- [113] Github, “Github, github.com.” <https://github.com>, 2022. [Online; accessed 7-January-2022].