FULL DUPLEX HYBRID ACOUSTIC/RF COMMUNICATION FOR UNDERWATER NETWORKED CONTROL SYSTEMS

by
SAEED NOURIZADEH AZAR

Submitted to the Graduate School of Social Sciences
in partial fulfilment of
the requirements for the degree of Doctor of Philosophy

Sabanci University
December 2022

ABSTRACT

FULL DUPLEX HYBRID ACOUSTIC/RF COMMUNICATION FOR UNDERWATER NETWORKED CONTROL SYSTEMS

SAEED NOURIZADEH AZAR

Computer Science Engineering Ph.D. Dissertation, December 2022

Dissertation Supervisor: Prof. Özgür Gürbüz
Dissertation Co-Supervisor: Assoc. Prof. Ahmet Onat

Keywords: Underwater communication, Full duplex communication, Co-simulation, Autonomous underwater vehicle, Networked control system, MAC, Acoustic

Underwater mission planning, monitoring, and coordination of heterogeneous autonomous underwater vehicle (AUV)s require a considerable amount of time and financial resources. This has led to the requirement of establishing reliable communication networks among unmanned underwater vehicle (UUV)s as well as a simulation environment to realistically model the system's dynamics before actual testing in sea trials. Even though existing solutions can model the dynamics of underwater vehicles, due to complexity, the integration of real-time communication networks has not been considered in the works. To address this issue, this thesis presents an innovative design and realistic co-simulation for a networked control systems (NCS) to achieve navigation of UUVs through communication and control, which is a critical component of real-world marine applications.

Traditionally, underwater communication has been based on acoustic communications, characterized by limited data rate and considerably large propagation delay. Taking this issue into consideration, in this thesis, a hybrid acoustic and radio frequency (RF) communication framework is proposed for the underwater NCS where an acoustic link is used for long distance communication and control, and an RF link is employed in the short range. Additionally, to maximize spectrum efficiency, adopting full duplex (FD) communication is proposed for both underwater acoustic and RF links. FD communication enables the feedback signal of the NCS to be

transmitted rapidly to several AUVs through simultaneous transmission and reception.

For the proposed underwater hybrid NCS, a docking scenario is considered, where AUVs perform maneuvers towards a docking station fixed at the seabed. In this scenario, the docking station determines the position of the nearby AUVs, and acoustic or RF communication links carry the position and navigation information from the docking station to AUVs via different medium access control (MAC) protocols. With the help of FD communication, it can be ensured that the underwater hybrid NCS system operates at maximum efficiency, providing the required feedback signal more frequently than NCS with half-duplex communication, resulting in faster and more accurate docking. The AUVs are equipped with two types of controllers for pursuing and actuating docking maneuver: Proportional Integral Derivative (PID) and Linear Quadratic Regulator (LQR) controller, whose gains and sampling times are determined according to the operation of the underwater hybrid NCS.

Depending on the communication protocol used in the NCS, protocol delays may be different which forces a change in the sampling times. The different delays of the control loop require further changes in the controller gains to avoid instability. For this purpose, in this thesis, optimization of the controller gains is proposed for the underwater hybrid NCS by applying Sequential Model Algorithm Configuration (SMAC) method for the PID controller and for the LQR controller, by mathematically modeling the hydrodynamics of the AUV to provide better control over disturbances and nonlinearity. By considering the full dynamics of the entire system for controlling the AUVs, the real-time behavior of the underwater networked control system is evaluated realistically using the proposed integrated co-simulation environment, which includes different simulators working together.

The performance results indicate that under calm water conditions, our proposed FD underwater hybrid NCS using LQR achieves the shortest docking time of approximately 62 seconds, while the corresponding SMAC optimized approach in FD mode takes around 97 seconds. Furthermore, using FD mode on the acoustic link with the LQR controller reduces the docking time by about 78 seconds. In contrast, for the PID-based method, the docking time is almost doubled to 148 seconds. The underwater hybrid NCS is also evaluated under realistic fluctuating water currents, using two controllers, different MAC protocols, and FD and HD communication modes. Our experiments indicate that with LQR, the proposed FD underwater hybrid NCS's docking time, when exposed to such currents, is 90 seconds, while the SMAC optimized PID takes approximately 175 seconds. In contrast, the conventional acoustic-based HD mode using LQR for realistic currents has a docking time of around 120 seconds, while the SMAC optimized PID takes about 245 seconds. The penalty to achieve improved performance using FD hybrid is spending 70% more motive energy than the acoustic only system. It is worth noting that communication modes using SMAC optimized PID cannot complete docking maneuvers if the current speed exceeds $0.3m/s$, while LQR based methods can handle current speeds up to $0.7m/s$. At this velocity, conventional acoustic-based systems take about 140% longer to complete docking than our proposed FD hybrid system. These results demonstrate the feasibility and advantages of using the proposed FD hybrid communication approach for AUV control.

ÖZET

SU ALTI AĞ BAĞLANTILI KONTROL SİSTEMLERİ İÇİN TAM ÇİFT YÖNLÜ HİBRİT AKUSTİK/RF HABERLEŞME

SAEED NOURIZADEH AZAR

BİLGİSAYAR MÜHENDİSLİĞİ DOKTORA TEZİ, ARALIK 2022

Tez Danışmanı: Prof. Dr. Özgür Gürbüz
Tez Eş-Danışmanı: Doç. Dr. Ahmet Onat

Anahtar Kelimeler: Sualtı haberleşmesi, Tam çift yönlü haberleşme, Birleşik simülasyon, Otonom sualtı araçları, Ağ bağlantılı Kontrol Sistemi, Ortama erişim kontrolü, akustik

Heterojen otonom sualtı araçlarının (OSA'lar) sualtı görev planlaması, izlenmesi ve koordinasyonu önemli miktarda zaman ve finansal kaynak gerektirir. Bu durum gerçek deniz ortamına çıkmadan önce insansız su altı araçları (İSAA'lar) arasındaki haberleşme ağının yanı sıra sistem dinamiklerinin gerçekçi bir şekilde modellenmesi için bir simülasyon ortamı oluşturulması gerekliliğini doğurmuştur. Mevcut çözümler, sualtı araçlarının dinamiklerini modelleme yeteneğine sahip olsa da karmaşıklık nedeniyle, önceki çalışmalarda haberleşme ağlarının entegrasyonu dikkate alınmamıştır. Bu konuyu ele almak için, bu tezde gerçek dünya deniz senaryolarının ayrılmaz bir parçası olan İSAA'ların birbirleriyle iletişim kurmasını sağlamak için Ağ bağlantılı kontrol sisteminin (AKS) yeni bir tasarımı ve gerçekçi benzetimi sunulmaktadır.

Sualtı haberleşmesi çoğunlukla akustik haberleşmeye dayanır. Akustik haberleşmede kullanılan kontrol kazançları sınırlı bir veri hızı ile birlikte yayılma gecikmesi nedeni ile kısıtlıdır. Bu tezde, uzun mesafeli haberleşme ve kontrol için akustik bağlantının ve kısa mesafede radyo frekansı (RF) bağlantısının bırlıkte kullanıldığı bir Ağ bağlantılı Kontrol Sistemi kurmak için hibrit bir akustik ve RF haberleşme sistemi önerilmiştir. Ayrıca, kullanılan haberleşme kanallarındaki veri hızlarını iyileştirmek için tam çift yönlü (TÇY) haberleşme kullanılması önerilmiştir. Tam çift yönlü haberleşme, yarı çift yönlü haberleşme ile aynı bant genişliğini kullanırken

kablosuz bağlantıların veri aktarım hızlarını ikiye katlama potansiyeline sahiptir.

Önerilen su altı hibrit ağ bağlantılı kontrol sistemi için bir uygulama olarak, yanaşma manevralarının OSA'lar tarafından gerçekleştirildiği bir senaryo ele alınmıştır. Önerilen senaryoda, akustik veya RF haberleşme yoluyla OSA'ların konumunu belirleyen bir yerleştirme istasyonu bulunmakta ve haberleşme türüne bağlı olarak araçların iletimini koordine etmek için farklı ortam erişim kontrolü (OEK) protokolleri kullanılmaktadır. TÇY haberleşme kullanılarak, su altı hibrit AKS sisteminin, standart yarı çift yönlü haberleşmeden daha hassas ve hızlı yanaşma manevraları yapması mümkün olabilmekte, gerekli seyir verileri daha yüksek sıklıkla sağlanarak AKS'nin daha yüksek verimlilikte çalışması sağlanabilmektedir. OSA'ler yanaşma manevrası yapmak için iki tür kontrolörle donatılmıştır: Orantılı İntegral Türev (PID) ve Lineer Kuadratik Düzenleyici (LQR) kontolörü. Kontrolörün kazançları ve örnekleme süreleri, su altı hibrit AKS'nin çalışmasına göre belirlenir.

AKS'de kullanılan haberleşme protokolüne bağlı olarak, paket gecikmeleri, dolayısıyla örnekleme süreleri farklılık gösterebilir, bu da kontrolörün kazanımlarının optimize edilmesini gerektirir. Bu amaçla, bu tezde, PID kontrolör için Sıralı Model Algoritması Konfigürasyonu (SMAC) yöntemi uygulanarak kontrolörün kazançlarının optimizasyonunu önerilmektedir. LQR kontrolör için de, bir OSA'nin hidrodinamiğini matematiksel olarak modelleyerek, sistemlerin bozulmaları ve nonlineerlikleri üzerinde daha iyi kontrol sağlanması amaçlanmıştır. Önerilen su altı ağ bağlantılı kontrol sisteminin gerçek zamanlı davranışı, OSA'ları kontrol etmek için tüm sistemin bütün dinamikleri göz önüne alınarak, farklı simülatörleri birlikte kullanan entegre bir simülasyon ortamında, gerçekçi bir şekilde değerlendirilmektedir.

Performans sonuçları, durgun su koşullarında, LQR kullanan önerilen TÇY hibrit AKS'nin yaklaşık 62 saniyelik en kısa yanaşma süresine ulaştığını, TÇY modunda SMAC ile optimize edilmiş yaklaşımın ise yaklaşık 97 saniye sürdüğünü göstermektedir. Ayrıca, LQR kontrolörü ile akustik bağlantıda TÇY modunun kullanılması yanaşma süresini yaklaşık 78 saniye azaltmaktadır. Buna karşılık, PID tabanlı yöntem için yerleştirme (kenetlenme) süresi neredeyse ikiye katlanarak 148 saniyeye çıkmıştır. Sualtı hibrit AKS, iki kontrolör, farklı OEK protokolleri ve TÇY ve Yarı Çift Yönlü (YÇY) iletişim modları kullanılarak gerçekçi dalgalanan su akımları altında da değerlendirilmektedir. Deneylerimiz, önerilen TÇY hibrit AKS'nin bu tür akımlara maruz kaldığında kenetlenme süresinin 90 saniye olduğunu, SMAC için optimize edilmiş PID'nin ise yaklaşık 175 saniye sürdüğünü göstermektedir. Buna karşılık, gerçekçi akımlar için LQR kullanan geleneksel akustik tabanlı AKS'de TÇY modunun yanaşma süresi yaklaşık 120 saniyeyken, SMAC için optimize edilmiş PID yaklaşık 245 saniye sürmektedir. Hibrit TÇY kullanarak edinilen performanstaki iyileştirmeye karşılık, akustik sistemden %70 daha fazla hareket enerjisi harcanmaktadır. Akıntı hızı 0,3 m/s'yi aşarsa, SMAC için optimize edilmiş PID kullanan iletişim modlarının yanaşma manevralarını tamamlayamadığı görülmüştür. Öte yandan, LQR tabanlı yöntemler 0,7 m/s'ye kadar akıntı hızlarını kaldırabilmektedir. Bu hızda, geleneksel akustik tabanlı sistemlerin yanaşmayı tamamlaması, önerdiğimiz TÇY hibrit sistemimizden yaklaşık %140 daha uzun sürer. Bu sonuçlar, OSA kontrolü için önerilen TÇY hibrit haberleşme yaklaşımının kullanılmasının uygulanabilirliğini ve avantajlarını göstermektedir.

ACKNOWLEDGEMENTS

First and foremost, I wish to express my sincere gratitude to my supervisors, Prof. Dr. Ozgur Gurbuz and Assoc. Prof Ahmet Onat for their invaluable advice, continuous support, and patience during my PhD study. Their immense knowledge and plentiful experience have encouraged me in all the time of my academic research and daily life. I would also like to thank Dr.Vahid Tavakol Aghaei for his support in my study. I would like to thank all the members in the Sabanci Telecommunication and Networking laboratory especially Dr Mikail Erdem for sharing his thoughts and consultations with me. Finally, I would like to express my gratitude to my parents, my wife and my son. Without their tremendous understanding and encouragement in the past few years, it would be impossible for me to complete my studies.

To my lovely wife Nastaran and my dear son Raymon.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

ADC Analog to Digital Converters. 28

ANN Artificial Neural Networks. 44

AUV autonomous underwater vehicle. iii, 1

CDMA Code Division Multiple Access. 26

CML Concurrent Mapping and Localization. 22

CSMA/CA Carrier Sense Multiple Access/Collision Avoidance. 32

DOF degree of freedom. 10

DS Docking Station. 5

DSIC digital SI cancellation. 26

DVL Doppler Velocity Log. 22

EKF Extended Kalman Filter. 22

FD full duplex. iii

FDD Frequency Division Duplex. 26

FDMA Frequency Division Multiple Access. 34

HD half duplex. 7

IBFD In-band full duplex. 26

INS Inertial Navigation System. 22

KF Kalman Filter. 22

LBL long baseline. 22

LQG Linear Quadratic Gaussian. 42

LQR Linear Quadratic Regulator. iv, 5

MAC medium access control. iv, 2

MI magnetic induction. 2

NCS networked control systems. iii, 1

NED North East Down. 10

PID Proportional Integral Derivative. iv, 4

QOS quality of service. 3

RF radio frequency. iii, 2

ROS Robot Operating System. 7

ROV Remotely Operated Vehicle. 2

S-ALOHA Slotted ALOHA. 32

S-CW FD Synchronized Contention Window Full Duplex. 38

SG Synchronization Gap. 34

SI self-interference. 26

SLAM Simultaneous Localization and Mapping. 22

SMAC Sequential Model Algorithm Configuration. iv, 5

SMC Sliding Mode Controller. 43

SNAME Society of Naval Architects and Marine Engineers. 20, 21

SNR Signal to Noise Ratio. 24

TDD Time Division Duplex. 26

TDMA Time Division Multiple Access. 32

TG Terminal Gap. 34

TI Transmit Interval. 34

USBL ultra-short baseline. 22

UUV unmanned underwater vehicle. iii

WR Waiting Room. 32

Chapter 1

INTRODUCTION

An autonomous underwater vehicle (AUV) is an underwater robot that works independently of an operator. An AUV can control the thrusters and propellers as well as operate sensors with an onboard computer and power supply to move the vehicle through the water to perform predefined critical missions. To perform complex underwater operations, swarms of AUVs can cooperate to undertake hazardous missions for divers. The use of AUVs permits a high level of accuracy for applications requiring precision, such as military missions, archaeology, and underwater infrastructure monitoring and maintenance. These operations also require high data rate communication between AUVs and base stations.

networked control systems (NCS)s are distributed systems, where the controller and sensor functions of a feedback control system are performed by different nodes which exchange messages among themselves over a communication network [1]. NCS approach has been especially preferred over conventional control systems and is extensively employed in industrial applications due to reduced complexity and cost [1]. Underwater NCSs, involving communication and control of AUVs, require reliable high-speed network communication between underwater vehicles. The successful implementation of underwater networks has been realized using optical waves, acoustic waves, and radio frequency [2].

Underwater communication systems mostly use acoustic technology since lower frequency waves suffer less from absorption loss. The acoustic waves in underwater communication are low-frequency signals with long wavelengths. Hence they can traverse long ranges in the order of kilometers for relaying information [2]. Acoustic communication is a proven technology for deep underwater as well as shallow water. However, in shallow water applications, it is severely affected by the time-varying multipath effect and high levels of ambient noise due to waves and other

movements [3]. Although there are some works such as [4] for decreasing the end to end delay among AUVs using acoustic links, the data rate and propagation speed of acoustic channels are insufficient for emerging applications like docking at an underwater base and controlling swarms of AUVs. For the precise control and coordination of AUVs, a high data rate and short sampling time are required. significantly by propagation delay due to the physical limitation factor of sound speed [5].

radio frequency (RF) communication is another alternative to provide high data rates and low propagation delay under the water without the line of sight requirement [6]. However, under the water, RF waves suffer from higher and frequency dependent absorption, which causes higher path loss, limiting the range of operation and requires careful calibration of frequency, antenna design, and transmission power [7]. Authors in [8] developed RF path loss models in different underwater conditions to address these issues. Propagation of RF waves from air into freshwater was studied in [9], and an optimum frequency range of 3 - 100 MHz was found for sending signals to a depth of 5m. As research in underwater RF communication is evolving, medium access control (MAC) schemes are also being studied. In [10], ALOHA and CSMA are compared, concluding that CSMA without acknowledgment is the most appropriate MAC for an underwater RF network with low traffic.

There is also the option of using magnetic induction (MI) communication to carry out underwater communication. Like RF, the communication range of MI is influenced by water conductivity and even though it has a high propagation speed of $33m/\mu s$, it can be used for short ranges [5, 11, 12]. Finally, optical waves have the most significant spectrum bandwidth and a faster propagation speed of 225 $m/\mu s$. However, water related features such as scattering and high absorption restrict the optical communication range to roughly 500m in clear water and about 10m in turbulent water [13]. Experiments using wide-beam LED transmitters [14], [15] report covering a range of 150m with a data rate of 10 Mbps employing underwater optical communication. Additionally, laboratory studies have reported data rates of 5.5 Gbps at 26m [16] and 500 Mbps at 100m by exploiting collimated laser beams [17] at the expense of becoming directional.

Campagnaro et al., in [18] proposed a multimodal optical and acoustic control system for underwater communication that allows the underwater vehicle to perform its mission. Additionally, they have offered proactive switching by applying a signaling mechanism to enable more reactive switching between acoustic and optical modes. To evaluate the performance of the multimodal acoustic and optical system, an inspection-class hybrid Remotely Operated Vehicle (ROV) was simulated in the DESERT simulator considering the application layers. Similarly, in their recent

work [19] considering the remote control design, Campagnaro et al. have investigated several wireless communication technologies such as acoustic, optic, RF, and magnetic induction (MI) and described their pros and cons for underwater communication. Furthermore, they evaluated AUVs' achievable quality of service (QOS) in a simulated underwater environment. Moreover, according to the autonomy level of the vehicle, control requirements for different applications, such as navigation, positioning and real-time video streaming, are proposed.

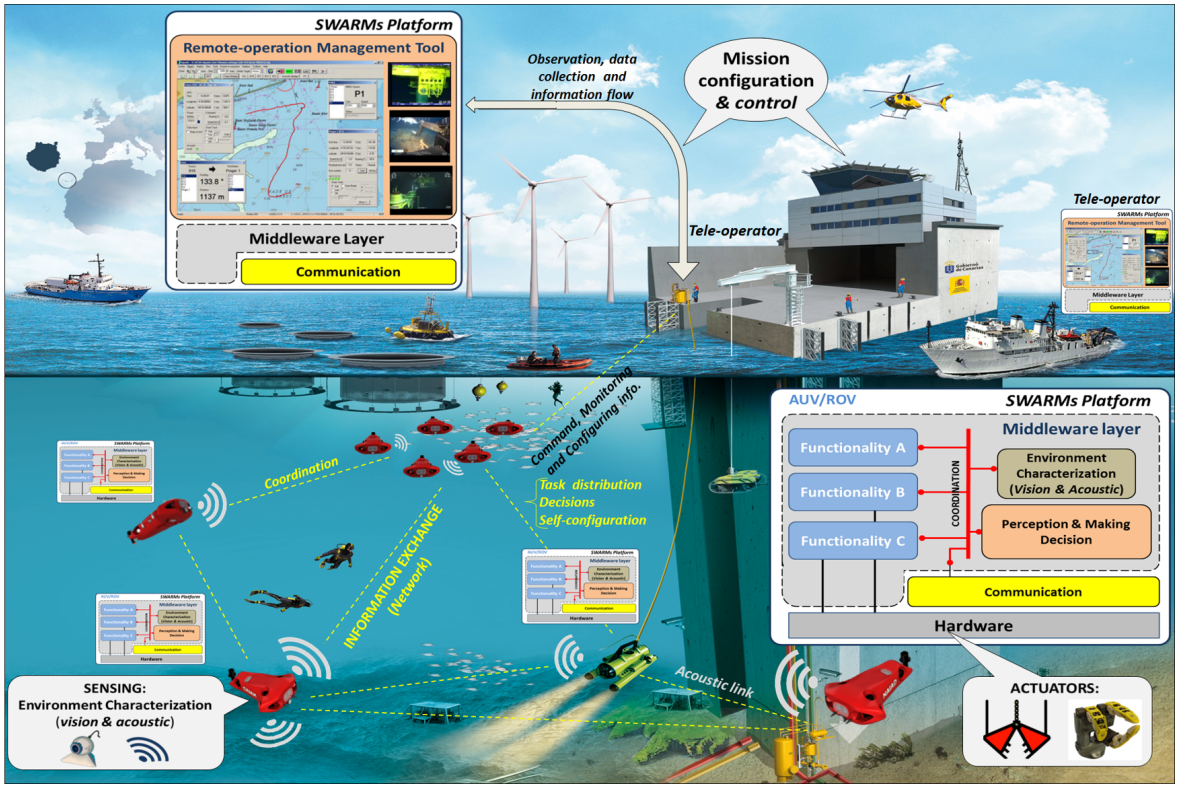


Figure 1.1 Collaboration among various UUVs in the SWARMS project to accomplish a mission.

In an effort to offer different communication modes for underwater applications, the authors in [5] have proposed a universal modem using the software defined networking concept to facilitate different communication modes over one device. The idea seems appealing and reduces the cost of integrating several modems for different communications modes. Several front-end modules are responsible for the transmission and reception of data with one of the available communication modes, including acoustic, RF, MI, and Optics. Authors in [20] have proposed establishing an underwater acoustic sensor network, where nodes utilize high speed connections when they have enough data to transmit, by performing surfacing and transmitting the data over high speed radio links for longer distances by using wireless terrestrial modems. Although some algorithms have been proposed to minimize the number of surfacing, this is only applicable to limited underwater applications, due to the con-

siderable amount of energy consumed by frequent surfacing and diving. Different above mentioned wireless communication methods have their advantages, disadvantages and limitations. Therefore, the optimal underwater communication mode should be determined according to application, water characteristics and operation range. Existing multimode solutions have been used for fixed underwater sensor networks, but not for controlling the underwater vehicle autonomously in an NCS, as presented in this thesis.

This thesis is based on our research for the EU-supported Horizon 2020 project, namely SWARMs. A key objective of SWARMs is to coordinate heterogeneous underwater vehicles from different vendors so that they can be managed and controlled remotely through a communication infrastructure by adding software interfaces and hardware (if needed) to the existing UUV/AUVs. Various scenarios and use cases were implemented for this goal within the SWARMs project. In addition to evaluating the results using simulators, sea trial missions have also been conducted. Figure 1.1 illustrates the interaction between an offshore command control system and a UUV as well as a supporting ship with developed software architecture and a communications network. This thesis considers a specific part of this comprehensive scenario for a docking maneuver application for guiding AUVs to a docking station fixed at the seabed.

In this thesis, we present a novel underwater hybrid NCS for the docking maneuver application, where AUVs are steered to the docking station, using position measurements sent over Acoustic or RF communication links in a networked control setting. The main contribution is the scheduling of acoustic and RF channels based on the distance of the AUV to the docking station, considering half-duplex and full-duplex modes with three different communication protocols as well as optimized control via three different controllers [21, 22].

The key component to enable the AUV to work autonomously is developing a robust NCS that performs navigation using remotely obtained position information. An NCS works by inserting a communication link within the control loop. In the docking application, AUV's position (which can be obtained by employing a sonar position sensor) is periodically measured by the docking station and sent to the AUV for control. A conventional Proportional Integral Derivative (PID) controller can be used for control, as it has been proven to be suitable in various applications. However, due to the sensitivity of derivative terms in PID to noise measurements and the uncertainty in estimating the linear and angular velocity, these issues must be addressed precisely for PID to function appropriately. Additionally, as a result of the nonlinearity of underwater vehicles and increased phase delay due to com-

munication, PID may not be able to provide sufficient control authority in varying underwater environments.

One approach to control the AUV is using a separate PID for each axis of movement [23]. Even though this method provides a solution for properly controlling AUVs, re-adjusting the PID by changing the environment and communication parameters remains challenging. In fact, by changing the MAC protocols and communication modes, the rate of receiving messages is affected, which in turn affects the performance of PID. This is mainly due to the elimination of nonlinear parameters of the system during the simplification of the model to be used with PID, resulting in poor performance during the mission. It should be noted that, due to the highly nonlinear nature of the AUV, the PID controller will neither guarantee system stability nor provide the system with optimal control. PID performance can be improved by utilizing Docking Station (DS) positioning systems equipped with arrays of acoustic positioning systems to pinpoint the positions of AUVs precisely. Nevertheless, the varying delays caused by different network protocols and during transmission in NCS may result in the controller becoming unstable.

For comparing the performances of several controllers with each other, optimization techniques for adjusting the controller gain are inevitable. Moreover, the gains of the controller should be determined automatically. Several approaches are available for determining optimal PID parameters, including Ziegler-Nicolls, Root locus, etc., which is essential in evaluating the various protocols fairly [24]. However, the nonlinearity of the system and lack of a dynamic model of the AUV make it challenging to apply these techniques and do not guarantee optimal results [25]. A possible solution would be to perform hyperparameter optimization using methods, such as Sequential Model Algorithm Configuration (SMAC) [26]. This method involves determining a cost function, iteratively configuring the parameters to fit the model, and examining the model's fitting. The parameters are then selected based on the fitting to the model. Nevertheless, this is an offline approach, which requires that the desired parameters be optimized in advance. The drawback of this method is that it may have difficulty handling uncertain events during the mission.

Alternatively, a Linear Quadratic Regulator (LQR) controller can be implemented to handle unexpected events during the mission. The LQR is one of the most widely used controllers in linear control systems. Using the LQR controller, closed-loop feedback systems in NCS can be controlled in an algorithmic manner [24, 27]. LQR uses a quadratic cost function in the linear system to minimize the cost function and provide optimal results. The AUV is more stable in the presence of external forces. The implementation of LQR, however, is complex when it comes to controlling large

systems with numerous parameters. Consequently, it is sensitive to modeling errors that adversely affect its performance.

Accordingly, in order to enable AUVs to complete the required missions successfully, several challenges across different domains must be tackled. These challenges include issues such as underwater communication technologies, reliable and rapid data delivery, precise positioning guidance, and distributed control. As the development of autonomous vehicles continues to garner more interest, we are motivated to conduct research on this topic and propose potential solutions to these challenges.

1.1 Contributions

In this work by taking into account the data rate constraint of acoustic communication and close range limitation of RF link, we propose a novel NCS for controlling AUV to perform docking maneuver by employing hybrid acoustic and RF communication. In the considered networked control scenario, the AUV location is remotely measured by a fixed docking station equipped with distance measuring sensors and transmitted to the AUV using the proposed hybrid communication network.

The proposed hybrid NCS uses the acoustic mode for long distances, which is greater than a threshold distance, and when it approaches the vicinity of the docking station (i.e., RF range), it switches to the RF mode. These messages provide the required information for the navigation of AUV. Low speed and large delay characteristics of the acoustic signals led to low control gains that are insufficient for precise navigation. However, employing high sampling frequency and high RF mode data rate at the docking station's proximity enables accurate docking maneuvering. Furthermore, we assess the effects of water currents on the underwater networked control system communication for the AUV navigation and demonstrate how the NCS is affected via changes in the communication model and characteristics of the water.

To improve the data rate and spectral efficiency of the communication links, we further introduce full duplex communication in our hybrid NCS. While full duplex communication has already been applied for underwater acoustic communication, this is the first attempt to introduce FD mode for underwater RF communication. Using FD communication in underwater hybrid NCS significantly improves the system performance due to significantly reduced (halved) delay, hence the sampling rate, which allows an increase of the sampling frequency and accuracy of control.

To achieve unbiased and automated tuning of the controller gains in NCS, this thesis employs the SMAC method to optimize PID controller gains and the model-based LQR controller. As a result of the optimal gains of the LQR controller, distur-

bances and changing the system parameters during the mission are more effectively controlled. This approach improves the performance and stability of the docking maneuver, enabling more reliable and efficient autonomous docking.

We have developed a novel integrated co-simulation environment for performance analysis, which allows real-time operation of the entire NCS. Moreover, the dynamic behavior of the underwater environment and movement of the AUVs are simulated realistically using UUV Simulator, Robot Operating System (ROS) and Gazebo, whereas the hybrid communication framework including the channel characteristics, protocols, control algorithm and continuous dynamics are modeled via Matlab True-Time on simulated real-time embedded computers. To the best of our knowledge, this is the first work that demonstrates an integrated co-simulation environment for testing underwater networked control applications taking into account the full dynamics of the system and controlling AUVs in a networked control framework using hybrid acoustic and RF communication. Via detailed simulations in the proposed integrated co-simulation environment, the performance of the proposed underwater hybrid NCS has been investigated in terms of time to dock, cumulative error, communication energy and motive energy considering both a calm water scenario and a realistic scenario modeling water currents effects.

Our experiments in the proposed integrated co-simulation environment show that, in a calm water scenario, the underwater hybrid NCS using LQR achieves the shortest docking time of approximately 62 seconds, compared to the equivalent SMAC optimized approach in FD mode, which takes around 97 seconds. The conventional acoustic-based method's time to dock using LQR and optimized PID controllers is approximately 85 and 165 seconds, respectively. However, using FD mode on the acoustic link with the LQR controller reduces the docking time to 78 seconds, while the PID-based method takes almost twice as long, with a docking time of 148 seconds.

Our experiments also show that an AUV using our proposed FD hybrid method completes docking in 90 seconds under realistic fluctuating water currents using LQR. In contrast, the conventional acoustic-based half duplex (HD) mode, also using LQR, takes approximately 120 seconds to dock under similar conditions. The results indicate that FD hybrid systems require 70% more motive energy than acoustic-only systems to achieve better performance. Furthermore, SMAC optimized PIDs are unable to complete docking maneuvers if the current speed exceeds $0.3m/s$, while LQR methods can handle speeds up to $0.7m/s$. Compared to our proposed FD hybrid system, traditional acoustic-based systems take approximately 140% longer to dock at this velocity.

1.2 Organization

The rest of the dissertation is organized as follows. Chapter 2 covers general background information about autonomous underwater vehicles, AUV's architecture, and different docking methods in the literature. Chapter 3 presents underwater communication technologies and channel modeling. Chapter 4 discusses the proposed system model as well as the MAC protocols adopted for the implementation of NCS. The focus of Chapter 5 is on designing and utilizing different controllers for AUVs. An integrated co-simulation environment for the underwater NCS is presented in Chapter 6. Chapter 7 discusses our simulation experiments and results. Conclusions along with further discussions are presented in Section 8.

Chapter 2

AUTONOMOUS UNDERWATER VEHICLES (AUV)

As depicted in Figure 2.1 there are several sensors associated with the underwater vehicle that are highlighted below for environmental sensing in this thesis.

2.1 AUV Modeling and Hydrodynamics

This section provides mathematical models of the UUVs and hydrodynamics. The vector based notations used in modeling are based on Fossen's equations of motion [29]. In fact, using matrices, differential equations with 6 DOF with coupling effects can be treated efficiently. Hence obtaining a hydrodynamic model is critical for simulating UUV and designing observers, controllers, and thruster allocation systems.

Using a controller is essential to provide autonomy to any vehicle. The dynamic of AUVs, similar to other naval vehicles such as ships, ROVs, and submarines, are nonlinear and extremely affected by the motion of currents and vulnerable to hydrodynamic uncertainties. Applying the open loop method for the movement of ROV/AUV results in spinning on the motion due to the fact that linear velocities are not steady during stabilization. In order for the AUV's velocity and position to be controlled, it needs a controller [30].

The following assumptions are considered to simplify obtaining the equations of motion.

- An AUV is a rigid body and entirely under the water
- It is assumed that water is a perfect fluid (i.e., incompressible, inviscid, and

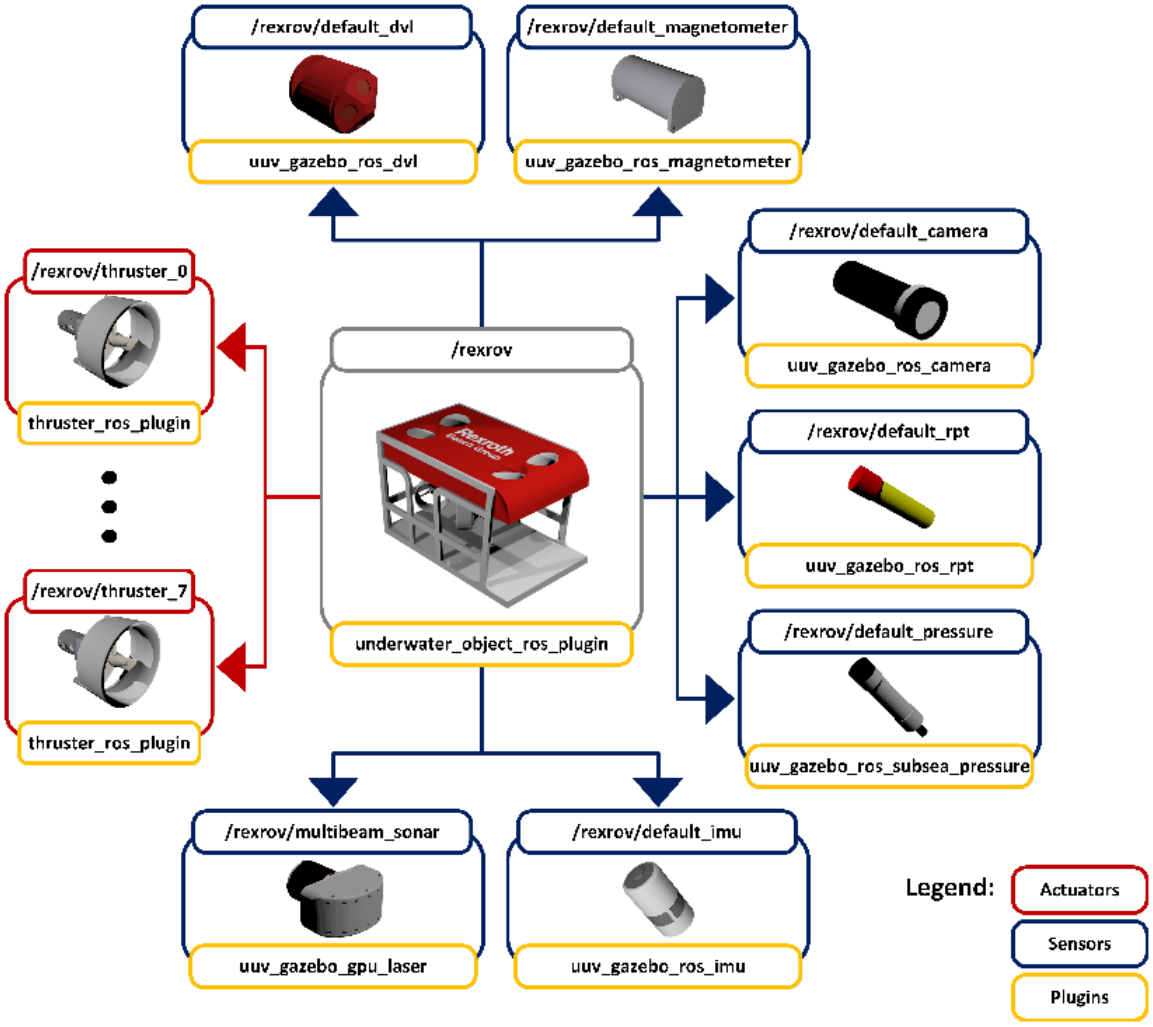


Figure 2.1 Different sensor plugins in REX AUV architecture [28].

frictionless)

- An AUV moves at a low speed
- The earth-fixed frame North East Down (NED) frame of reference is inertial
- The moving reference is the body frame that is fixed to the vehicle
- Water currents effects are neglected

2.1.1 AUV Kinematics

Mathematical modeling is the most crucial part of designing and simulating underwater vehicles and there is a lot of research in academia regarding system modeling. Details of the equations of motion for any marine vehicle were provided in [29], which comprehensively derived the equations of motion for underwater vehicles. In modeling the equations of motion for AUV a generalized six degree of freedom (DOF) is

considered. Furthermore, it is assumed that AUV is a rigid body and earth rotation is insignificant. Also, gravitational, hydrostatic, hydrodynamic and inertial are the forces that are applied to the AUV. The combination of these forces determines the behavior of the vehicle. The dynamic of AUV was categorized as follows:

- Kinematics: Studying the geometrical features of motion, without considering the forces that cause the motion
- Kinetics: Analyzing the forces affecting the motion.

In the following sections, AUV's coordinate frame and kinematics and dynamics will be explained [31].

2.1.2 Coordinate Systems

To determine the position and orientation of an AUV in a 3D environment, the vehicle's movement is studied in six degrees of freedom. Position and translation are determined by the first three coordinates, which are (x, y, z). As depicted in Figure 2.2 the next three elements (ϕ, θ, ψ) in the coordination vector determine the orientation of the AUV (according to the world frame). In marine science, the terminologies that are used for describing these features are surge, sway, heave, roll, pitch, and yaw, respectively. Clearly, the position, orientation, as well as linear and angular velocity of the vehicle, should be defined according to the reference coordinate system. For this purpose, orthogonal coordinate axes are joined to the center point (this will be determined in the next section) in the vehicle's body as a reference frame. In the same manner, the force and moments that will be applied to the AUV should also use the same reference frame [31].

The notation of [29, 32] is used for determining the coordination and according to the convention of the Society of Naval Architects and Marine Engineers (SNAME), the x-direction is longitudinal (along its length), the y-direction is transversal (perpendicular to its length), and z-direction is normal (from top to bottom).

Since the AUV we will use is robust in roll and pitch, then 4 DOF may satisfy our modeling requirements. However, to achieve a solution that can be applied to a wide range of AUVs and ROVs, we consider modeling in 6 DOF. Table 2.1 demonstrates the SNAME convention for describing the movement of marine crafts.

For using the kinematics equations of motion, understanding the coordinate frames which are used for representing the position and orientation is a crucial step in mathematical modeling. In this research, we are using two orthogonal reference frames as follows:

DOF	Description	Forces	Velocities	Positions
1	Movements in the x axis (surge)	X	u	x
2	Movements in the y axis (sway)	Y	v	y
3	Movements in the z axis (heave)	Z	w	z
4	Rotations about the x axis (roll)	K	p	ϕ
5	Rotations about the y axis (pitch)	L	q	θ
6	Rotations about the z axis (yaw)	M	r	ψ

Table 2.1 SNAME convention.

- World or Inertia frame: This is earth fixed frame XYZ which is defined in reference to the center of the earth as depicted in Figure 2.2. North, East, and Down (NED) are the three orthogonal axes of this frame. Hence a right-hand reference frame with unit vectors $\vec{I}, \vec{J}, \vec{K}$ is created. A vehicle's position in this earth-fixed frame has the following vector items:

$$r_{O'} = [X\vec{I} + Y\vec{J} + Z\vec{K}] \tag{2.1}$$

- A body reference frame $\mathrm{O}'_{\mathrm{xyz}}$, with unit vectors $\vec{i}, \vec{j}, \vec{k}$ and at origin of O′, is defined that is rotating and moving with AUV.

Computation of all vehicle body forces will be performed with respect to the origin O'. Moreover, two other critical points regarding marine vehicles are the center of gravity (CG i.e., mass) and the center of buoyancy (CB). As CG represents the vehicle's first-moment centroid, CB represents its first-moment centroid of volumetric displacement. In this research origin, O' is considered the same as the center of gravity [31]. According to the body origin frame, the position vector of the center of buoyancy is ρ_{B}:

$$\rho_{\mathrm{B}} = [\mathrm{x_B i} + \mathrm{y_B j} + \mathrm{z_B k}] \tag{2.2}$$

Euler Angles

When transforming from one coordinate frame to another, Euler's rotation theorem dictates that three successive rotations must be performed using Euler angles. For this reason, a rotation matrix is created by using three elementary rotations.

According to the world reference (earth-fixed), frame convention rotation of roll (ϕ), pitch (θ), and yaw (ψ) should be performed sequentially and in order.

Therefore, by determining an azimuth rotation ψ, a positive rotation over the body Z direction is performed. After that rotation θ, (positive up) about the new Y direction

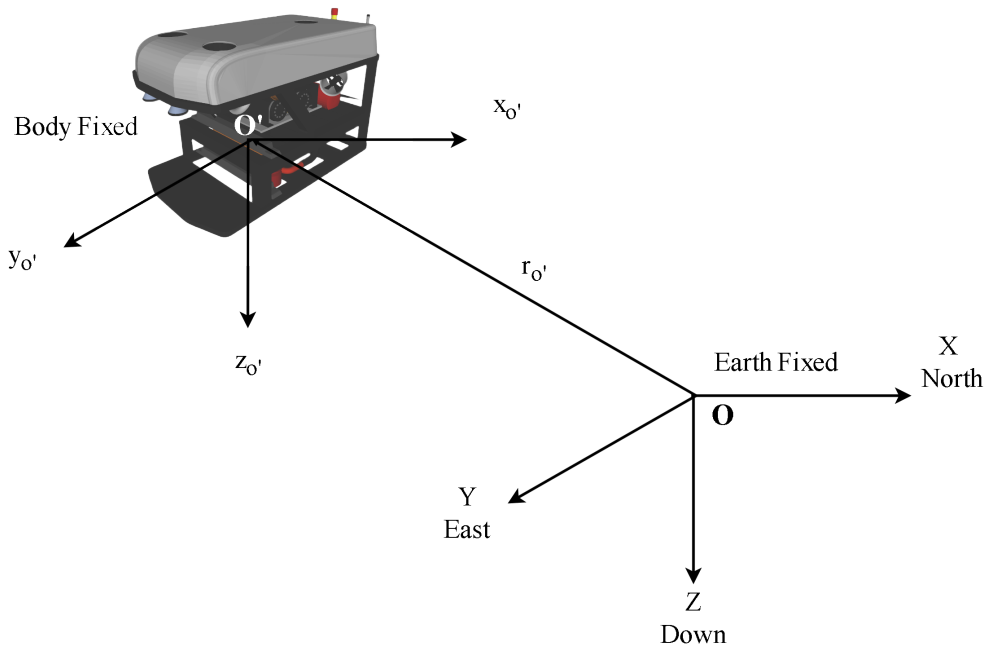


Figure 2.2 Earth fixed and body fixed coordinate frames.

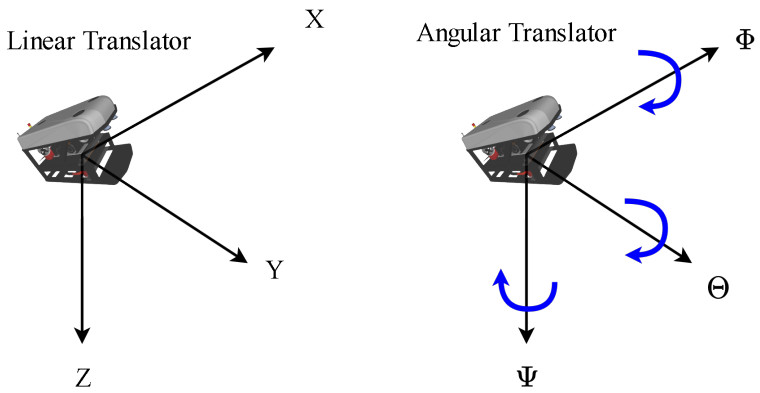


Figure 2.3 Body fixed coordinate system linear and angular velocity convention.

should be achieved and finally, positive rotation ϕ, about the new X direction should be performed. Figure 2.3 depicts the rotation and angular velocity of body fixed frame [31].

For example, the coordinate of vector, $r_o = [\mathrm{X}_o, \mathrm{Y}_o, \mathrm{Z}_o]$, in world frame if changes by ϕ angle then coordination of the new position $r_1 = [\mathrm{X}_1, \mathrm{Y}_1, \mathrm{Z}_1]$ in the new reference frame with respect to the world frame is:

$$Y_1 = Y_o \cos\phi + Z_o \sin\phi$$
$$Z_1 = -Y_o \sin\phi + Z_o \cos\phi$$

and $\mathrm{Z}_1 = \mathrm{Z}_\mathrm{o}$. The matrix form of this relation can be shown as:

$$r_1 = [R]^{-1}_{x_o,\phi} r_0$$

Here the rotation matrix $[R]$ is an orthogonal matrix, which means the transpose of [R] is equaled to its inverse.

$$[R]^T = [R]^{-1}$$

Multiplying any vector r_o by rotation matrix results in creating a similar vector in a new rotated coordinate. Total combined rotational transformation can be obtained by repeating the sequences of rotations.

$$[R] = [R]_{z_o,\psi}[R]_{y_o,\theta}[R]_{x_o,\phi} \tag{2.3}$$

Expanding (2.3) results in:

$$[R] = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix} \tag{2.4}$$

Applying matrix multiplication on (2.4) will provide:

$$[R] = \begin{bmatrix} \cos\psi\cos\theta & \cos\psi\sin\theta\sin\phi - \sin\psi c\phi & \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi \\ \sin\psi\cos\theta & \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{bmatrix} \tag{2.5}$$

Thus, any position vector in a rotating reference frame can be represented via the coordinate of the original reference by using the following relations,

$$r_{ijk} = [R]^{-1} r_{IJK} \tag{2.6}$$

Kinematics

Kinematics is studying the movement of an object without referencing the mass and force causing the motion. Linear and angular velocities of the vehicle (in the body frame) and their transformation to the world coordinate frame are considered in kinematics. The velocity of a vehicle in the world frame can be represented as:

$$\dot{r} = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} \tag{2.7}$$

Multiplication of the linear part of the body frame velocity vector by rotation matrix in (2.5), the translational velocities can be calculated:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = \begin{bmatrix} \ R \ \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{2.8}$$

Similarly, using world coordinate frame velocities we can calculate the body frame velocities by:

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} \ R \ \end{bmatrix}^T \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} \tag{2.9}$$

The rotation rates (Euler angle) in the world frame can be calculated using the rotation matrix in the body frame and applying non-orthogonal linear transformations as follows [29].

$$\begin{aligned} \dot{\phi} &= p + q\sin(\phi)\tan(\theta) + r\cos(\phi)\tan(\theta) \\ \dot{\theta} &= q\cos(\phi) - r\cos(\phi) \\ \dot{\psi} &= \tfrac{q\sin(\phi) + r\cos(\phi)}{\cos(\theta)} \end{aligned} \tag{2.10}$$

To represent in the matrix form a new transformation matrix from the body to the world frame can be used as:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = [T] \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2.11}$$

Here T is:

$$[T] = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & -\sin\phi/\cos\theta & \frac{\cos\phi}{\cos\theta} \end{bmatrix} \tag{2.12}$$

It should be noted that for the minor angular rotation we can consider the following assignments:

$$\begin{aligned} \dot{\phi} &= p \\ \dot{\theta} &= q \\ \dot{\psi} &= r \end{aligned} \tag{2.13}$$

It is worth noting that the rotation matrix, [T] is not orthogonal (unlike [R]) i.e. $[\mathrm{T}]^{-1} \neq [\mathrm{T}]^{\mathrm{T}}$. Inverting equation (2.12) and multiplying by the modification rate of Euler angles in the world frame, we can calculate the angular velocity vector as follows:

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = [T]^{-1} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{2.14}$$

$$[T]^{-1} = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{bmatrix} \tag{2.15}$$

Velocity values in the matrix notation form are as follows:

$$[V]_{body} = \begin{bmatrix} u \\ v \\ w \\ p \\ q \\ r \end{bmatrix} \tag{2.16}$$

$$[V]_{\text{earth}} = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{2.17}$$

Finally, a transformation from boy to world and vice versa using matrix form are represented by (2.18) and (2.19) respectively:

$$[V]_{earth} = \begin{bmatrix} [R] & 0 \\ 0 & [T] \end{bmatrix} \begin{bmatrix} V \end{bmatrix}_{\text{body}} \tag{2.18}$$

$$[V]_{body} = \begin{bmatrix} [R]^T & 0 \\ 0 & [T]^{-1} \end{bmatrix} \begin{bmatrix} V \end{bmatrix}_{\text{earth}} \tag{2.19}$$

To summarize kinematic relationships between velocities in the world frame using the body frame can be expressed by expanding the equation (2.18) and plugging the equations (2.5), (2.12) and (2.16) [31]. The result can be written as follows:

$$\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} u\cos\theta\sin\psi + v(-\cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi) + w(\sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi \\ u\cos\theta\sin\psi + v(\cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi) + w(-\sin\phi\cos\psi + \cos\phi\sin\theta\sin\psi) \\ -u\sin\theta + v\sin\phi\cos\theta + w\cos\phi\cos\theta \\ p + q\sin\phi\tan\theta + r\cos\phi\tan\theta \\ q\cos\phi - r\sin\phi \\ (q\sin\phi + r\cos\phi)/\cos\theta \end{bmatrix} \tag{2.20}$$

2.1.3 Equations of Motion

To model, the dynamics of the vehicle Newton's equations of motion will be used [29]. In the following equations, position and velocity vectors are represented by η and ν respectively. It should be noted that the position vector η is in inertia/world frame whereas, the velocity vector ν is in the body frame.

$$\dot{\eta} = J_{\Theta}(\eta)\nu \tag{2.21}$$

$$M\dot{\nu} + C(\nu)\nu + D(\nu)\nu + g(\eta) = \tau \tag{2.22}$$

In equation 2.22, τ is used for representing the control force and torque. Movement of the underwater vehicle can be represented by Fossen notation [29]

It is essential to transform the values between the body and Inertia frame using the following equation:

$$\dot{\eta} = \mathbf{J}_{\Theta}(\eta)\nu \tag{2.23}$$

Here $\mathbf{J}_{\Theta}(\eta)$ is the BODY-to-Inertia transformation matrix

Rigid body, hydrostatics, and hydrodynamics are the crucial components of any marine vehicle which will be modeled in the following sections [29].

2.1.4 Rigid Body Kinetics

The rigid body (a body in which the relative position of all its points is constant) kinetics can be expressed as:

$$\mathbf{M}_{RB}\dot{\nu} + \mathbf{C}_{RB}(\nu)\nu = \tau_{RB} \tag{2.24}$$

Where M_{RB} represents the matrix for the rigid body. The effect of rotation of the vehicle in the body frame with respect to Inertia is Coriolis and centripetal effect and is represented by C_{RB} in the above equation. Finally, the forces and moments (in the body frame) are represented by τ_{RB}. M_{RB} has a feature of uniqueness regarding the geometry of the vehicle. It is composed of the inertia of the vehicle, position of gravity, and mass. M_{RB} is the symmetric, definite and constant. Hence following conditions in (2.25) are valid for M_{RB}.

$$\mathbf{M}_{RB} = \mathbf{M}_{RB}^{\top} > 0, \dot{\mathbf{M}}_{RB} = \mathbf{0}_{6\times 6} \tag{2.25}$$

Based on [29], M_{RB} is described as:

$$\mathbf{M}_{RB} = \begin{bmatrix} m\mathbf{I}_{3\times 3} & -m\mathbf{S}\left(\mathbf{r}_g^b\right) \\ m\mathbf{S}\left(\mathbf{r}_g^b\right) & \mathbf{I}_b \end{bmatrix} \tag{2.26}$$

$$\mathbf{M}_{RB} = \begin{bmatrix} m & 0 & 0 & 0 & mz_g & -my_g \\ 0 & m & 0 & -mz_g & 0 & mx_g \\ 0 & 0 & m & my_g & -mx_g & 0 \\ 0 & -mz_g & my_g & I_x & -I_{xy} & -I_{xz} \\ mz_g & 0 & -mx_g & -I_{yx} & I_y & -I_{yz} \\ -my_g & mx_g & 0 & -I_{zx} & -I_{zy} & I_z \end{bmatrix} \tag{2.27}$$

Where an identity matrix, inertia matrix and the skew-symmetric matrix are represented by $\mathbf{I}_{3\times 3}$, $\mathbf{I}_b$ and $\mathbf{S}\left(\mathbf{r}_g^b\right)$ respectively. Also, m is the mass of the vehicle and $\mathbf{r}_g^b$ is the position of the center of gravity (CG) with respect to the center of origin (CO). Here $\mathbf{S}\left(\mathbf{r}_g^b\right)$ is the cross-product and can be defined as:

$$\mathbf{S}\left(\mathbf{r}_g^b\right) = \begin{bmatrix} 0 & -z_g & y_g \\ z_g & 0 & -x_g \\ -y_g & x_g & 0 \end{bmatrix} \tag{2.28}$$

and $\mathbf{r}_g^b = \begin{bmatrix} x_g & y_g & z_g \end{bmatrix}^{\top}$.

There are several ways for defining Coriolis and centripetal matrix $\mathbf{C_{RB}}(\nu)$. In this research according to [33] Lagrangian parameterization is used as follows.

$$\mathbf{C}_{RB}(\nu) = \begin{bmatrix} 0_{3\times 3} & -m\mathbf{S}\left(\nu_1\right) - m\mathbf{S}\left(\nu_2\right)\mathbf{S}\left(\mathbf{r}_g^b\right) \\ -m\mathbf{S}\left(\nu_1\right) + m\mathbf{S}\left(\mathbf{r}_g^b\right)\mathbf{S}\left(\nu_2\right) & -\mathbf{S}\left(\mathbf{I}_b\nu_2\right) \end{bmatrix} \tag{2.29}$$

Where, $\nu_1 = \mathbf{v}_{b/n}^b$, $\nu_2 = \mathbf{w}_{b/n}^b$.

2.1.5 Hydrostatics

From a hydrostatic perspective, the restoring forces are composed of the gravitational and buoyancy forces and it is similar to the spring force in a mass damper

spring system. The effect of restoring forces and moments are represented in the hydrostatic vector $\mathbf{g}(\eta)$ [29] and can be determined by the following vector:

$$\mathbf{g}(\eta) = \begin{bmatrix} (W-B)s(\theta) \\ -(W-B)c(\theta)s(\phi) \\ -(W-B)c(\theta)c(\phi) \\ -(y_gW - y_bB)\,c(\theta)c(\phi) + (z_gW - z_bB)\,c(\theta)s(\phi) \\ (z_gW - z_bB)\,s(\theta) + (x_gW - x_bB)\,c(\theta)c(\phi) \\ -(x_gW - x_bB)\,c(\theta)s(\phi) - (y_gW - y_bB)\,s(\theta) \end{bmatrix} \tag{2.30}$$

Where the weight of AUV is $W = mg$, the buoyancy force is $B = \rho g \nabla$. Here ∇ is the amount of volume displacement by AUV, the density of water is, ρ and g is the constant of gravity. The position of the center of gravity (CG) and center of buoyancy (CB) is represented by: $\mathbf{r}_g^b = \begin{bmatrix} x_g & y_g & z_g \end{bmatrix}^\top$ and $\mathbf{r}_b^b = \begin{bmatrix} x_b & y_b & z_b \end{bmatrix}^\top$ respectively.

2.1.6 Hydrodynamics

hydrodynamic forces and moments applied on AUV in 6 DOF according to [34] can be represented as:

$$\tau_{hyd} = -\mathbf{M}_A \dot{\nu}_r - \mathbf{C}_A(\nu_r)\,\nu_r - \mathbf{D}(\nu_r)\,\nu_r \tag{2.31}$$

Here added mass system inertia matrix is represented by $\mathbf{M}_A = \mathbf{M}_A^\top > 0 \in \mathbb{R}^{6\times 6}$. Coriolis-centripetal and damping matrices are also represented by $\mathbf{C}_A = -\mathbf{C}_A^\top \in \mathbb{R}^{6\times 6}$ and $\mathbf{D}(\nu) \in \mathbb{R}^{6\times 6}$ respectively.

In this research we assume that AUV operates in calm water; hence the added mass matrix $\mathbf{M}_A$ can be assumed approximately constant according to [29] can be defined as:

$$\mathbf{M}_A = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix} = \begin{bmatrix} X_{\dot{u}} & X_{\dot{v}} & X_{\dot{w}} & X_{\dot{p}} & X_{\dot{q}} & X_{\dot{r}} \\ Y_{\dot{u}} & Y_{\dot{v}} & Y_{\dot{w}} & Y_{\dot{p}} & Y_{\dot{q}} & Y_{\dot{r}} \\ Z_{\dot{u}} & Z_{\dot{v}} & Z_{\dot{w}} & Z_{\dot{p}} & Z_{\dot{q}} & Z_{\dot{r}} \\ K_{\dot{u}} & K_{\dot{v}} & K_{\dot{w}} & K_{\dot{p}} & K_{\dot{q}} & K_{\dot{r}} \\ M_{\dot{u}} & M_{\dot{v}} & M_{\dot{w}} & M_{\dot{p}} & M_{\dot{q}} & M_{\dot{r}} \\ N_{\dot{u}} & X_{\dot{v}} & N_{\dot{w}} & N_{\dot{p}} & N_{\dot{q}} & N_{\dot{r}} \end{bmatrix} \tag{2.32}$$

For specifying the coefficients in the matrix Society of Naval Architects and Marine

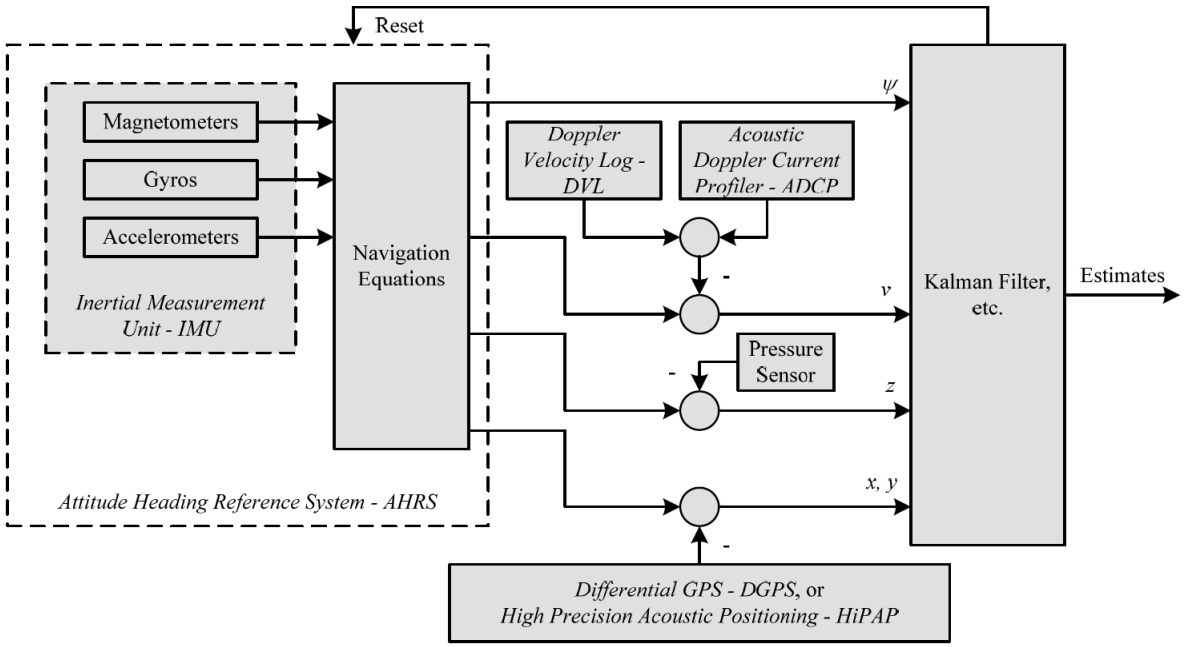


Figure 2.4 AUV general navigation architecture [35].

Engineers (SNAME) notation [32] is used. For example, $X_{\dot{\omega}}$ represents the applied added mass force X along the x-axis because of the fluid acceleration $\dot{\omega}$ in the z-direction.

$$\mathrm{C}_A = \begin{bmatrix} 0_{3\times 3} & -\mathrm{S}\left(\mathrm{A}_{11}\nu_1 + \mathrm{A}_{12}\nu_2\right) \\ -\mathrm{S}\left(\mathrm{A}_{11}\nu_1 + \mathrm{A}_{12}\nu_2\right) & -\mathrm{S}\left(\mathrm{A}_{21}\nu_1 + \mathrm{A}_{22}\nu_2\right) \end{bmatrix} \tag{2.33}$$

2.1.7 Damping

Damping for the underwater craft operating in 6 DOF, is highly nonlinear and separation of a different form of damping is a challenging task. Nevertheless, to model damping it is conventional to separate it into linear and nonlinear parts as follows:

$$\mathbf{D}\left(\nu_r\right) = \mathbf{D}_L + \mathbf{D}_{NL}\left(\nu_r\right) \tag{2.34}$$

Where $\mathbf{D}_L$ is the velocity-independent, linear damping terms, and $\mathrm{D}_N L\left(\nu_r\right)$ is the nonlinear part which is affected by changing the velocity of the vehicle.

2.2 Underwater Vehicles Navigation

The navigation of autonomous underwater vehicles is quite challenging as they lack access to GPS, as opposed to autonomous aerial and land vehicles. The severity of this problem also depends on the application. For example, navigation on military

and underwater maintenance missions is more significant than oceanographic surveys. This requires the use of several different navigational strategies when vehicles dive for missions [36]. Existing solutions for AUV navigation [37] are mainly based on dead reckoning and inertial navigation, acoustic navigation, and geophysical navigation. An Inertial Navigation System (INS) is the basis of the dead reckoning which incorporates the attitude heading reference system as well. In order to improve navigation, other methods, such as Doppler Velocity Log (DVL) for determining velocity based on earth reference frame and specifying depth using pressure sensors, should be included [35, 38].

The AUV's position is determined by transponder beacon messages in the acoustic positioning method. Mainly acoustic positioning can be done in two basic ways: long baseline (LBL) and ultra-short baseline (USBL). While the former requires the installation of transponders on the bottom of the ocean, the latter utilizes GPS-adjusted transponders installed on surface vehicles. The operating range of these methods is limited. For deep water, LBL operates at a distance of 10 km and this decreases to 4 km at low depth. Hence this method is suitable for fixed-position applications. A system based on USBL can cover a range of 500 m in shallow water [36]. Because USBL depends on surface vehicles, it is not suitable for military operations. Consequently, navigation sensors should be selected and configured based on the type of mission that an AUV is expected to perform. Figure 2.4 demonstrates the general navigation architecture of an AUV [35].

Traditionally, the Kalman Filter (KF) [39] is used to predict the status of a linear, Markovian, and Gaussian system. Thus, due to the nonlinearity of the AUV system, an Extended Kalman Filter (EKF) or particle filter (PF) [40] method should be applied to estimate navigation. WhileSimultaneous Localization and Mapping (SLAM) and Concurrent Mapping and Localization (CML) [41] are typically used for acoustic positioning, PF is particularly preferred for inertial positioning. Moreover, the navigation error of AUVs is extremely dependent on mission type and sensor characteristics, making it challenging to analyze such errors beforehand. Therefore, it's critical to develop and define high-fidelity simulations for the planned missions [35].

In overall, AUVs can take short-range missions, up to about 10km, with an adjusted INS for survey missions. During longer missions, the AUV trajectory has a major impact on navigation system precision. When the AUV returns to an area it has previously visited, a variety of geophysical techniques correct incremental inaccuracies.

Chapter 3

UNDERWATER COMMUNICATION

Compared to terrestrial communication, underwater communication has several distinctive characteristics that make it completely different from overseas communication, such as salinity, temperature, pressure, water currents, winds, etc. These phenomena result in several issues, including high attenuation, scattering, multipath fading, and absorption of waves, which make underwater communications quite challenging [42]. For example, high frequency sound waves are strongly absorbed by seawater, which severely limits the available bandwidth.

Underwater wireless transmissions [42] can be carried out using two leading technologies including:

- **Acoustic communication**: Sound waves are the most common method of transmitting data through the water. There are two types of acoustic communication: low-frequency sound waves and high-frequency sound waves. While low-frequency acoustic can travel long distances, high-frequency acoustic has a higher data rate but a shorter transmission distance. In addition to the Doppler effect and significant propagation delay, acoustic waves are highly influenced by intersymbol interference [43].
- **Radio-frequency (RF) communication**: This method uses electromagnetic waves to transmit data through the water. Despite its high data throughput at short distances, this method suffers from a mild Doppler effect. In water, electromagnetism waves are attenuated heavily, thus restricting the data rate and range of signals.

In this chapter, we review the communication technologies and protocols considered

for communicating AUVs for the networked control system.

3.1 Acoustic Communication

Undoubtedly one of the harshest environments for data transmission is the underwater acoustic communication channel. For long distances, the optimal channel capacity is less than 50 kbps for Signal to Noise Ratio (SNR) of 20 dB by using an ordinary modem with a data rate of less than 10 kbps [44]. A contemporary commercial underwater acoustic modem, like EvoLogics $S2CR48/78$ at the range of 1000 m, can offer a data rate of up to 31.2 kbps. Channel frequency, chemical and physical properties of the water, and the shape of the environment affect the acoustic propagation in water. Spreading loss and absorption loss are two main reasons for a path loss on the acoustic channel [44]. The expanding area that acoustic wave covers while it propagates from the transmitter is the reason for the spreading loss as given in (3.1) as follows:

$$PL_{sp}(r) = k\ 10log(r), \tag{3.1}$$

where k represents the spreading factor and r is the distance.

In contrast, absorption loss is due to losing signal in the form of thermal energy because of friction and ionic relaxation of the acoustic signal propagation from a projector to hydrophone in the water as given in (3.2) as follows:

$$PL_{ab}(r,f) = 10log(\alpha(f))r, \tag{3.2}$$

where absorption coefficient, α is affected by the characteristics of the water and considering the frequency of an acoustic wave (i.e. f) it can be calculated using Thorp's expression as:

$$\alpha(f) = \frac{0.11f^2}{1+f^2} + \frac{44f^2}{4100+f^2} + 2.75\times10^{-4}f^2 + 0.0033, \tag{3.3}$$

where f is the frequency of the acoustic signal in KHz. Adding the spreading and absorption losses the total path loss can be determined as:

$$PL(r,f) = PL_{sp}(r) + PL_{ab}(r,f). \tag{3.4}$$

Before applying the fading we subtract the path loss from the transmitted signal

$P_t(dB)$ to determine the received power in dB as:

$$P_r(dB) = P_t(dB) - PL. \tag{3.5}$$

Multipath and noise are the main obstacles in acoustic communication. We assume flat fading in this work, where all multipath components arrive at the receiver with similar delays. Therefore, Rayleigh fading is adopted to model the variations due to multipath. Thus the received power is exponentially distributed, where the power level calculated in (3.5), converted to watts is the mean of the exponential distribution. To distinguish the received signal correctly, and consider an acceptable SNR ratio the received power should be above a threshold value.

3.2 RF Communication

Although RF signal has a high data rate and high propagation speed, it severely suffers from high path loss in underwater communication. For a 10 MHz signal, in a freshwater environment, the maximum achievable data rate is about 3 Mbps [7]. The path loss for the RF channel is highly affected by the conductivity, permittivity and permeability of water. Neglecting the air-water boundary loss, the path loss can be calculated [45] as:

$$PL = RE(\gamma)\frac{20}{ln(10)}r, \tag{3.6}$$

here RE reflects real part, r denotes distance; γ is propagation constant given by:

$$\gamma = j\omega\sqrt{(\mu\epsilon - j\frac{\sigma\mu}{\omega}}. \tag{3.7}$$

Here ω is the frequency in rad/s, ϵ denotes the total permittivity of water, μ stands for the permeability of free space, and σ represents water's conductivity [45]. Finally, the total path loss equation can be obtained by using (3.6) and (3.7) as follows:

$$PL = RE(j\omega\sqrt{(\mu\epsilon - j\frac{\sigma\mu}{\omega})}\frac{20}{ln(10)}r. \tag{3.8}$$

The average received signal power is calculated by considering the above path loss formulation for the RF channel, and Rayleigh fading is applied as exponentially distributed power, as in the acoustic channel, followed by the calculated SNR.

3.3 Full Duplex Communication

It has been believed for a long time that half-duplex data transmission is the only possible method for wireless communication systems [46]. The main principle of half duplex is separating uplinks and downlinks by assigning a designated time slot in the Time Division Duplex (TDD), or by assigning separate frequency bands in the Frequency Division Duplex (FDD), or assigning a unique code to each node to share the same frequency, as in Code Division Multiple Access (CDMA) to share the same frequency band among several users. Due to this separation, a node's own transmission will not interfere with the received data. This is referred to as self-interference (SI). In-band full duplex (IBFD) communication allows for simultaneous transmission and reception over the same frequency band, which has the potential to double the link throughput and spectral efficiency. As the demand for faster communication links grows, employing FD is one of the candidate technologies for future wireless communication systems, providing that the SI is canceled or suppressed down to a low level, allowing the receiver can detect the incoming signal [47–49]. There are numerous potential applications for FD powered communications, in the domain of defense, security, etc., which require more network capacity [50].

By transmitting and receiving data simultaneously in the same frequency band, the FD communication mode can also maximize the capacity of acoustic links [51,52]. A node's own transmission is expected to produce a high level of SI, which can exceed the noise floor by 100 dB in certain communication scenarios. Consequently, the system's performance is affected by a reduction in residual SI concerning the signal to noise ratio (SNR) of the desired signal [53].

Therefore it is essential to suppress the SI signal substantially, down to the noise level, to enable in-band FD communication. Without any modification to the specifications of available solutions, such as connecting FD microwave, cellphone adaptive duplexer, etc., SI cancellation can be employed [54]. The SI cancellation method can further improve next generation wireless communication networks by enabling virtualization of the spectrum, multiple division duplexing methods, and improved interference management.

Figure 3.1 illustrates the three stages of full duplex SI cancellation, namely passive antenna suppression (AS), analog cancellation (AC), and digital SI cancellation (DSIC). Analog cancellation operates at the RF level to cancel the SI using further circuitry, whereas DSIC is employed at the baseband level.

When two antennas are used in wireless FD communication, AS helps to cancel passive suppression. Besides incorporating a circulator or duplexer, antenna separation

permits SI cancellation on a single antenna. Signals should initially be distorted at the transmitter antenna, the same modification should be applied over the channel and on the receiver side prior to obtaining SI in the receiver node [55].

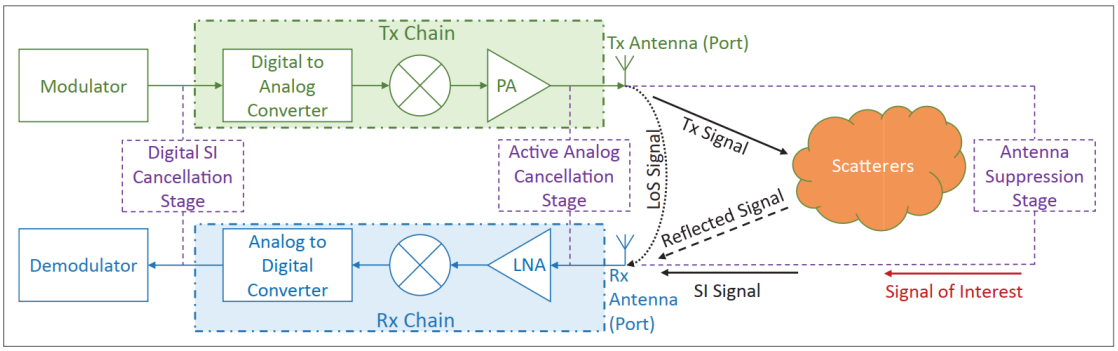


Figure 3.1 Main components of the FD wireless communication system [55].

Antenna SI Cancellation: By adequately designing RX/TX antenna systems, the purpose of antenna SI-cancellation techniques is to minimize the impact of SI on receiving antennas. SI-cancellation can be achieved by embedding antennas at the receiving antennas or by separating, polarizing, and isolating antennas through directional antennas. One of the above approaches can be used depending on the physical limitations and application type. For example, while passive suppression using antenna separation and isolation drastically declines SI signal in a relay system, it is not applicable to employ that method in mobile applications due to size restrictions. For these applications, a method based on passive suppression of SI is described for a small antenna separation by the polarization that can achieve 60 dB passive suppression. Physically separated antennas for TX and RX or a single antenna capable of performing TX and RX at the same time are the two main approaches for designing an FD transceiver.

RF SI Cancellation The RF SI-cancellation can be performed by estimating the received SI signal and removing it from the received signal; hence prior to entering into LNA and ADC, the SI is suppressed. In Figure 3.1, the RF transmitted signal is obtained at the TX chain and PA output and subtracted from the received signal at the RF SI cancellation stage. RE SI cancellation comprises two steps, namely analog RF SI-cancellation and digital RF SI-cancellation. In the initial phase of SI cancellation, the internal coupling and reflections are suppressed by analog RF SI cancellation, while additional random external reflections can be handled by digital RF SI cancellation.

Basedband SI Cancellation: Through the use of various signal processing algorithms on the received signal, SI cancellation in the baseband is designed to decrease the remaining SI after the ADC. Hence, before entering the ADC, reducing SI as

much as possible in the two previous stages (i.e., antenna SI cancellation and RF SI cancellation) is critical for achieving successful SI cancellation in the baseband. As the signal has been converted to digital mode, advanced digital processing can be applied to it. All details that change the transmitted signal, such as channel characteristics and RF components' nonlinearities, should be determined before subtracting the received residual SI. Therefore, for reproducing a precise copy of the received SI signal, an estimation of the SI channel and the characteristics of the transceiver is required.

Because of the low frequency signal of underwater acoustic communication, high precision Analog to Digital Converters (ADC)s can achieve better digital SI cancellation results compared to over water RF communication. Hence, in the literature, several studies have attempted to improve the performance of digital SI cancellation in underwater FD acoustic communication using low-complexity recursive least-squares (RLS) adaptive filters or digitalized power amplifiers [53].

Chapter 4

UNDERWATER HYBRID NETWORKED CONTROL SYSTEM

Offshore operations require a team of diverse professionals involved in a challenging and hazardous environment. Additionally, the reliance on human labor to achieve these missions represents a serious threat to the future growth and development of the maritime industry. Replacing manned underwater vehicles with unmanned underwater vehicles eliminates the danger to divers and allows the operation to be carried out without involving humans and their failures. For this, it is necessary to establish an efficient and reliable communications network among offshore control centers, AUVs, ROVs, and supporting ships.

In this chapter, the system model and network scenario for the proposed underwater hybrid NCS are introduced. This is followed by the proposed node models, which implement different communication modes and protocols for data flow and two types of controllers in the closed loop of underwater hybrid NCS.

4.1 Proposed Hybrid Networked Controlled System (NCS)

In our underwater scenario, shown in Figure 4.1, we consider the docking maneuver application, where a docking station (DS), located at the bottom of the ocean, provides a safe place for AUVs to dock, charge their batteries and transmit data to offshore centers via a high-speed data link and AUVs are steered towards the DS via the proposed hybrid NCS. The docking station is equipped with sensors,

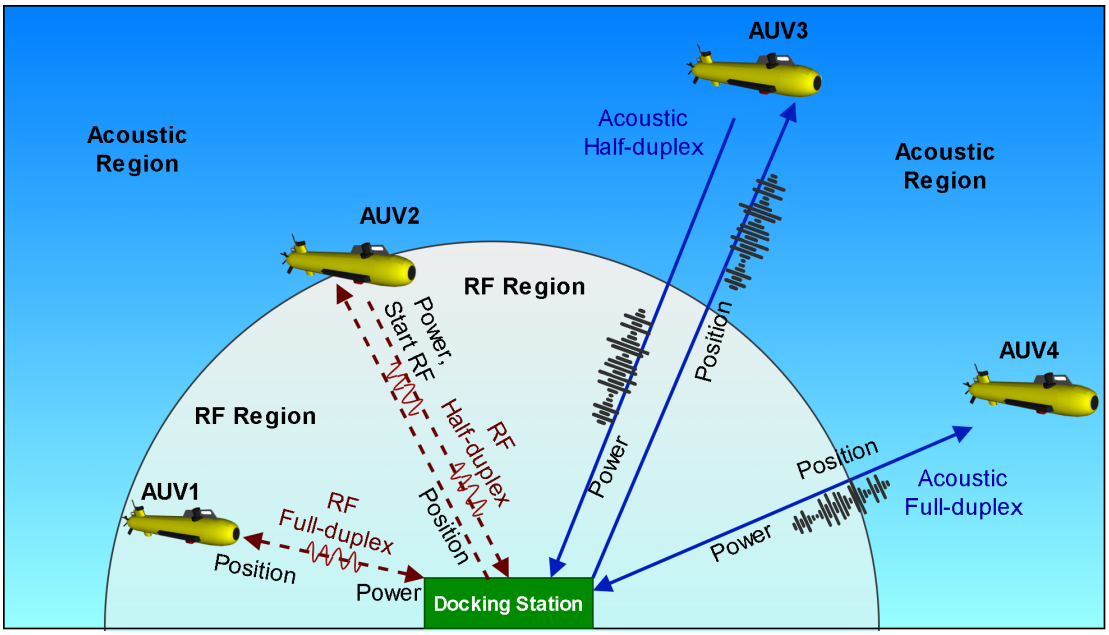


Figure 4.1 An overview of our proposed underwater hybrid full duplex NCS for AUVs' docking maneuver.

measuring the position and velocity of the AUVs as they are too bulky or expensive to be installed on the vehicles. All measurements are periodically sent to the AUVs over a reliable communication channel so that they can calculate their routes to the docking station and perform docking successfully. The proposed underwater hybrid networked control system involves the following components:

- Docking station with devices that can measure the position of the AUVs,
- Communication channel (Acoustic or RF) that is selected based on the distance of the AUVs to the docking station, which is used to send the measurements to the AUVs,
- FD or HD transmission modes,
- Communication protocol to exchange position information with multiple AUVs at the same time,
- Control algorithms to control the AUVs' thrusters based on the measured position and goal position information.

The block diagram for the proposed underwater hybrid NCS implementing docking maneuver is illustrated in Figure 4.2. AUV and Docking station blocks represent the two critical components of our system. The entire procedure is conducted in a closed-loop manner, beginning with the docking station's remote measurement of the AUVs' telemetry data, such as position, orientation, and linear and angular velocities, due to the absence of a navigation system on most of the AUVs. Based on the operation

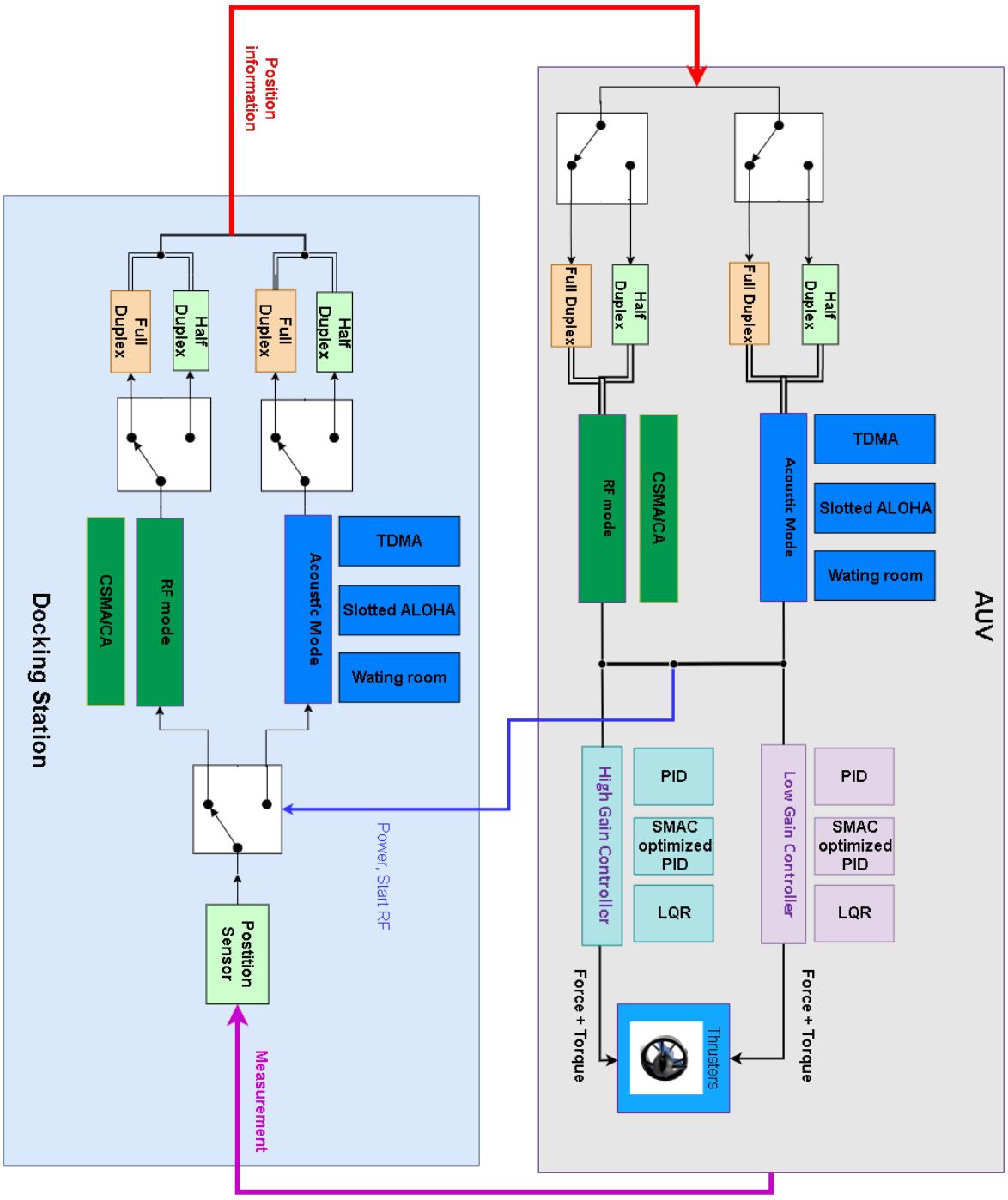


Figure 4.2 Hybrid underwater networked control system node view: AUV and docking station. Red connection is the uplink, the blue connection is the downlink and the purple connection is the measurement of the AUV by the docking station.

range of each communication link, the obtained data will be transmitted using either acoustic or RF modes. While for RF mode, conventional Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) is employed for the acoustic mode in our NCS, three different protocols, namely Time Division Multiple Access (TDMA), Slotted ALOHA (S-ALOHA) and Waiting Room (WR) protocol, are adopted.

Additionally, FD mode is proposed for simultaneous transmission and reception over the same channel to maximize channel capacity. Utilizing the FD communication mode in combination with the hybrid approach yields quicker transmission of position data to the AUV. When the message arrives at AUV, in accordance with the transmitter's preferred mode of communication, the corresponding mode of communication is used to receive data.

As illustrated in Figure 4.2, DS initiates the flow of information to signal the AUV to approach and dock by sending information via RF mode. The position information is transmitted using FD mode allows for rapid control. When the AUV receives the signal via the RF receiver, it switches to RF mode and inspects the control field of the packet to determine the availability of FD communication mode. The AUV sets the communication mode to FD, activates the high gain controller and uses a selected controller alternative (i.e., prior to mission start) to generate force and torque for AUV steering. This loop is iterated, with the transmission of the AUV's updated position measured through the DS's position sensor at each sampling time.

Three types of controllers are adapted for actuating and pursuing docking: a conventional PID, SMAC optimized PID and LQR. The experimental circumstances and the mode of communication determine the controller selection. While conventional PIDs can be employed in idle scenarios (e.g., calm water), when severe circumstances (i.e., high currents) prevail, the other two controllers are required to be used in order to produce forces and torques.

At each sampling time, the calculated forces and torques provide inputs to thrusters blocks that distribute the inputs among thrusters available to rotate the propellers. The AUV transmits messages, such as power, start RF, etc., either in FD or HD mode.

4.2 Communication Protocols for Hybrid NCS

AUVs can communicate effectively and efficiently through communication protocols regardless of the type of hardware and software employed. In order to accomplish this, standards and rules need to be defined. Once these rules and standards are followed, nodes can communicate and exchange data efficiently over a network. To

use the channel effectively and to have reliable communication, it is essential to coordinate transmission in the channel, which is done through Medium Access Control (MAC) protocols.

In the implementation of the MAC protocols in the proposed NCS, packets carry the control information between the docking station and multiple AUVs. The packet transmitted from the docking station has a length of 512 bits, including fields such as receiver ID, position, reference position, and time stamp. On the reverse link, the packet sent from the AUVs has 64 bits involving the node ID and data fields.

In the following subsections, we present the power control mechanism and MAC protocols for the hybrid acoustic and RF underwater networked control system as well as the corresponding channel models.

4.2.1 Medium Access Control (MAC) Schemes for Acoustic Mode

We consider two different periods for the frame time of the MAC protocol in acoustic mode, as depicted in Figure 4.3. While the docking station uses the first downstream frame time to broadcast the message to the AUVs, the second period is used by AUVs as an upstream channel that we propose to implement one of TDMA, Slotted ALOHA (S-ALOHA) and waiting room (WR) protocols as shown in Figures 4.3.(a), 4.3.(b) and 4.3.(c). To perform docking maneuver successfully in the NCS, the position information of all AUVs needs to be transmitted periodically. Therefore, the TDMA scheme is assigned for the downstream channel to transmit messages to the AUVs sequentially. Through these messages, while an intended AUV captures its location information to implement its move towards the docking station, all other AUVs can also receive the same packets and learn about the other AUVs' location, which helps them to avoid physical collisions among the vehicles. To ensure that all packets from AUVs are received correctly, a short delay time equal to the maximum propagation delay is added at the beginning of the docking station's period.

Frame Time for Acoustic Mode

Propagation delay is the main reason for the long delay in acoustic communication. This delay is the source of phase delay and it should be incorporated carefully in the distributed design of the control loop of NCS. According to the maximum number of supported vehicles (i.e., V_{MAX}), the frame time for AUVs and docking station is designed. The length of a message, M_{DS}, and the data rate of the acoustic modem, R_{AC}, are also considered in calculating frame time so that the slot time for the docking station is found as $DS_{SL} = M_{DS}/R_{AC}$ and total frame time of docking

station is calculated as $TF_{DS} = DS_{SL} * V_{MAX}$. The number of bits in the message M_V and propagation delay of acoustic signal v_{AC} should be considered in calculating the AUVs' slot times and total frame time, i.e., TF_{AUV}. Hence each AUV's slot time can be obtained as $T_{SV} = M_V/R_{AC} + d/v_{AC}$, where d is the distance from the docking station. Multiplying T_{SV} by the maximum number of vehicles, V_{MAX}, determines the total AUV frame time. Finally, a summation of AUV and docking station frame times will give us the MAC protocol's total frame time as follows: $T_F = TF_{DS} + TF_{AUV}$.

Time Division Multiple Access (TDMA) Protocol

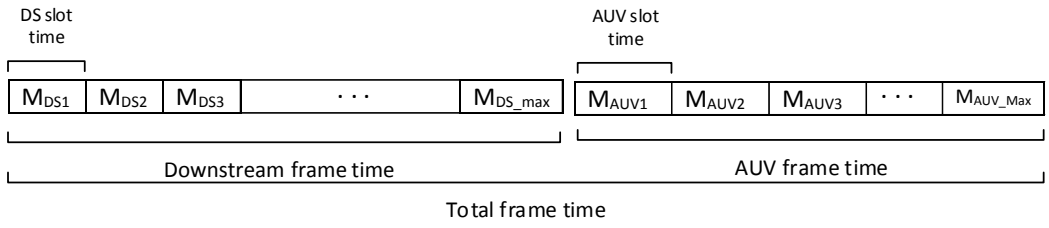
In the TDMA scheme, the communication channel is partitioned into several time slots, with only one node (i.e., docking station or AUV) permitted to transmit or receive at any given time. Based on the frame time calculated in our customized protocol implementation, the allocated time slot for each node circulates periodically. In this manner, nodes can only send and receive messages at specific intervals. In contrast with Frequency Division Multiple Access (FDMA) schemes, TDMA uses a buffer-and-burst approach for data transfer, which results in discrete data transfers. In general, frames consist of three segments: a preamble, an information message, and a tail segment [56]. Without loss of generality, in order to avoid transmitting extra data, the preamble and tail segments of the frame are omitted in this thesis. Hence, for each vehicle, a separate duration equal to the AUV slot time for transmitting using the acoustic upstream channel to the docking station is allocated. The TDMA frame time is depicted in Figure 4.3.(a). In this scheme total number of slots is determined as $N = V_{MAX}$.

Slotted ALOHA Protocol

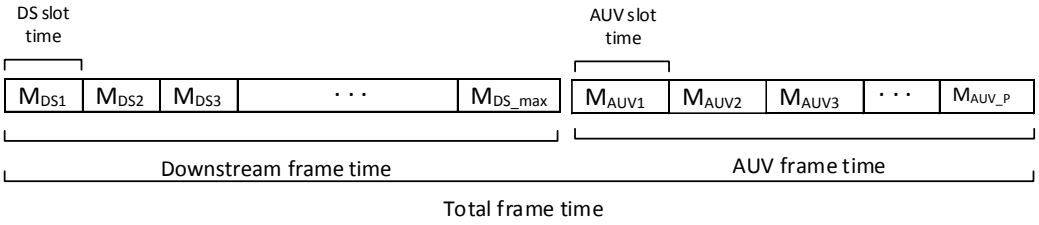
The upstream AUV period in S-ALOHA starts after waiting for the guard time of propagation delay, following the downstream period. The guard time is added to ensure that the last AUV has received the packet from the docking station and to synchronize all AUVs. In this scheme, some percentage of propagation delay should be considered in calculating AUV slot time [57]. Therefore 70% of additional transmission time and propagation delay is assigned for AUV slot time. The total number of slots is set to P and $P \leq N$, where N is the total number of AUVs.

Waiting Room Protocol

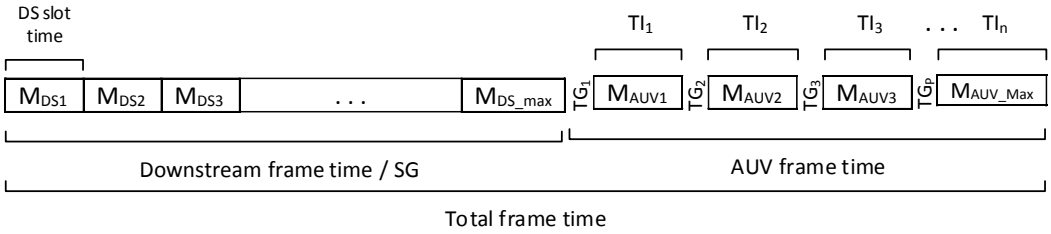
In this scheme, a different Terminal Gap (TG) value is assigned for each AUV with transmission interests. Furthermore, each AUV has a Transmit Interval (TI), which is the limit for the AUV transmission duration. A Synchronization Gap (SG) time



(a) TDMA upstream



(b) S-ALOHA upstream



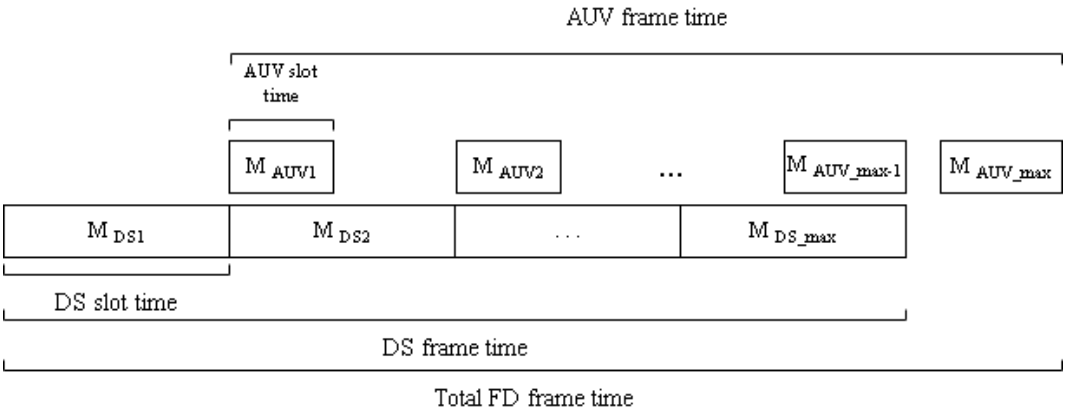
(c) WR upstream

Figure 4.3 Frame structure and MAC schemes for acoustic mode

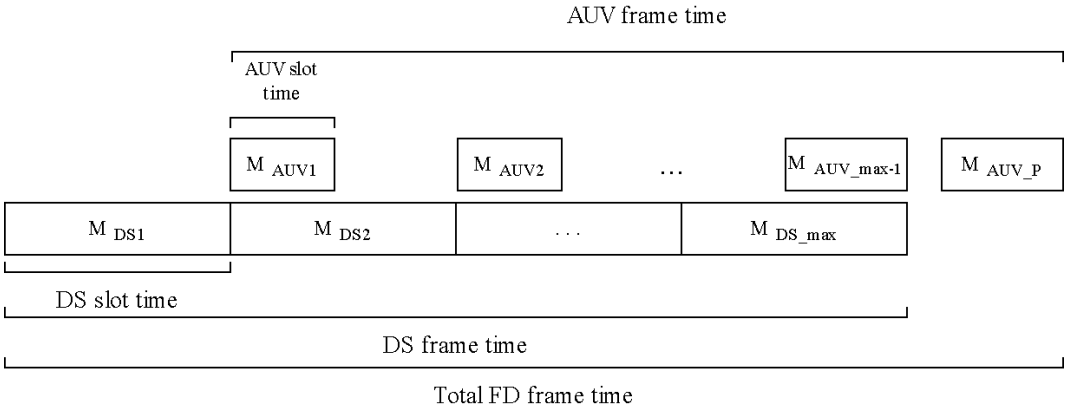
is also set for all of the nodes. The protocol works as follows: During the SG, there are no transmissions on the network and the nodes commit for transmissions in the current frame. At the end of SG, each node starts a countdown timer of TI_i and waits for it to expire. The node with the shortest TI expires first and it starts transmission. All other nodes pause their timers until the end of a transmission. The process repeats until all committed nodes complete their transmissions, and no more transmissions start, which signals the SG of the next frame Figure 4.3.(c). In our case, SG coincides with the downstream period, where the docking station transmits.

Full Duplex (FD) TDMA Protocol

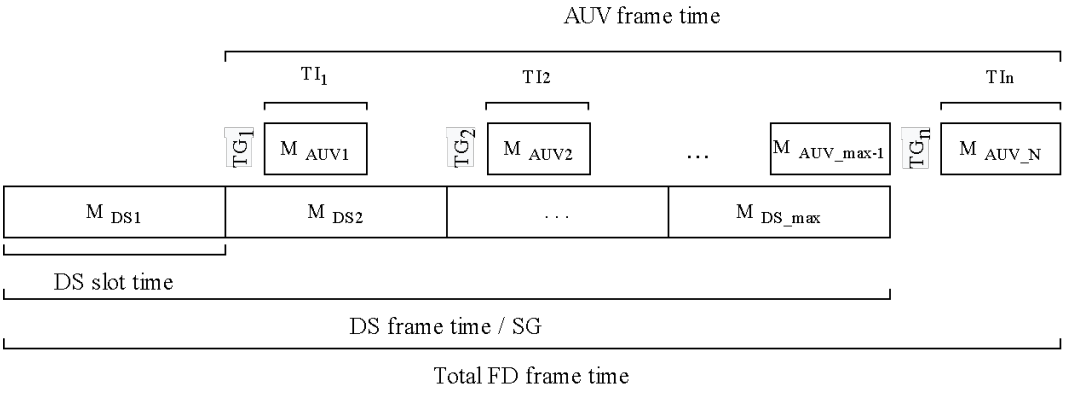
In the FD TDMA, a similar process is employed as in HD TDMA, where the frame time is divided into two segments. While the first segment is allocated to the docking station for transmitting messages to the vehicles, the second segment is assigned to the vehicles for transmitting messages back to the docking station. Separating the transmissions into two segments and assigning a specific time slot to each allows TDMA-based protocols to prevent collisions and ensure fair access to the shared



(a) FD TDMA



(b) FD S-ALOHA



(c) FD WR

Figure 4.4 FD enabled frame structure and MAC schemes for acoustic mode

communication channel for all vehicles. With FD TDMA, docking station and AUVs can share bandwidth efficiently and communicate simultaneously without interfering with each other. Figure 4.4a illustrates how the AUV frame time can overlap with the docking station frame time, resulting in a frame time of half that of HD TDMA. Nevertheless, to allow DS to publish a message first and then receive a replay from the corresponding AUV, there should be one slot time delay (i.e., M_{DS1}) at the beginning of the overlapping as well as one additional AUV slot time at the end.

Full Duplex Slotted ALOHA Protocol

A schedule-based network program cannot work efficiently when network technology changes frequently, numerous nodes attempt to access the channel, and data is generated rapidly and unpredictably. Alternatively, when there are large numbers of nodes in a network, random access protocols manage the shared channel more efficiently than schedule-based protocols. Therefore, as a representative of random access based protocols, we have chosen to utilize slotted ALOHA, which we have modified to work in FD mode. Node transmission needs to be coordinated appropriately; otherwise, channel efficiency will decline severely. When transmission occurs outside planned slots, channel efficiency cannot be maximized, and the received messages cannot be decoded due to collusion, resulting in reduced throughput [58]. Due to the docking station's periodic transmission of position information, there is always downstream traffic. However, given the fact that there are only a few AUVs in the NCS, and AUVs occasionally send data to the DS, there is a low probability of a collision. Based on Figure 4.4b, the frame time for FD S-ALOHA is equal to the FD TDMA frame time due to the overlap between AUV and DS frames. Thus, AUVs should wait for one DS slot time for transmission, allowing the DS message to propagate to the AUVs. During the new time slot, if the AUVs need to transmit a message, they can access the channel and transmit it to the DS. Therefore, an additional AUV time slot should be considered at the end of the total frame time for completing the last AUV transmission.

Full Duplex Waiting Room Protocol

After sending a message, each node must wait for a specified period before sending another message. The three intervals in this protocol do not prevent other nodes from accessing the communication channel. All nodes in the waiting room share a Synchronization Gap (SG) interval. This interval controls the sequence of entrances to the waiting room. Terminal Gaps (TG) control channel access and are different for each node. This parameter should be larger than propagation delays. Transmit Interval (TI) allows the host not to monopolize the channel and it is the same for all

nodes. In order to use this protocol in FD mode, we should consider the total frame time to be equal to the synchronization gap since this will allow both transmission and reception to coincide at each sampling time as depicted in Figure 4.4c. However, since TG is present in this protocol, WR's total FD frame time is slightly higher than FD TDMA and FD S-ALOHA.

4.2.2 MAC Scheme for RF Mode

Since the RF channel has high data rate and small propagation delay, we have chosen to use the Carrier Sense Multiple Access(CSMA) type access for both downstream and upstream channels in RF mode. In CSMA, the node with data to send first senses the medium and sends after confirming that there are no on-going transmissions. If two transmissions coincide, a collision occurs, and the nodes wait for a random amount of time and try again. The docking station sends its messages periodically. For the AUV transmissions, a small number of AUVs share the upstream channel via CSMA. Since only the AUV nodes that are within RF range of the docking station operate in RF mode (hence at the late part of the docking maneuver), the number of contending AUVs is typically only one. Furthermore, since the data rate is high, packet transmission durations are short. Therefore, the probability of collisions is low, making CSMA efficient during RF mode.

Full Duplex MAC Scheme for RF Mode

In general, the main characteristic of FD is its ability to maximize data rate and spectral efficiency. Even though some studies demonstrated that FD could provide a twofold gain over HD, this gain could deviate significantly due to spatial reuse, and asynchronous contention effects in dense networks [59–61].

Several studies have been conducted in the literature to design FD MAC protocols due to their importance. As part of developing a prototype including a backoff algorithm to improve fairness, an FD MAC protocol, called ContraFlow, is proposed in [62], which was evaluated for small-size networks. A distributed FD MAC, known as FD-MAC [63], was designed and implemented on the WARP software defined radio (SDR) boards [64], which implements virtual contention resolving and shared random backoff in the proposed protocol.

The FD mode has been considered for RF communication in the implemented NCS, which has been inspired by the Synchronized Contention Window Full Duplex (S-CW FD) MAC protocol. Aside from supporting bidirectional and relaying FD scenarios, the S-CS FD MAC protocol also supports HD communication mode and is compatible with IEEE 802.11 [61]. The reason for selecting this protocol is that

it is easy to implement and can be utilized rapidly for relaying and bidirectional communication.

It is necessary to synchronize the transmission time of the two nodes participating in the FD communication to operate in FD mode. This is accomplished by sending packets to each other, which include a new field within the control field of a packet to indicate the duration of the backoff. Following a successful HD transmission, two nodes are synchronized to transmit in FD mode simultaneously. Details regarding timing and protocol can be found in [61].

4.3 Control Algorithms for Hybrid NCS

PID

We consider two controllers via PID and LQR approaches to steer the AUV and implement docking. PID controllers are particularly useful when the mathematical model of the system is difficult to obtain using analytical design methods. Different design techniques can be used to specify controller parameters as long as a mathematical model of the closed-loop system can be determined. However, for adjusting PID gains, the experimental approach can be used if analytical or computational methods cannot be used due to the complexity of the model [24].

The following equation describes a general equation for a PID controller.

$$u(t) = K\left(e(t) + \frac{1}{T_i}\int^t e(s)ds + T_d\frac{de(t)}{dt}\right) \tag{4.1}$$

Essentially, PID aims to continuously maintain setpoints by tuning control parameters. PID works according to the measurement of error e, which is the difference between command signals u, and plant output y. Here, a proportional gain is set by K, a time integration time is set by T_i, and a time derivative time is set by T_d [65].

Depending on the communication mode and protocol selected in our proposed NCS, the delay of the network changes. As a result, adjusting the gains for the PID is necessary. The manual tuning of the system is not feasible due to the fact that the tuning may be altered to favor a specific communication protocol or mode of operation. In the absence of such a controller, the results of the NCS may be skewed and unfair comparisons can be made.

SMAC Optimized PID

This method aims to optimize the PID gains of NCS for different communication frameworks by iteratively tuning the gains until certain performance criteria are

satisfied. This ensures that the results are fair and unbiased and allows a more reliable evaluation of the performance of different communication frameworks under the same conditions. Therefore, this leads to selecting the suitable communication framework for use in NCS. In fact, SMAC attempts to fit the model iteratively by adjusting the parameters and evaluating the fit of the model. Afterward, the parameters for the next iteration are selected based on the fit of the model [26].

Linear Quadratic Regulator (LQR)

The SMAC optimized PID has the disadvantage that it needs to be adjusted according to environmental changes. Therefore, underwater vehicles require a controller that can handle nonlinearity and uncertainty. Hence, we employ an LQR controller, one of the most widely used controllers in linear control systems. This controller provides a systematic way of obtaining control gains in closed-loop feedback systems [24, 27].

Basically, LQR is a mathematical control design technique that is used in the design of linear time-invariant (LTI) controllers. LQR attempts to minimize a cost function by finding a feedback control gain. An example of a cost function would be to sum the squared error between the desired and the actual output. Furthermore, LQR can be applied by defining an LTI system and its state-space representation. Due to the nonlinearity of the mathematical model of the AUV, linearization is required in order to represent the system in state space. Additionally, it is necessary to define a quadratic cost function for the input states and errors. Riccati equation is used to minimize the cost function. The following chapter will discuss the modeling and linearization of the AUV.

In this chapter, we have discussed the importance of applying a networked control system that implements our proposed multi-mode communication system to improve the performance of the NCS. Additionally, it should also be noted that different protocols and communication modes have different frame times for data transmission. Due to this variable time delay, traditional controller tuning methods cannot be used with the hybrid NCS. It is necessary to adjust the controller gains in accordance with network protocol parameters, such as sampling time, etc. Consequently, the controller has to be automatically adjusted to provide optimal results, which adds another feature to the system. This imposes another feature on the system that the controller should be adjusted automatically to provide optimized results. Our tuning method in upcoming chapters involves optimizing discrete time PID controllers using SMAC and adopting and implementing optimizing LQR controllers.

Chapter 5

AUV CONTROLLER DESIGN

One of the main differences in the various communication protocols is that their frame time is different from each other according to the nature of the algorithm. This will reflect on the sampling time of the system. The shorter sampling time results in a more precise and smoother movement of the vehicle. However, this requires that for each of the protocols, the gains of the controller have to be adjusted manually in order to perform stable docking maneuver operations. Furthermore, during the experiment, it is observed that decreasing the sampling time even by half has been even less effective in comparison with changing the gain of the controller. It is important to have an auto-tuning method so that for each protocol, the controller gains will be tuned to the same performance requirements.

Besides, it is required to design a more robust controller for different network protocols, and for that, it is essential to determine the mathematical modeling of the AUV. Accordingly, different forces and moments that will be applied to the vehicle should be determined. Also, different coordinate systems for the underwater vehicle as well as their transformation to each other should be specified. Therefore, in the following sections, the mathematical model of the system is provided.

5.1 AUV Controllers

A closed-loop PID-type control system can improve AUVs' maneuverability and performance in the presence of environmental disturbances. Data from sensors and navigation systems is used to provide feedback to a closed-loop control system. Moreover, it is common for traditional autopilot designs of AUVs to use multiple PID controllers, each controlling one DOF.

PID controllers were traditionally employed for controlling AUVs. However, due

to the sensitivity of derivative terms to measurement noise and the challenge of detecting (and estimating) the linear and angular velocity, most AUV/ROV systems employed simple P or PI controllers. Because the AUV system to be controlled exhibits extremely nonlinear, it should be considered that the adoption of the PID controller neither guarantees system stability nor provide optimal control of the system. A decoupling control design is one of the practical approaches for controlling AUVs. In this case, the equation of motion in 6 DOF is separated into three steerings, speed and diving sub-controllers. For example, each of the subsystems in [23] was controlled by several PID controllers. Depth and steering control are the two fundamental duties of AUVs. To achieve these duties, several control systems have been used; each has benefits and drawbacks. In general, algorithms can be divided into two categories: linear algorithms and nonlinear algorithms [66].

5.1.1 Linear AUV Controllers

A linear model of a system is determined by utilizing the system behavior in a particular scenario (cruising speed, angle of attack. etc,). Having simplified the model results in a more straightforward control design but restricts UUV from operating under certain circumstances due to neglecting the nonlinear feature of the model. The PID controller stated earlier is an example of a linear controller because each proportion, integration, and differentiation component of PID is mathematically linear. As an example, a separated PD controller for navigating UUV is used in [67].

Methods exist for auto-tuning the gains of the PID controller. Although traditional methods such as Ziegler Nichols [24] are popular, modern equivalents also deserve attention, such as the SMAC to optimize the gains. This method involves determining a cost function by iteratively configuring the parameters to fit the model and examining the model's fitting aims to obtain optimal gains [26].

Other popular linear controllers include the linear quadratic regulator (LQR). Utilizing the state feedback control method, which uses a quadratic cost function to minimize the cost function, this method provides optimal controller gains. Similarly, the Linear Quadratic Gaussian (LQG) method is another linear control methodology that is ideal for uncertain linear systems perturbed by:

- Additive white Gaussian noise
- Uncertainty of states information. (i.e., either state information is not available

or affected by noise which is not applicable to a closed feedback loop).

There is also additive white Gaussian noise and quadratic cost that disturb the state information. Some works, such as [68], use this method to control underwater vehicles.

5.1.2 Nonlinear AUV Controllers

Nonlinear control methods have been employed in the literature for specific challenges and particular unmanned vehicles in some research institutes. A Sliding Mode Controller (SMC) is a robust controller used when there is uncertainty in the state information.

The nominal control of SMC is still assumed to be linear, even though it is a nonlinear control method (such as [23] and [69]). Furthermore, SMC is also demonstrated in [70] using a simplified nonlinear model of the vehicle for the nominal control. A chattering effect, however, is a characteristic of SMC that may lead to the discovery of unmodeled high-frequency modes. In this situation, System performance is degraded by these modes, and it is even possible for the system to become unstable during these modes. In addition to decreased fin lifespans, chattering leads to an increase in electricity consumption. Nevertheless, the trajectory control of ROVs has recently been proposed with a chatter-free SMC [71], [72].

Other approaches have later been proposed, which use a fully nonlinear model. For example, the Lyapunov and backstepping techniques are used extensively in [66]. A conventional Lyapunov-based guidance system with PI type controller [73]was used to overcome both measurement bias and unmodeled kinematic interactions between a UUV and its environment. Furthermore, a nonlinear adaptive controller for diving control of an AUV is suggested in [74] using the traditional backstepping algorithm. A method called Higher Order Sliding Mode (HOSM) is proposed by [66] in order to eliminate chattering and improve control performance. In [75], a nonlinear output-feedback control technique based on HOSM is used to control an underwater vehicle prototype equipped with hydro-jet propulsion.

Research has been pursuing novel approaches to the underwater vehicle control problem because of its challenging nature. Recent studies have focused on intelligent and adaptive control methods. The current state of the art uses neural networks, fuzzy reasoning, and hybrid methods in this field. Neural networks are powerful tools because of their ability to estimate nonlinear mathematical functions. Additionally, neural networks can be trained to adapt to changing input-output relationships in

different scenarios. As a result, neural networks may help control unknown nonlinear systems, such as AUVs. Besides their ability to handle nonlinearity, neural networks also possess several other characteristics that make them appropriate for control applications [76].

- Parallel structure: Neural networks are characterized by the parallel structure of their implementation, which facilitates the construction of parallel control systems. As a result, neural network-based processing systems are robust and fast.
- Compatibility with hardware: Several integrated circuits (ICs) are available on the market that can be used to implement Artificial Neural Networks (ANN).
- Multivariable nature: Due to their ability to map functions with many inputs and outputs, neural networks are particularly suitable for controlling multivariable systems.

In the past, several neural network controller schemes have been proposed and implemented [76], some of which have been designed especially for underwater vehicle control:

1.1 Identification and modeling:

- Direct Inverse Modeling
- Forward Modeling
- Indirect Inverse Modeling

1.2 Direct control:

- Predictive Control
- Model Reference Control
- Supervised Control
- Critic Control
- Direct Inverse Control
- Internal Model Control

Control systems can be implemented using neural networks due to the simplicity of offline learning. Moreover, a neural network controller's speed is adequate because it is trained beforehand (similar to tuning a conventional controller). The controller achieves a rapid response during runtime without any weight adjustments being

made. The controller produced by this algorithm is not adaptive, and changes in system parameters will likely result in a poor controller. The use of a controller that continuously updates the neural network weights is an appealing alternative to offline training. For the performance evaluation of a neural network, the controller is initially established, and the controller weights are then modified to enhance the system performance, mainly by decreasing output errors. In order to use this approach, the optimal weight changes must be calculated based on the system inputs and outputs, as well as the reference trajectory, in order to maintain the stability of the system.

A considerable amount of literature indicates that most network controllers for AUVs are direct controllers, forming the majority of the control system. AUV neural network controllers that are trained offline and are not adaptive have been described in [77, 78].

A fuzzy logic controller is another type of controller developed and used to control different scenarios robustly. In order to design a fuzzy controller, it is necessary to understand the dynamics of the controlled system. It has been shown that fuzzy logic control can be used to control the motion of underwater vehicles [44-46], and an example of applying a sliding mode fuzzy logic control to control is presented in [79].

5.2 Proportional Integral Derivative (PID) Controller

For steering the AUV and damping its response, we have implemented a PID controller for controlling the orthogonal axis of the AUV during its cruise. Pitch and roll control run as local control loops based on onboard sensors, depth, forward motion and yaw is controlled based on the position data received over the network, and lateral movement is uncontrolled and is governed by the hydrodynamic forces. A discrete-time approximation of the control loops [65] is implemented as described in (5.1)-(5.3) below:

$$
\begin{aligned}
e(kt_s) &= r(kt_s) - y(kt_s) && (5.1)\\
P(kt_s) &= K_p\ e(kt_s) && (5.2)\\
D(kt_s) &= \frac{T_d}{T_d + Nt_s} Dt_s(k-1) - \frac{KT_dN}{T_d + Nt_s}(y(kt_s) - yt_s(k-1)) && (5.3)\\
It_s(k+1) &= I(kt_s) + \frac{Kt_s}{T_i} e(kt_s) && (5.4)\\
u(kt_s) &= P(kt_s) + D(kt_s) + I(kt_s) && (5.5)
\end{aligned}
$$

where $r(kt_s)$, $y(kt_s)$ are the desired and measured values of the control at k^{th} multiple of the sampling period t_s, and their difference $e(kt_s)$ represents the position and orientation error. The proportional, derivative, and integral terms of the PID controller are calculated separately using Tustin's approximation [65] in equation (5.2), (5.3), and (5.4). Moreover, the gains of each term of PID controller (i.e., proportional gain K_p, derivative gain KT_d, and integral gain $\frac{K}{T_i}$) at each NCS loop and the sampling period t_s are determined periodically based on the communication mode and protocols. Separate onboard controllers for acoustic and RF links calculate the control input for each axis of the AUV. Receiving position information from the docking station, the AUVs compute the control signal $u(kt_s)$ using equation (5.5) for commanding thrusters.

5.3 Linear Quadratic Regulator (LQR) Controller

Linear Quadratic Regulator (LQR) is one of the most extensively used controllers in linear control systems. This controller provides a systematic way of obtaining control gains in closed-loop feedback systems [24, 27]. LQR can calculate optimal controller gains by assuming a linear transfer function, specifying a quadratic cost function, and setting the reference to zero. Therefore, obtaining a linear transfer function or equivalent state space model is a prerequisite for utilizing LQR effectively.

The system that we want to control should be in state-space representation. Thus suppose we have a system in the following state-space form.

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{5.6}$$

Here the system states variable is represented by $\mathbf{x}$ and $\mathbf{u}$ is the control input vector. The $\mathbf{A}$ and $\mathbf{B}$ matrices are used for introducing physical and control dynamics to the system model respectively.

Our goal is to obtain K, a gain matrix of the optimal control

$$\mathbf{u(t)} = -\mathbf{Kx(t)} \tag{5.7}$$

LQR aims to minimize the weighted sum of the control inputs and energy of the states according to the following cost function [30]:

$$J = \int_0^\infty (\mathrm{x}^\intercal \mathrm{Q}\mathrm{x} + \mathrm{u}^\intercal \mathrm{R}\mathrm{u})dt \tag{5.8}$$

Here **Q** and **R** are positive-definite (or positive-semidefinite) matrices. The amount of energy that is consumed for controlling a plant can be adjusted by the second term of the (5.8). The calculated control value using (5.7) is optimal, thus if some of the unknown parameters of the matrix **K** are calculated to minimize the cost function then we can conclude that $\mathbf{u(t)} = -\mathbf{Kx(t)}$ is optimal for any initial state. A closed-loop feedback controller block diagram is depicted in figure 5.1.

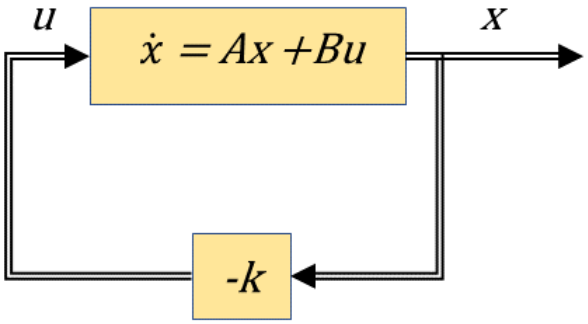


Figure 5.1 Optimal regulator system

To solve the optimization problem by plugging (5.7) to the (5.6) following can be derived.

$$\dot{\mathbf{x}} = \mathbf{Ax} - \mathbf{BKx} = (\mathbf{A} - \mathbf{BK})\mathbf{x} \tag{5.9}$$

Suppose that the matrix $\mathbf{A} - \mathbf{BK}$ is stable, that is the eigenvalues of $\mathbf{A} - \mathbf{BK}$ have negative real parts. Hence plugging (5.7) into (5.8) provides

$$\begin{aligned} J &= \int_0^\infty (\mathrm{x}^\intercal \mathrm{Q}\mathrm{x} + \mathrm{x}^\intercal\ \mathrm{K}^\intercal \mathrm{R}\mathrm{K}\mathrm{x})\, dt \\ &= \int_0^\infty \mathrm{x}^\intercal (\mathrm{Q} + \mathbf{K}^\intercal \mathrm{R}\mathrm{K})\, \mathrm{x} dt \end{aligned} \tag{5.10}$$

substitution following part

$$\mathbf{x}^\intercal (\mathbf{Q} + \mathbf{K}^\intercal \mathbf{R}\mathbf{K})\, \mathbf{x} = -\frac{d}{dt}(\mathbf{x}^\intercal \mathbf{P}\mathbf{x}) \tag{5.11}$$

where $\mathbf{P}$ is a positive-definite matrix. We can get the following equation.

$$\mathbf{x}^\intercal(\mathbf{Q}+\mathbf{K}^\intercal\mathbf{R}\mathbf{K})\mathbf{x} = -\dot{\mathbf{x}}^\intercal\mathbf{P}\mathbf{x} - \mathbf{x}^\intercal\mathbf{P}\dot{\mathbf{x}} = -\mathbf{x}^\intercal[(\mathbf{A}-\mathbf{B}\mathbf{K})^\intercal\mathbf{P}+\mathbf{P}(\mathbf{A}-\mathbf{B}\mathbf{K})]\mathbf{x} \quad (5.12)$$

Checking two sides of (5.12) and considering that it should be held for any value of $\mathbf{x}$, the following equation can be written:

$$(\mathbf{A}-\mathbf{B}\mathbf{K})^\intercal\mathbf{P}+\mathbf{P}(\mathbf{A}-\mathbf{B}\mathbf{K}) = -(\mathbf{Q}+\mathbf{K}^\intercal\mathbf{R}\mathbf{K}) \quad (5.13)$$

Here $\mathbf{A}-\mathbf{B}\mathbf{K}$ is a stable matrix [24], hence there exists a positive-definite matrix $\mathbf{P}$ that satisfies (5.13).

Therefore, we need to find the $\mathbf{P}$ from (5.13) and check if it is positive definite. It should be noted that more than one $\mathbf{P}$ which satisfies the equation may exist. Also, it should be considered that there exists one positive-definite matrix $\mathbf{P}$ to satisfy this equation if the system is stable.

The cost function $\mathbf{J}$ can be calculated as

$$J = \int_0^\infty \mathbf{x}^\intercal(\mathbf{Q}+\mathbf{K}^\intercal\mathbf{R}\mathbf{K})\mathbf{x}dt = -\mathbf{x}^\intercal\mathbf{P}\mathbf{x}|_0^\infty = -\mathbf{x}^\intercal(\infty)\mathbf{P}\mathbf{x}(\infty)+\mathbf{x}^\intercal(0)\mathbf{P}\mathbf{x}(0) \quad (5.14)$$

We suppose $\mathbf{x}(\infty)\to\mathbf{0}$, due to fact that all eigenvalues of $\mathbf{A}-\mathbf{B}\mathbf{K}$ are considered to have negative real parts; thus we can get

$$J = \mathbf{x}^\intercal(0)\mathbf{P}\mathbf{x}(0) \quad (5.15)$$

As a result, the cost function $\mathbf{J}$ can be derived in terms of the initial condition $\mathbf{x}(0)$ and $\mathbf{P}$.

We assumed that $\mathbf{R}$ is a positive-definite matrix; hence to find the solution of LQR we can write

$$\mathbf{R} = \mathbf{T}^\intercal\mathbf{T}$$

Considering that $\mathbf{T}$ is a nonsingular matrix, (5.13) can be modified as following

$$(\mathbf{A}^\intercal-\mathbf{K}^\intercal\mathbf{B}^\intercal)\mathbf{P}+\mathbf{P}(\mathbf{A}-\mathbf{B}\mathbf{K})+\mathbf{Q}+\mathbf{K}^\intercal\mathbf{T}^\intercal\mathbf{T}\mathbf{K} = \mathbf{0} \quad (5.16)$$

This can also be rewritten as

$$\mathbf{A}^\intercal\mathbf{P}+\mathbf{P}\mathbf{A}+\left[\mathbf{T}\mathbf{K}-(\mathbf{T}^\intercal)^{-1}\mathbf{B}^\intercal\mathbf{P}\right]^\intercal\left[\mathbf{T}\mathbf{K}-(\mathbf{T}^\intercal)^{-1}\mathbf{B}^\intercal\mathbf{P}\right]-\mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\intercal\mathbf{P}+\mathbf{Q}=\mathbf{0} \tag{5.17}$$

To minimize the cost function $\mathbf{J}$ according to the $\mathbf{K}$ gains following equation needs to be minimized.

$$\mathbf{x}^\intercal\left[\mathbf{T}\mathbf{K}-(\mathbf{T}^\intercal)^{-1}\mathbf{B}^\intercal\mathbf{P}\right]^\intercal\left[\mathbf{T}\mathbf{K}-(\mathbf{T}^\intercal)^{-1}\mathbf{B}^\intercal\mathbf{P}\right]\mathbf{x} \tag{5.18}$$

It should be noted that minimum appears when the last expression (nonnegative) is either equal to zero or

$$\mathbf{T}\mathbf{K}=(\mathbf{T}^\intercal)^{-1}\mathbf{B}^\intercal\mathbf{P}$$

Therefore,

$$\mathbf{K}=\mathbf{T}^{-1}(\mathbf{T}^\intercal)^{-1}\mathbf{B}^\intercal\mathbf{P}=\mathbf{R}^{-1}\mathbf{B}^\intercal\mathbf{P} \tag{5.19}$$

Equation (5.19) provides the optimal gain $\mathbf{K}$. Therefore, the optimal controller to the LQR using the cost function in (5.10) is linear and can be determined by

$$\mathbf{u}(t)=-\mathbf{K}\mathbf{x}(t)=-\mathbf{R}^{-1}\mathbf{B}^\intercal\mathbf{P}\mathbf{x}(t) \tag{5.20}$$

The matrix $\mathbf{P}$ in (5.18) either should be hold in (5.13) or in the following equation:

$$\mathbf{A}^\intercal\mathbf{P}+\mathbf{P}\mathbf{A}-\mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\intercal\mathbf{P}+\mathbf{Q}=\mathbf{0} \tag{5.21}$$

Equation (5.21) is known as the reduced-matrix Riccati equation. Hence for designing a controller using Riccati equation following steps should be performed [24].

1. Solve equation (5.21) for the matrix $\mathbf{P}$.

2. Substitute $\mathbf{P}$ matrix into (5.19) for obtaining the optimal $\mathbf{K}$ gains.

5.3.1 Adopting LQR Controller for the AUVs

Based on the dynamics model developed for an autonomous underwater vehicle (AUV), we consider a basic LQR control simulation in MATLAB. The AUV controller controls simultaneously in six degrees of freedom. The following provides a brief overview of the controller's model. The following section provides a brief overview of the model that is used in our LQR controller.

5.3.2 State-Space Model

A class of models called state-space models utilize state variables as the basis for describing a system by a collection of first-order differential equations or difference equations instead of the set of nth-order differential equations or difference equations. Furthermore, if a set of first-order differential equations is linear with respect to state and input variables, the model can be described as a linear state space model. Hence to represent the dynamic and physics obtained by the movement of a vehicle and the state-space model can be used as follows:

$$\dot{x} = Ax + Bu \tag{5.22}$$

By calculating the optimal thrust output, u, based on the vehicle's current state, x, the LQR control system aims to reach a target state. A matrix represents the relationship between the internal states of the system, and underlying vehicle physics, whereas the B matrix describes how inputs are introduced into the system or, more precisely which states are affected. Besides $\dot{x}$ is the derivative of x.

Here the outputs of the system are represented by vector Y. In this regard, it is important to note that the outputs may not always reflect the state variables. Matrix C here demonstrates how the states are combined to produce the outputs, and D is the matrix that allows the inputs to bypass the system without affecting the outputs and be fed forward to the outputs. There are a total of 12 elements in the vehicle's overall state, x, which is made up of the following:
x : The x position of the AUV
y: The y position of the AUV
z: The z position of the AUV
ϕ: The vehicle's roll (i.e. the rotation about the x-axis)
θ: The vehicle's pitch (i.e. the rotation about the y-axis)
ψ: The vehicle's yaw (i.e. the rotation about the z-axis)
u :The x-component of the vehicle's velocity
v: The y-component of the vehicle's velocity
w: The z-component of the vehicle's velocity
p: The vehicle's angular velocity in the *roll* direction
q: The vehicle's angular velocity in the *pitch* direction
r: The vehicle's angular velocity in the *yaw* direction

NOTE: the x value represents both the x-coordinate of the vehicle's position as well as its overall state. In this work, the symbol represents the overall condition of the

vehicle, unless otherwise specified.

5.3.3 Modeling LQR by Derivation

For the dynamics model, a vehicle-like vector model of the marine craft was used (i.e., the A and B matrices). Several matrices describe a vehicle's physical behavior. Matrices are determined by at least one of the 12 states. This vehicle's model to be used by the LQR controller was derived using the following references.

The following section provides detailed explanations of how the vehicle's LQR model is derived.

5.3.4 Non-Linear Dynamics Model

Underwater vehicles can be modeled using non-linear dynamics as follows:

$$Mv + C(v)v + D(v)v + G(n) = \tau \tag{5.23}$$

Where,
M is the mass/inertia matrix
C is the coriolis matrix
D is the damping matrix
G is the gravity/buoyancy matrix
τ is the external forces/moments matrix
μ is the velocity vector
η is the pose vector
$\dot{\mu}$ is the derivative of mu with respect to time
Various important characteristics of vehicles, such as mass and drag coefficients, are incorporated into these matrices.

5.3.5 State-Space Representation of Non-Linear Dynamics Model

Here we have $x = [\eta\ \mu]^T$ as a 12-element state vector consisting of η, which is the 6-element pose vector, and μ, which is the 6-element velocity vector (In both cases, six DOF are represented)

The transformation matrix $J(\eta)$ represents the vehicle's pose from BODY coordinates to NED. According to the state-space representation, the non-linear dynamics model can be stated as follows:

$$
\begin{aligned}
&F = Ax + Bu = f(x,t) + g(u,t) \\
&f(x,t) = \begin{bmatrix} 0_{6x6} & J(\eta) \\ 0_{6x6} & -M^{-1}(C-D) \end{bmatrix} \begin{bmatrix} \eta \\ v \end{bmatrix} + \begin{bmatrix} 0_{6x1} \\ -M^{-1}G \end{bmatrix} \\
&g(u,t) = \begin{bmatrix} 0_{6x1} \\ -M^{-1}\tau \end{bmatrix}
\end{aligned} \tag{5.24}
$$

τ refers to the thruster dynamics, which can be calculated as follows:

$$
\begin{aligned}
\tau &= Tu \\
u &= f_t \bar{u} \\
\bar{u} &= \delta|\delta|
\end{aligned} \tag{5.25}
$$

In this model, T represents the thrust allocation matrix (where each thruster's configuration is specified). Besides, the thrust effort (%) is converted to thrust force (N) by f_t, and the δ represents the raw thrust value.

5.3.6 Jacobian Linearization

In order to use LQR for non-linear models, the non-linear state space model must be linearized.
Assume that F is a non-linear state-space model, u is the control output vector, and x is the state vector. The linear A and B matrices of non-linear dynamics can be determined through *Jacobian Linearization*.

$$
\begin{aligned}
A &= \frac{\partial F}{\partial x} \\
B &= \left.\frac{\partial F}{\partial u}\right|_{u=[11111]}
\end{aligned} \tag{5.26}
$$

The A matrix varies with the vehicles' state, even though the B matrix remains constant. Consequently, linearization can be performed in real-time with respect to the current state. Hence, based on the system's current state, the A matrix is regenerated in real-time. Our linearized state-space model now looks like this:

$$
\dot{x} = Ax + Bu \tag{5.27}
$$

5.3.7 Linear Time Invariant AUV Equations of Motion

In order to use the LQR controller, we should check whether the system is controllable or not. In case of satisfaction of this requirement, we can build the state of the system anywhere we like considering the physical limitation of the system. To verify that the system states are controllable, its corresponding controllability matrix should have rank **n**, i.e., the number of independent columns/rows. Equation 5.28 represents the general form of controllability matrix $\mathcal{C}$, where **n** represents the number of states in the system.

$$\begin{aligned} &\dot{x} = Ax + Bu \\ &\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -M^{-1}[C+D] & -M^{-1}G \\ J & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} M^{-1} \\ 0 \end{bmatrix} u \end{aligned} \tag{5.28}$$

We expect a value of 12 for the rank since our controllability is a matrix of 12×12. In this way can check if our system is appropriate for applying the LQR controller for obtaining optimal gains **K**. In the LQR method, the control effort (**u**) and error from a reference point (i.e., zero in LQR) can be controlled by using **R** and **Q** Matrices in minimizing the cost function. For adjusting the parameters, initially, **R** is set to one, and $C'C$ should be assigned for the **Q**. As a result, we can set equal significance on control and state variables. Hence for controlling the outputs the non-zero value in **Q** can be changed for obtaining desired values [80].

5.3.8 LQR Cost Matrices

Once the state-space model has been linearized, the final step is to provide the cost matrices for Q and R. Essentially, these matrices are diagonal matrices representing either a state cost (Q matrix) or a thrust cost (R matrix). A Q matrix is a matrix with a dimension $n \times n$, where n is the number of elements in the state. Similarly, m is the number of thrusters in the diagonal matrix R. Through trial and error, the values for these elements were determined experimentally. LQR control system tuning and calibration are based on Q and R matrices.

After determining the four required matrices (i.e., A, B, Q, and R), the LQR can be used inside a loop to generate the gains and thrusters output as follows:

Utilizing the MATLAB *lqr* function, K can be obtained

$$K = lqr(A, B, Q, R) \tag{5.29}$$

Then the raw control output in thrust-force (N) is generated:

$$\overline{du}_{lqr} = -K\left(x - x_{\text{target}}\right) \tag{5.30}$$

After that, it is transformed to thrust-effort (%) using f_t^{-1} the inverse of the conversion factor:

$$du_{lqr} = f_t^{-1}\overline{du}_{lqr} \tag{5.31}$$

5.3.9 Feedforward Force

Since gravity matrices are neither functions of state nor of control output, the linearization of the non-linear model eliminated the Gravity matrix term. For this reason, the control output was modified to include a feedforward force representing gravity and buoyancy forces. By applying this force, the vehicle is able to counteract the effects of gravity and buoyancy. Putting the feedforward force into practice looks like this:

$$du_{feedforward} = f_t^{-1}[T * G] \tag{5.32}$$

Here T is the thrust allocation and G represents gravity matrices. Moreover, the G matrix is used to calculate gravity and buoyancy forces. The thrust allocation matrix is then used to allocate the forces to the appropriate thrusters. A thrust effort is then calculated based on the thrust force. Hence, by calculating the feedforward thrust, the final thrust output consists of both thrust components:

$$du_{feedforward} = du_{lqr} - du_{feedforward} \tag{5.33}$$

Finally, a thruster controller converts the thrust effort to RPM upon receiving the final du value.

Using the matrices value related to the vehicle dynamic explained in the previous section, we can populate these parameters into the vehicle configuration script in MATLAB. It should be noted that in order to obtain accurate values for these matrices, expensive experimental setups need to be done. However, the values of these Matrices can be approximated by making some simplifying assumptions.

5.4 Hyperparameter Optimization

Most machine learning methods require different parameters, learning rates, weights, or constraints to be tuned in order to generalize diverse data patterns. These values, commonly referred to as hyperparameters, should be tuned so that the ML algorithm can solve the problem effectively. The process of selecting a collection of ideal hyperparameters for adjusting or training ML algorithms is known as hyperparameter optimization. As a result, a hyperparameter is a value used to tune the learning of the ML model. Therefore, hyperparameter optimization identifies a set of hyperparameters that results in an optimum model by minimizing a cost function. The objective function returns the corresponding loss when a set of hyperparameters is used [81]. Hyperparameter optimization can be implemented in several ways, which are discussed in the following sections:

5.4.1 Grid Search

Grid searches refer to the process of searching through a portion of the ML's hyperparameter space that the user has selected for search. Performance metrics determined through cross-validation on the training set are used for directing a grid search. Before running a grid search, determining the value and specifying boundaries are essential steps because parameter space may include unbounded or real value [82].

5.4.2 Random Search

Random search operates by selecting randomly among entire possibilities instated of performing a comprehensive search. This approach can be applied both in discrete and continuous spaces. Random search provides better results than the grid search in the case of a limited number of hyperparameters [83].

Although grid search is simple to implement and suitable for parallel execution, considering the dimension of hyperparameters and increasing the number of trails, it increases exponentially. Hence it is inefficient for the enormous hyperparameter space.

To improve the grid search performance before applying grid search, some methods, such as calculating the likelihood of each point in the grid and determining a new grid based on the most significant likelihood, were proposed to improve the grid search performance [84]. Using this alleviation, the new grid's size is halved and repeated until the results reach a local minimum.

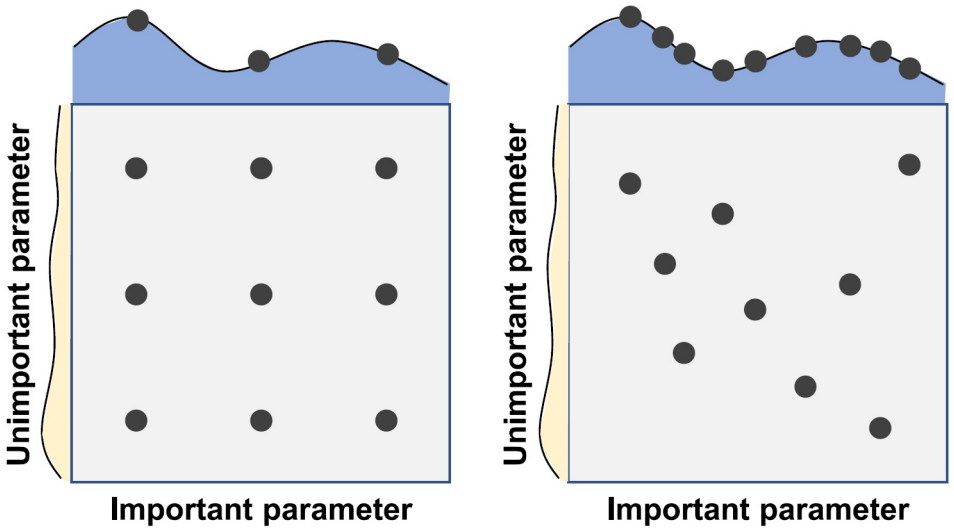


Figure 5.2 Optimizing a two-dimensional space for nine trails using grid search and random search [83].

Studies reveal that random search is more efficient and applicable than grid search, depicted in figure 5.2; however, finding optimal value using random search is not guaranteed. In other words, a longer search time in grid search increases the possibility of discovering ideal hyperparameters [83, 85].

5.4.3 Bayesian Optimization

One of the efficient approaches for the costly black-box optimization problem is the Bayesian optimization technique. Detailed information regarding Bayesian optimization can be found in [85,86]. The strategy in Bayesian optimization is applying a validation set and constructing a probabilistic model translating hyperparameters to objective metrics. Hence for globally optimizing expensive black-box functions, Bayesian optimization is an efficient optimization approach.

According to the [85] notations. The sample pairs (θ_i, y_i) are stored in dataset D, where $\theta_i \in \Theta$ and y_i indicate a sampled neural architecture and evaluation result, respectively. The algorithm steps are as follows.

2.1 Adjusting the probabilistic model M to be suitable for the dataset D.

2.2 Choosing the next acceptable neural architecture from the probabilistic model M by using the S function.

2.3 Evaluating the performance of the nominated neural architecture by using f function (i.e., this is a costly operation)

2.4 Updating the dataset D by inserting the new pair of results (θ_i, y_i)

The whole procedure iterates T times, which is determined according to the overall time and consumed resource. Typical surrogate models used for the Bayesian optimization approach are Gaussian process [87], Random Forest [26], and tree-structured Parzen estimator [88]. Although Gradient-based optimization is one of the widespread surrogate models, scaling cubically according to the size of the data set is a significant issue in applying the Gaussian process [26]. On the other hand, Random Forest provides better results on larger data sample and handle larger space.

5.4.4 Gradient-Based Optimization

Gradient-based optimization (GO) methods are another alternative for hyperparameters optimization [85, 89, 90]. To optimize the hyperparameters, GO algorithms take advantage of the gradient information enabling it to enhance the performance of HPO considerably.

5.4.5 Sequential Model Algorithm Configuration (SMAC)

As discussed in the previous section, it is beneficial to apply ML models to solve algorithm configurations for diverse problems. Perhaps one of the most useful usages is optimizing the parameters of the training set problem. Later these parameters can be utilized to enhance the application's performance in real-world usage. For example, IBM ILOG CPLEX includes 76 different parameters for its operation that must be configured accurately to enhance performance [26].

Similarly, in our docking maneuver application, there are many parameters whose tuning has a significant impact on its performance. For example, in one of the experiments for decreasing the transmission slot time to improve the throughput and decreasing docking time, we noticed that although we could half the transmission time, docking time did not decrease as expected. Inspecting the issue reveals that the slight improvement in tunning controller parameters significantly reduces the docking time. Hence, we concluded that the several parameters for our docking maneuver use case should be optimized automatically. Therefore, not only it ensures that the proposed hybrid method works correctly, but also, the comparison among different protocols can be made consistently and fairly.

There exist two approaches to characterize parameter configuration and optimization. The first black box-based method is model-free, which is not too complicated to implement. The research was performed recently and aimed to enhance its performance significantly. [26, 91, 92]. The most applied model-free configuration methods are F-RACE [92] and PARAMILS algorithm [93].

Algorithm Framework 1: Sequential Model-Based Optimization (SMBO)

$\mathbf{R}$ keeps track of all target algorithm runs performed so far and their performances (*i.e.*, SMBO's training data $\{([\boldsymbol{\theta}_1, \boldsymbol{x}_1], o_1), \ldots, ([\boldsymbol{\theta}_n, \boldsymbol{x}_n], o_n)\}$), $\mathcal{M}$ is SMBO's model, $\vec{\boldsymbol{\Theta}}_{new}$ is a list of promising configurations, and t_{fit} and t_{select} are the runtimes required to fit the model and select configurations, respectively.

Input : Target algorithm A with parameter configuration space $\boldsymbol{\Theta}$; instance set Π; cost metric $\hat{c}$

Output : Optimized (incumbent) parameter configuration, $\boldsymbol{\theta}_{inc}$

$[\mathbf{R}, \boldsymbol{\theta}_{inc}] \leftarrow$ *Initialize*($\boldsymbol{\Theta}$, Π);

repeat

  $[\mathcal{M}, t_{fit}] \leftarrow$ *FitModel*($\mathbf{R}$);

  $[\vec{\boldsymbol{\Theta}}_{new}, t_{select}] \leftarrow$ *SelectConfigurations*($\mathcal{M}$, $\boldsymbol{\theta}_{inc}$, $\boldsymbol{\Theta}$);

  $[\mathbf{R}, \boldsymbol{\theta}_{inc}] \leftarrow$ *Intensify*($\vec{\boldsymbol{\Theta}}_{new}$, $\boldsymbol{\theta}_{inc}$, $\mathcal{M}$, $\mathbf{R}$, $t_{fit} + t_{select}$, Π, $\hat{c}$);

until *total time budget for configuration exhausted*;

return $\boldsymbol{\theta}_{inc}$;

Figure 5.3 Sequential Model-based Algorithm [26].

On the other hand, to improve the performance even higher, recent studies also focus on the model-based algorithm configuration. For example, sequential model-based optimization (SMBO) iteratively tries to configure the parameters to fit the model and examine fitting on the model. Then according to fitting on the model, the parameters are selected for the next iterations [26]. Acting like a black box for optimization, SMBO suffers from restrictions that make applying automated ML-based optimizations unsuitable. The main restrictions of using SMBO algorithms are as follows:

- Supporting only numerical parameters
- Not supporting parallel instance for optimizing the selected algorithm
- Not terminating the algorithm with poor results.

The Sequential Model-based Algorithm Configuration (SMAC)for the configuration of the parameters is capable of overcoming the first two limitations but still suffers from the third restriction that will not terminate on the poor performing algorithm. Sequential model-based optimization algorithms predict the performance by building a regression model and optimizing the algorithm by using the model.

Considering the training set $\{(\theta_1, o_1), ..., (\theta_n, o_n)\}$ of the model and $\theta_i = (\theta_{i,1}, ..., \theta_{i,d})$ is the all sample parameters of the selected algorithm d. Here o_i represents the observed performance value by running configuration θ_i. The goal is by providing a new configuration θ_{i+1}, the model estimates its performance o_{i+1}.

The pseudo-code for the time-limited SMBO method is provided in Algorithm 5.3. In this algorithm, after initialization, a fitting model is performed. Using the result list, the algorithm decides which parameter configurations to execute and iterates

this process until the allowed execution time is reached [26].

In overall, it is imperative to avoid manually tuning the PID gains for different communication schemes. Hence the SMAC algorithm is used to automatically adjust the hyperparameters. The parameters that need to be adjusted, as well as the objective function, are passed to the SMAC algorithm, which iteratively runs simulations to find optimal values for the desired parameters. In this manner, the SMAC algorithm can resolve the complex combinational problem of determining optimal gains for the controller [94].

Chapter 6

UNDERWATER NETWORKED CONTROL SYSTEM CO-SIMULATION ENVIRONMENT

Simulation is an essential research and development tool in the ever-more complex field of robotics and unmanned vehicles. For example, a novel control algorithm should be tested on the developed vehicle before implementation, especially for unmanned vehicles. Failure of the algorithm or malfunction of the vehicle can result in potentially costly and hazardous results. Employing a virtual simulation environment close to real-world circumstances not only eliminates all risks associated with real-world experiments but also provides the opportunity to verify algorithms' correctness. Moreover, in the case of swarms of vehicles, without verifying their correctness in a simulation, it may be impossible to ensure their controllability in the actual environment. In recent years, for different purposes, numerous simulators have been developed. A general purpose simulation software is typically the most common, as it is intended to simulate any environment or vehicle regardless of its type. In contrast, very few of these simulators are used or tested with underwater applications, and even fewer demonstrate UUVs as their main purpose [95].

Considering the harsh environment where AUVs operate to achieve a mission, utilizing the proposed integrated co-simulation considerably decreases the time, cost, and risk of a mission by reducing the number of trails to obtain optimal values [96]. For this purpose, we have developed an integrated co-simulation environment of the hybrid RF and acoustic underwater networked control system for implementing

the docking maneuver application. The physics is simulated in a realistic manner by Gazebo [97], whereas the control and communication protocols simulation is provided realistically in Matlab True-Time. By creating such a design, we have employed the best properties of the different simulators.

For a realistic simulation environment for NCS, we have employed Gazebo, which makes it possible to implement complex robots in modeled dynamic environments in realistic scenarios. Furthermore, within Gazebo, we have employed UUV Simulator [98], which is a simulator explicitly designed to model underwater dynamics [29] accurately. UUV Simulator can realistically simulate unmanned underwater vehicles, including AUVs and ROVs, considering the external disturbances, and hydrodynamic and hydrostatic effects of water.

Developing a realistic underwater simulator requires obtaining the necessary parameters based on actual underwater vehicles. For this purpose, the specification of the ROV SF 30k [99] is used in mathematical modeling, linearization and graphical representation RexROV [100] in the co-simulation environment.

Hence, the values used for rigid body and added mass matrices during system modeling and simulation are based on the water tank or sea trial experiments performed on ROV SF30K [99], Minerva [34].

6.1 Underwater Simulation Requirements

A proper simulation environment is required for testing the dynamic of the AUV as well as the communication protocol. In this section, we are going to provide detailed information on the simulation environment for the AUVs and how to develop and configure algorithms inside it. Several different simulation environments were provided for the evaluation of the underwater application. The functional requirements for the appropriate simulator [101] are determined as follows:

- Simulation environment should be capable of working with Robot Operating System (ROS) in order to take advantage of the ROS library for using sensors, actuators, etc.
- There should be the capabilities of publishing and subscribing to topics in order to collaborate with external programs (i.e., Matlab).
- Simulation environment should handle the simulation of several instances of AUVs since some scenarios require multiple AUVs collaboration for achieving a mission.

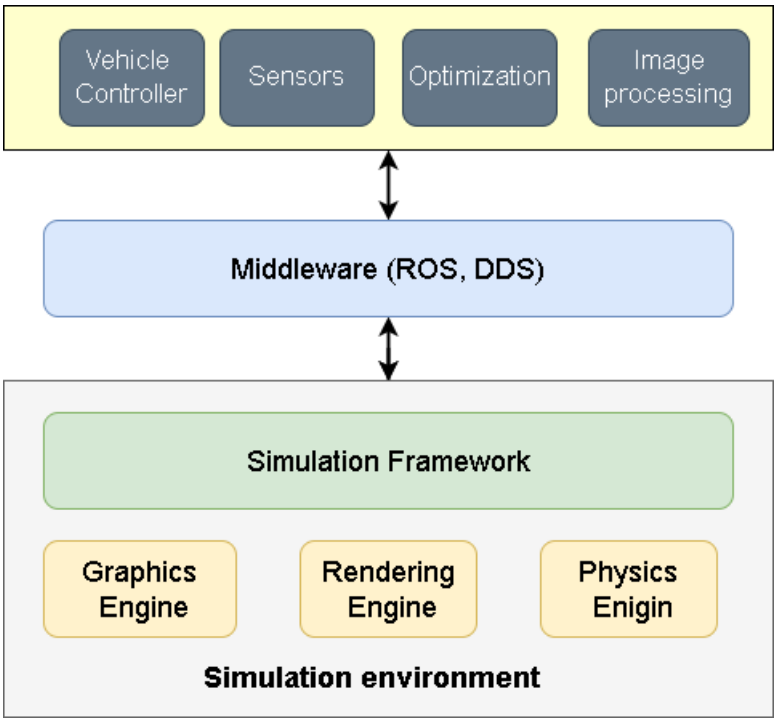


Figure 6.1 Typical software architecture of a simulator

.

- Simulator should provide a physics engine that supports the simulation of hydrodynamic and hydrostatic forces, rigid body, and restoring forces.
- Inserting a new vehicle into the simulator should be supported without changing the source code.
- It should include a rendering engine for visualizing the operation.

6.1.1 Simulator Architecture

This section investigates available simulators that are used for simulating robots and UUVs. Hence, conducting a review of the simulators is essential to determine whether they are suitable for simulating AUVs, as well as swarms of UUVs in a mission. The following features serve as the basis for simulators that are used for robotics:

- Providing an adequate level of physical fidelity to model actuaries and manipulators
- The capability of connecting using middleware or programming interfaces.
- Supporting different acoustic and optical sensors.
- Active community and/or support, sufficient documentation, and previous us-

age in the literature

UWSim, MORSE, and Gazebo are three of the simulation tools that are mostly for the design and simulation of mission planning and navigation. This is because they allow the integration of a variety of sensors, such as sonar, navigation, and multibeam sonar into a simulator. Simulators used in robotics should consist of the following components.

- Rendering engine
- Physics engine
- Simulation framework

Generally, rendering engines provide real-time high-level rendering features for 2D or 3D mesh drawing, different points of view, camera, and scene graphs such as Ogre3D [102] and Irrlicht rendering engines. In contrast to the rendering engine, the physics engine focuses on the interaction between different objects in the scene. By coupling the physical engine, such as Bullet [103] and Open Dynamics Engine (ODE) [104] with rendering engines, objects can move continuously. Figure 6.1 depicts the main components involved in the architecture of a typical simulator [95].

In robotics, middleware is commonly referred to as a high-level software layer that connects the robot's low-level control subsystems. Robot Operating System (ROS) is an instance of commonly used middleware that is used for node creation and provides communication among nodes via a message-passing mechanism. In the ROS network, the simulator can be separated as an individual node from the control node. This decoupling provides several benefits such as reliability and scalability of the system. Hence rather than modifying the simulator, control algorithms can be developed and integrated into the simulation software. This enables different nodes to utilize various controllers and switching and swapping can be done easily. Moreover, robots with middleware on board may also integrate with a simulator, allowing hardware-in-the-loop (HiL) simulation to be carried out [95].

6.1.2 Real-Time System Simulation using TrueTime

To add computation and network layer functionalities into our co-simulation, we have used the TrueTime library. This library was mainly developed for the simulation of distributed real-time control systems close to the physical layer. In fact, traditional control design using MATLAB/Simulink usually omits the temporal effects arising from the actual implementation of controllers. Controllers are often implemented as tasks in a real-time kernel and communicate with other nodes over

a network. Consequently, the constraints of the target system, e.g., limited CPU speed and network bandwidth, must be taken into account at design time. Hence TrueTime makes it possible to simulate the timely behavior of real-time kernels executing controller tasks [105].

TrueTime makes it possible to simulate networked control loops. It is possible to schedule tasks using different scheduling policies (e.g., priority-based preemptive scheduling and earliest-deadline-first (EDF) scheduling [105, 106]. In this thesis, we have used EDF scheduling to orchestrate the execution of tasks by prioritizing those with the earliest deadline.

In this thesis, various networking functionalities are implemented using TrueTime network blocks. The event-driven network block executes when messages enter or leave the network. A message contains information about the sending and receiving computer node IDs, arbitrary payload (typically measurement signals or control signals), the length of the message, and optional real-time attributes such as a priority or a deadline. In the network block, it is possible to specify the transmission rate, the medium access control protocol (CSMA/CD, CSMA/CA, round robin, FDMA, or TDMA), and other network-related parameters. A long message can be split into frames transmitted in sequence, each with additional overhead. When the simulated transmission of a message has been completed, it is put in a buffer at the receiving computer node, which is notified by a simulated hardware interrupt.

Accordingly, we designed our underwater network by explicitly modeling the acoustic and RF channel characteristics and using the network blocks of TrueTime, path loss, fading, delaying of signals, etc., to implement both acoustic and RF channels. Furthermore, different acoustic communication protocols such as TDMA, WR, and S-ALOHA have been implemented by selecting and adapting robust data-link, MAC, and network protocols. In addition to RF communication, we opted to employ the CSMA/CA protocol. By providing an efficient switching mechanism, shifting from an acoustic link to RF transmission has been achieved during the mission. By using hybrid acoustic and RF communication mechanisms, we can provide reliable communication network infrastructure among cooperating vehicles and other elements involved in a mission. Moreover, to improve the data rate and decrease the delay, we have proposed to use in-band full duplex communication, which has the potential to double the data rate of both acoustic and RF links. To assess the proposed system, it is necessary to compare hybrid and full duplex communication with conventional acoustic based links.

While most AUVs are usually tailored to accomplish a specific task using a proposed communication and control platform, heterogeneous UUVs from different vendors

can work together to accomplish a mission. Considering the OSI network reference model, our proposed communication network implements and utilizes the OSI model's lower layers, namely, the physical layer, MAC layer, and network layer. Separating the tasks based on the OSI reference model leads to a robust communication network protocol stack for underwater networks.

6.2 Underwater Simulation Tools

Table 6.1 Comparison of different simulators [95]

Simulator and main features List					
Simulator	Physical engine	programming API	simulating sensors	Visual Fidelity	Extensibility
UWSim	High	Single	Yes	High	Low
MORSE	High	Multiple	Yes	Medium	High
Gazebo	High	Single	Yes	Medium	High

In the following, some of the most popular simulators are reviewed in terms of their capability to include external tools, provide rendering and physics engines, and provide enhanced features by adding plugins to make them suitable for underwater applications. Table 6.1 summarized a comparison between each simulator discussed in this section [95].

6.2.1 UWSim

The rendering engine and physics engine of UWSim are based on OpenSceneGraph [107]. As part of its physics fidelity, UWsim utilizes Bullet for collision detection and rigid body dynamics. In addition to physically interacting with rigid bodies, robotic arms attached to vehicles may also interact with objects. Through ROS, UWSim provides access to external software such as MATLAB by employing topics to control a vehicle or receive sensor data from ROS nodes. For instance, an image message broadcast by ROS may be captured by a node and processed for vehicle control. Furthermore, using multiple virtual cameras enables to simulate a stream of video, Sonar, force, pressure, and IMU sensors as well as GPS, can all be simulated in UWSim. Almost all of these sensors transfer data via ROS message types except for the pressure sensor, which uses a custom message type. A range camera is employed to mimic sonar behavior by taking a picture of the depth buffer of the renderer and transferring it through ROS topics. UWSim can effectively simulate underwater vehicles by enabling rendering for the environment and adding robots without effectors and actuators. The unique feature of UWSim is its use of OsgOcean, a plugin for Open scene Graph that enables highly realistic simulations of

underwater environments. Although UWsim uses XML scripts for launching several UUVs, increasing the number of vehicles will be cumbersome to manage in UWsim. Additionally, extending software is not easy to perform in UWsim.

6.2.2 Gazebo

Gazebo is one of the widely used open source simulation environments developed by the Open Source Robotics Foundation (OSRF). With the help of four physics engines (ODE, Bullet, Simbody, and DART) and OGRE for visualization rendering, Gazebo is used as a general purpose robotic simulation. Customized capabilities such as sensors, environments, etc can be integrated into the Gazebo by using custom plugins using $C++$ API. This provides a high level of modularity for system development. Socket based method via Google Protobufs is used as an interface for the plug-ins, which separates server operation from GUI. Gazebo also supports multi-robot simulation using SDF files to configure all objects in the simulation. Furthermore, there is a binding between ROS and Gazebo which makes the interaction among them fairly straightforward. Moreover, Gazebo provides a buoyancy plugin for simulating the behavior of underwater objects. The Buoyancy Plugin computes the buoyancy force for each link of the object and applies the force to the center of the volume of the link. Fluid density, a center of volume, and volume are the required parameters that should be specified in the SDF file [108].

6.2.3 UUV Simulator

Although Gazebo provides the buoyancy plugin for underwater object simulation, it suffers from the lack of essential features such as damping forces, underwater currents, added mass, Coriolis and centripetal force. For augmenting these features to our simulation environment we are using UUV Simulator which is a Gazebo based plugin that provides required features for the simulation of underwater vehicles in the Gazebo environment [100]. UUV Simulator also provides collision detection and rendering features for a realistic simulation environment. SDF and URDF files which are similar to XML files are used in Gazebo. While SDF is used to describe the objects, vehicles, and worlds in Gazebo, URDF is used to describe the robot's features according to the ROS framework.

Whole robot related data such as kinematic and dynamic, joint friction, sensors, texture, etc., are stored in SDF files. Furthermore, other vehicle related data such as geometry files for rendering and collision detection should also provide to the simulator. It is worth noting that information for rendering and collision detection are stored in COLLADA and STL files respectively. Since collision detection can

take a considerable amount of time for the object with large geometries the collision detection and rendering files are separated. All the objects need to be inserted into the Underwater world plugins. The water currents are also presented in the simulator either by constant or random speed using a Gauss-Markov process.

The equations of motion which were derived before were also implemented via the underwater Object model plugin and for each vehicle, it will be initiated to provide underwater effects.

During the initialization of URDF files, physical parameters are set for the Underwater Object plug-in by receiving the messages published by the underwater current, then the velocity according to the surrounding waters is calculated.

The essential components of the AUV are the vehicle base links, actuators, sensors, and manipulators. To apply hydrodynamic and hydrostatic forces and moments to the body of the vehicle, underwater plugins use the implementation of Fossen's equations of motion for calculating the force and moment. AUVs are assumed as a single link with thrusters and sensors connected to them via fixed joints. With this assumption hydrodynamic and hydrostatic forces can be easily applied to the main body link. Gazebo uses conventional ENU (East-North-Up) for the object coordination system. However, in marine science (as well as the equation of motion) NED (North-East-Down) coordinate frame is used for coordinating an object. Hence it is essential to keep track of coordinate frames and convert from one frame to another when it is required. To simplify this operation UUV Simulator provides a NED coordinate along with a base link with a conventional ENU coordinate frame [101].

Separation of thrusters from the main body enables the calculation of thrust allocation matrices for the vehicle in the simulation automatically without adding them to the vehicle configuration file. Hence without changing the source code it is possible to use different thruster configurations for the vehicle.

For the vehicle that we are using in this thesis actuator and sensor have no physical significance to the body of the vehicle. In other words, there is no need for sensors and actuators to be positioned in the collision geometry of the AUV's main body.

The REX ROV Vehicle that we are using has the following files structure:

- Launch: includes the launch file for integrating the REX ROV vehicle to our custom underwater world in UUV Simulator
- Mesh: includes rendering Collada and collision geometry STL files

- Robots: the URDF description of the REX ROV configuration files in the format of $< robotname >< configuration\_name > .urdf.xacro$
- URDF: includes robot description files with actuators and sensors plugin information in the format of $< robotname > base.xacro$.

This modular structure in implementing the simulation environment and integrating with Gazebo and ROS provides an efficient way for co-simulation with other computation and simulation tools such as Matlab.

To sum up, implementing communication protocols in Matlab Simulink and using ROS as a bridge for connecting to the Gazebo enables us to provide a real-time co-simulation environment for testing different controllers and future machine learning-based algorithms.

6.2.4 Modeling of Water Currents

Evaluating the way the vehicle behaves under various disturbances is a critical part of testing the performance of the designed controller. The option is to apply force and torque directly via the GUI interface in our Gazebo-based simulator or call a ROS service to accomplish this task. Moreover, it is possible to provide current velocity to the vehicles using the underwater world Gazebo plugin [100].

Ocean currents are formed by the interaction of gravity, wind friction, and variations in water density, resulting in horizontal and vertical water circulation. These currents are also affected by heat exchange and salinity changes at the sea surface, known as thermohaline waves. The oceans have two water spheres, the cold and warm spheres, and the Coriolis force affects the major currents differently in the northern and southern hemispheres. Planetary interactions, like gravity, can generate tidal waves that reach speeds of 2 to 3 m/s or more along coasts [29].

Considering Water Currents in Equations of Motion

The following model can be used to simulate water currents and their impact on underwater vehicles.

$$M_{RB}\dot{\nu} + C_{RB}(\nu)\nu + g(\eta) + g_0 + M_A\dot{\nu}_r + C_A(\nu_r)\nu_r + D(\nu_r)\nu_r = \tau_{wind} + \tau_{wave} + \tau$$

Here, $\nu_r = \nu - \nu_c$ represents the vector of relative velocity. In the case of an irrotational fluid, ν_c represents the generalized water current speed.

$$\nu_c = [u_c, v_c, w_c, 0, 0, 0]^T \tag{6.1}$$

Where u_c, v_c, w_c are represented in body frame.

To model the water current velocity of V_c, the angle of attack α_c and the sideslip angle β_c relative to the underwater vehicle should be considered. A Gauss-Markov process of the first order can be used to generate water current speed for computer simulations. Hence it is possible to account for slight deviations near an average value in velocity, horizontal, and vertical angle using Gauss-Markov processes.

$$\dot{V_c} + \mu V_c = w \tag{6.2}$$

Here a Gaussian white noise is represented by w, while $\mu \geq 0$ is a constant. In case μ is equal to zero, this model can be considered a random walk, corresponding to the integration of white noise over time. In the integration process, a saturating element is typically used to limit the speed of current integration.

$$v_{min} \leq V_c(t) \leq V_{max} \tag{6.3}$$

Fixing the direction of the current is achieved by providing constant values for α_c and β_c. Hence, the simulation's current direction can be changed over time using α_c and β_c [29].

6.3 Proposed Co-Simulation Environment

In this work, Robot Operating System (ROS) [109] provides a flexible framework for programming and modeling robots and controlling their operations. It primarily serves as a middleware to connect different parts of the NCS. However, ROS lacks realistic models of communication physics and protocols and models of real-time embedded computers. To augment these areas, the True-Time toolbox of Matlab Simulink has been employed in this work. Executing the source code of the actual model in the embedded computers of True-Time helps schedule the tasks for co-ordinating different parts of the co-simulation environment. Moreover, True-Time supports the simulation of wireless networks in the MAC layer and supports several built-in communication protocols with extension possibilities.

Providing a co-simulation environment by integrating the aforementioned simulators is one of the main contributions in this thesis. Figure 6.2 depicts a workflow of

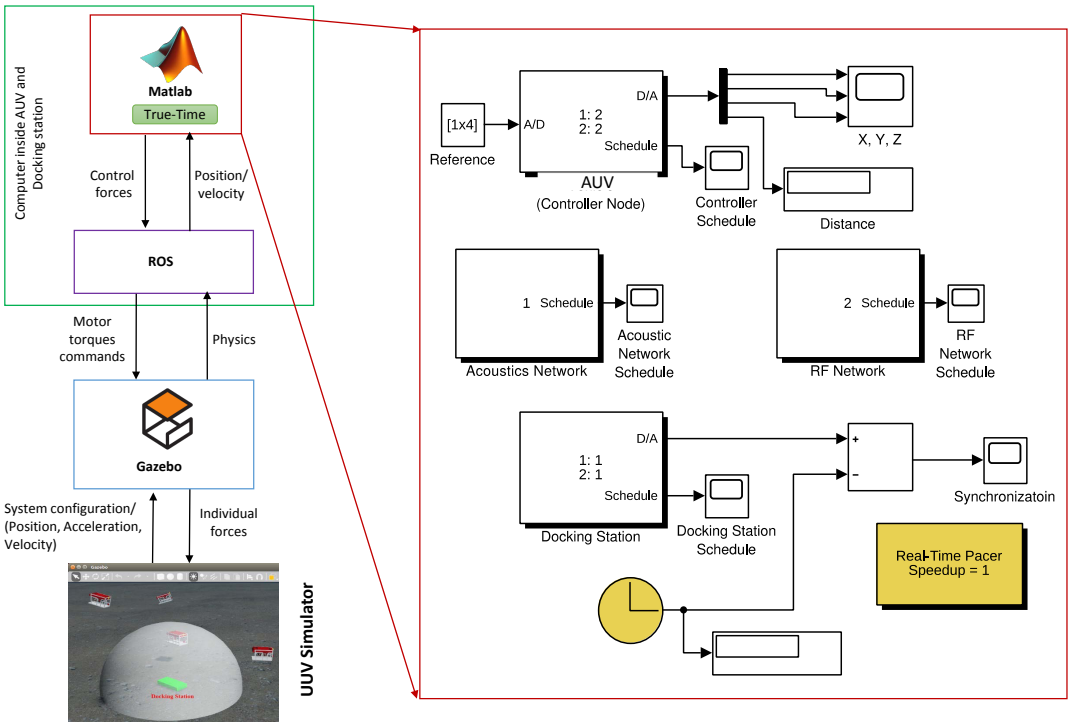


Figure 6.2 Workflow and Simulink models for proposed co-simulation framework

the messages and interaction of different components and a detailed Simulink model. Our goal is to incorporate different co-simulation modules to provide an environment for developing and testing different missions performed autonomously using the underwater networked control system. However, synchronization of Matlab True-Time and Gazebo is an issue in co-simulation as they both apply dynamically variable time steps in their solvers, causing their simulation time to progress in different time steps. To synchronize different simulators' clock time, we use the messaging capabilities of ROS. Once two simulators are initialized, True-Time employs Gazebo's clock to update its timer. Determining the AUVs' position, controlling their movement and attitude, as well as the communication network and links, are modeled using True-Time.

The Matlab True-Time model is depicted as the inset in Figure 6.2. The block labeled "AUV" implements the onboard real-time computer in the level of detail of source code execution. The actual source code that will run on the AUV is executed over this block. The "Acoustic Network" blocks and "RF Network" provide communication and packet delivery models over acoustic and RF links between the vehicles and the docking station. The communication links model both the MAC protocol and the physical layer so that physical effects, such as path loss considering the vehicle's instantaneous position with respect to the docking station, and physical characteristics of water, such as salinity, can be implemented in the simulator. The "Docking Station" block at the bottom of Figure 6.2 implements the docking station's

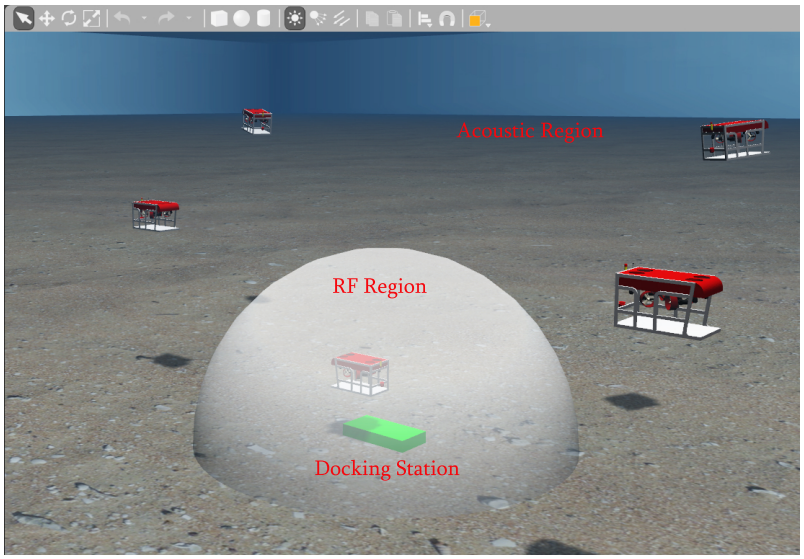


Figure 6.3 Acoustic and RF operation ranges of multiple AUVs with respect to the docking station. The point of view is underwater.

real-time embedded computer. This block is responsible for employing the position sensor to determine the AUVs' location and transmitting position data to the AUVs.

6.4 Hybrid Networked Control System Model

The docking station is positioned on the seafloor, to provide a safe place for keeping AUVs, charging their batteries, and transmitting collected data via a high data-rate wired link. Our docking scenario is based on the omnidirectional system in which AUV can enter to docking station from any direction. However, docking an AUV in the sample space of docking needs high precision to ensure that AUV does not damage the docking station. Figure 6.3 depicts a scenario in which an AUV performs a docking maneuver while the rest of them move around the docking station. To leverage the advantages of both RF and acoustic waves for better navigation of AUVs, we propose a hybrid communication system employing acoustic communication over long distances and RF communication over short ranges. Figure 6.4 depicts a block diagram of the system in which acoustic and RF links used on the networked control system are specified.

AUVs usually are not equipped with a positioning system due to the high complexity and weight of such equipment. However, we can take advantage of the NCS for solving this problem. For this purpose, a message regarding the position information is broadcasted periodically by the docking station to all AUVs, using a position detection device called Ultra Short Base Line (USBL) [110]. Upon receiving these messages and using an onboard implemented feedback control algorithm, the error

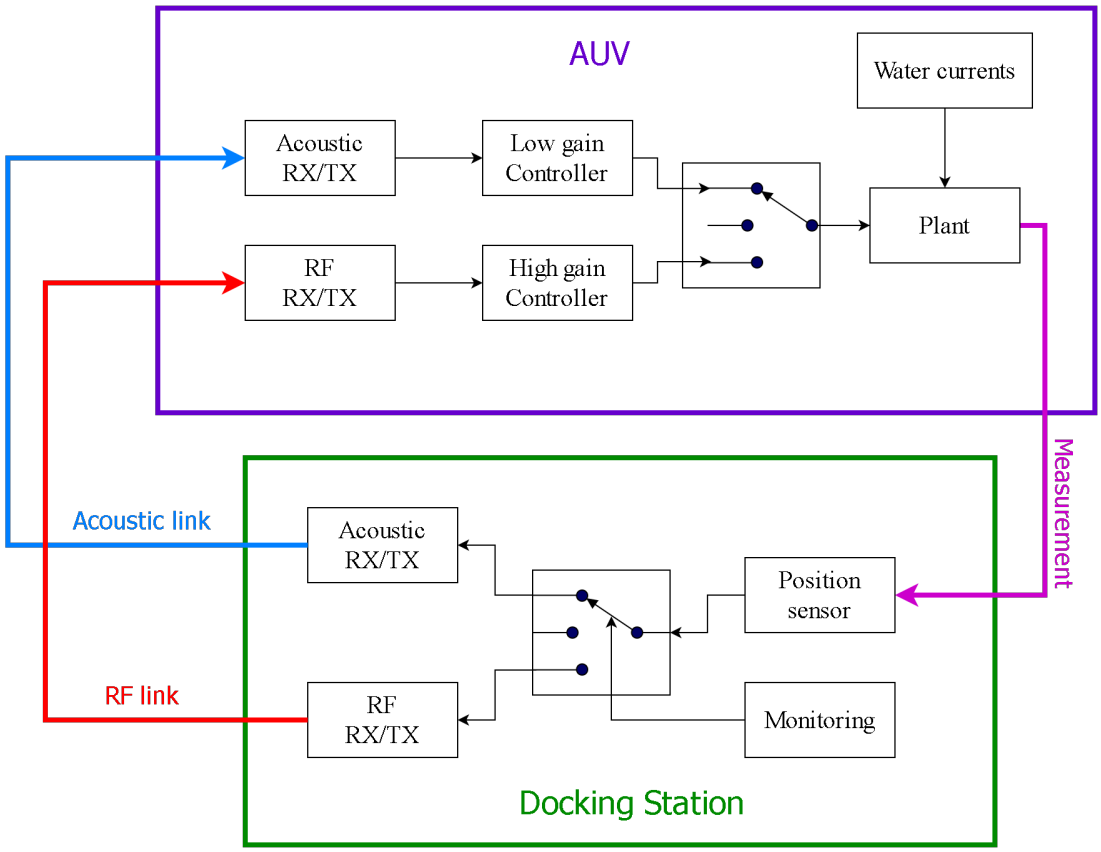


Figure 6.4 Block diagram of the proposed hybrid networked control system.

is measured and the required force and torque are generated to be applied to the AUVs propellers.

For successful docking, the communication between the AUVs and the station should be reliable. Regardless of operating mode, the docking station always initiates the communication by transmitting a packet with initial transmission power. If an AUV did not receive the signal by the expected time (e.g. due to fading), the required power level for correct reception of the signal is calculated by AUV and transmitted to the docking station. Therefore, by adjusting the transmission power, the docking station tries to improve the reliability of communication.

Essentially, in our proposed NCS, all complex node orchestration and communication will be handled by ROS as middleware. Figure 6.5 illustrates multiple ROS nodes and their dependency on each other. In order to generate this graph, ROS rqt_graph feature is used to visualize the computation graph of the nodes and their interactions. Additionally, it also represents a ROS distributed message passing through topics in both the subscriber and publisher mechanisms. The nodes in Figure 6.5 are represented by ovals, and messages published through topics are shown as rectangles

Although distance has the utmost importance for switching between acoustic

Figure 6.5 Communication of spawned ROS nodes through ROS topics.

and RF modes other criteria [6] to change from one mode to another in order to minimize interference and reduce the required time can be employed as follows:

- Priority given to higher-speed modes based on SNR threshold.
- Distance-based energy efficiency [J/bit/m] – to reduce network energy
- Transmit time – Considering non-uniform node density, throughput should be maximized and latency should be minimized.
- Node available energy efficiency [J/bit]
- Maximizing results of the coverage range × transmission time
- Throughput of each area or volume.
- Maximizing per-area throughput, taking into account the locations of the nodes in the surrounding area, the product of transmission time, and neighbors.

Ultimately, the advantages of the proposed integrated co-simulation environment in our research can be summarized as follows:

- We simulate computers and communication protocols down to the hardware simulation level with high precision using TrueTime.
- To simulate the physics, including hydrodynamics to fine details, we use Gazebo, ROS and UUV-Simulator which are de facto standards and tools for robotic developments.
- By considering the physical layer details of the network, we simulate the underwater communication channel (acoustic and RF). In this way, our integrated co-simulation environment can be used efficiently to design and test novel underwater network protocols and schemes.
- We integrate these methods by linking the various simulators appropriately and allowing the complete system to run in real time. As far as we are aware, this is the first known work to implement the mentioned systems to obtain a high accuracy underwater simulator using a networked control strategy.
- We implement the control algorithms on the simulated computers in discrete time. In order to do this, the dynamic of underwater vehicles is studied and modeled precisely. In this way, new underwater vehicles can be tested using our integrated co-simulation environment.

- The NCS we have developed in this thesis can be utilized to control different heterogeneous underwater vehicles remotely to perform desired tasks collaboratively to achieve a mission.
- In our co-simulation, we have integrated photorealistic 3D environments based on game engines such as Unity and Unreal. In this way, we will develop and test algorithms using the images from the AUV's sonar and camera easily and efficiently.

6.4.1 Stability Issues due to Gain Scheduling

The underwater environment has a high damping coefficient. Due to this property, an instability that may result from energy accumulation in the system due to gain scheduling is dissipated into the water. Due to this property of the AUV, oscillations or instability due to gain scheduling do not appear in the proposed method. This fact can also be observed in the results in Chapter 7 where no issues with the system's stability have been observed in any of the simulations.

Chapter 7

PERFORMANCE RESULTS

In this section, we present the performance analysis of the proposed underwater hybrid networked control system by presenting simulation results for performing docking maneuver in the calm water scenario (ideal case) and in the presence of water currents.

To evaluate the performance of the proposed system, the metrics that we used are a time to dock, motive power, cumulative error, and communication energy, which are considered as the common metrics in these types of applications [7]. Here time to dock represents a measure of how long it takes for an AUV to reach 2% of the target distance. Motive power is defined as the required force that is used for moving the AUV. The communication energy metric is the total energy consumed by an AUV to send and receive all (data and control) packets and cumulative error is the area between the docking station position and the AUV position (i.e., the integral of positioning error).

7.1 Simulation Settings and Parameters

To model the communication channel, our simulation setup is based on the following parameters. We have made use of the parameters of EvoLogics S2C R48/78 and WFS seatooth® S300 for underwater acoustic and RF modems, respectively. For motive energy calculations we have assumed the parameters of NeuMotors 1925-3Y. Maximum motive power is taken as 546 W. The MAC protocols are designed for eight AUVs (i.e. $V_{MAX} = 8$). Deploying AUVs in an underwater mission is costly, and coordinating multiple AUVs is challenging. We limit the number of AUVs in our use case to eight vehicles to address this. However, if the protocol needs to support more vehicles, it can be easily extended to accommodate them. Table 7.1

Table 7.1 Different MAC scheme parameters.

Parameters	Value
Docking station message length M_{DS}	512 bits
Acoustic modem data rate R_{AC}	10 Kbps
Docking station total frame time TF_{DS}	$0.4096s$
AUV message length M_V	64 bits
Distance to the docking station d	$50m$
Acoustic wave speed v_{AC}	1500 m/s
AUVs total frame time TF_{AUV} using TDMA	$0.034s$
AUVs total frame time TF_{AUV} using S-ALOHA	$0.030s$
Transmit Interval in WR TI	$0.01s$
Terminal Gap in WR TG	$0.030s$
Max number of slots for S-ALOHA (P)	$N = 8$

represents the parameters for MAC scheme design.

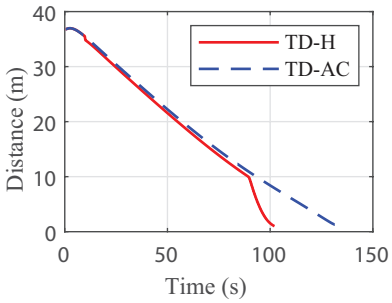
The docking station is positioned at [0, 0, -100]. In total, there are 5 vehicles in the simulation, 4 of them are located at the position of 50 meters away from the docking station and they are kept hovering in that position. The arbitrary initial position of the AUV which will perform the docking maneuver is [-20, 20, -75]. At the start of the simulation, only the acoustic link is active. AUVs receive packets from the docking station over an acoustic link. If they receive a message within an expected time interval, they accept the message and use the contents. Otherwise, they calculate the required signal power for proper reception and transmit it to the docking station. When the AUV reaches 15 meter distance from the docking station, it sends a "turn on RF" message in the upstream phase. The RF link is activated exclusively and controls the AUV through a high gain controller when it is closer than 10 m to the docking station. The high gain is possible because of the reduced delay in the control loop which affords better stability. The simulation parameters which we used are presented in Table 7.2. It also includes the parameters that have been used for calculating the path loss characteristics. After path loss, Rayleigh fading is applied to obtain the instantaneous received power level.

7.2 HD Hybrid NCS with PID Controller

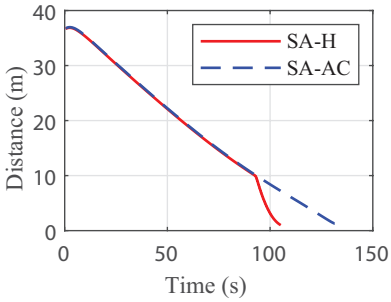
The goal for an AUV is to reach the docking station as quickly as possible and to perform precise docking. For this reason, time to dock under various water conditions and communication protocols is a good performance measure. The distance vs. time graph of a typical docking maneuver of an AUV for different protocols is shown in Figure 7.1. We aim to quickly perform landing on the docking station

Table 7.2 Simulation parameters.

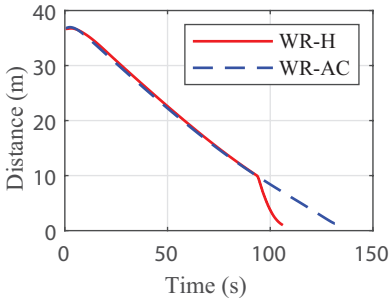
Parameters	Acoustic	RF
Carrier frequency	100 KHz	10 MHz
Data rate	10 Kbps	3 Mbps
Frame time	0.74 s	0.04 s
Signal power limit	4.5 W	3 W
Circuit power P_c	1.1 W	4.5 W
DS Packet time t_{ds}	3.84×10^{-2} s	0.28×10^{-4} s
Sampling period t_s	0.74 s	0.04 s
Proportional gain $K_p[x, y, z]$	[75, 75, 225]	[155,155,455]
Integral gain $K_i[x, y, z]$	[10, 10, 10]	[195,195,195]
Derivative gain $K_d[x, y, z]$	[65,65,55]	[270,270,55]
Derivative time T_p	0.74	0.04
Spreading Factor k	1.5	N/A
Threshold	$1.9mW$	$2mW$
Permittivity ϵ	N/A	$80(8.854 \times 10^{-12})F/m$
Permeability μ	N/A	$4\pi \times 10^{-7}$ H/m
Conductivity σ	N/A	0.01 S/m



(a) TDMA

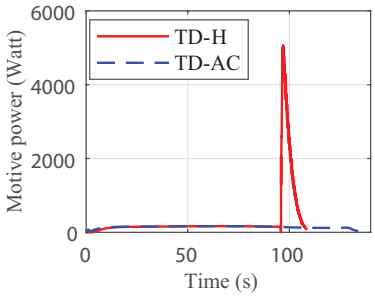


(b) Slotted-ALOHA

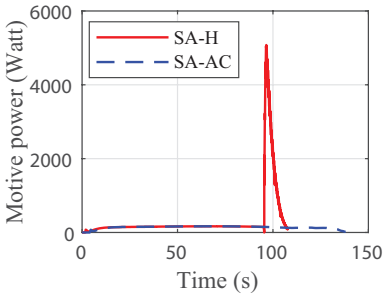


(c) Waiting Room

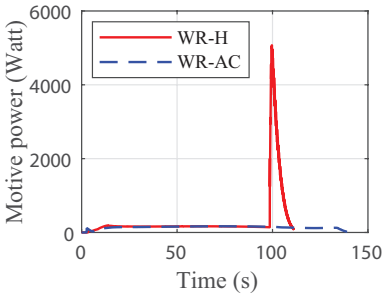
Figure 7.1 Time to dock of the AUV for acoustic-only (AC) and proposed hybrid (H) systems using TDMA, S-ALOHA, and WR, for the calm water scenario.



(a) TDMA



(b) Slotted-ALOHA



(c) Waiting Room

Figure 7.2 Motive power of AUV for acoustic-only and proposed hybrid systems using TDMA, S-ALOHA and WR for the calm water scenario.

without overshoot and steady-state error despite the physics of the water.

Figure 7.1 shows the performance of the three protocols in calm water; the proposed hybrid system uses solid lines, and the acoustic-only system uses dotted lines. The time to dock metric is introduced to measure how long it takes for an AUV to reach 2% of the target distance. Figure 7.1.(a) demonstrates that the AUV using the proposed hybrid system with the TDMA protocol can reach the docking station in about 102 s which is shorter than the acoustic-only system which takes about 140s. Similarly, in Figure 7.1.(b) and Figure 7.1.(c), we can observe that time to dock for the hybrid system is considerably less than the acoustic-only system. This is due to the fact that the AUV switches to the high gain controller offered by the RF link and utilizing increased sampling frequency enables the AUV to apply a more aggressive control effort; therefore, the final approach is quicker and more precise than the acoustic-only controller.

Time to dock is an essential parameter in the performance; however, it may come at the cost of high energy requirements. Moreover, the proposed method in this thesis is a viable hybrid approach where high power is only required toward the end of the docking maneuver. These results show the performance of our method from that perspective. Figure 7.2, depicts the motive power of thrusters over the docking time. To calculate the motive power, the thruster output is multiplied by the AUV

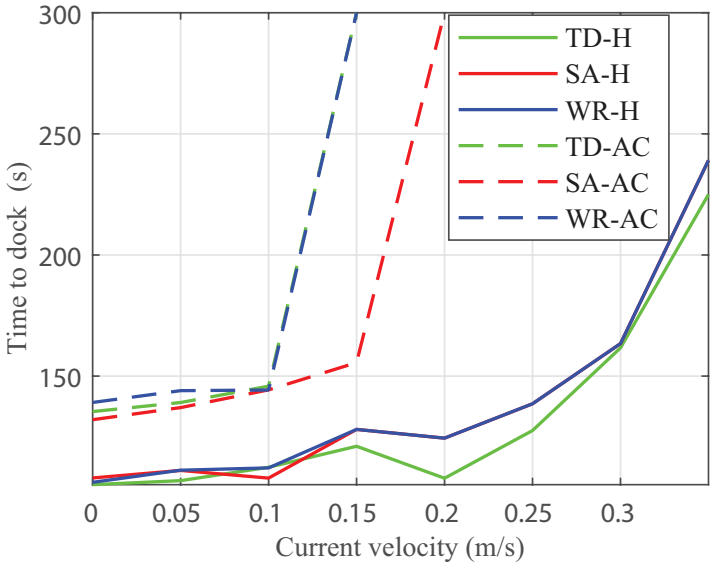
velocity. The output of the thruster is calculated by multiplying the control signal u by a constant value, which relates the thrust force to the control signal. Hence using $F_m = u/1.17$, gives us the output of the thruster and motive power is calculated by $P_m = F_m \times v$ where v is the velocity of AUV. This provides instantaneous power and by integrating motive power over time we can determine the motive energy. Low control effort in Figure 7.2.(a), 7.2.(b) and 7.2.(c) indicates that initially, only the low gain controller is used during acoustic communication. When an AUV approaches the RF operating region, the proposed controller switches to the high gain controller, which affords faster response control signals for navigating the AUV toward the docking station, but at the expense of much higher motive power. This can be verified by a short term rise in all three communication protocols in Figure 7.2.

High water currents cause AUVs to experience difficulty in approaching the docking station and making precise maneuvers in its vicinity. Since water currents cannot be avoided during real-world operations it is crucial to verify their effect on the operation of the vehicle. To consider the realistic underwater conditions of oceans and seas, we have applied disturbances to the modeled AUV in the form of water currents. Figure 7.3.(a) shows that the AUV using TDMA based acoustic-only system has time to dock of about 138 s when there is zero disturbance, but as the disturbance is increased to 0.15 m/s, the AUV cannot reach the docking station during the allowed simulation time of 300 s. The value 300 s represents that although AUV can approach the docking station's vicinity, it cannot perform docking successfully due to water currents. The AUV using the hybrid system with TDMA protocol, on the other hand, has time to dock at about 105 s, which is 32% faster than acoustic-only under zero disturbance. Although the performance worsens with increasing disturbance, compared with the acoustic-only system, the proposed hybrid system, due to faster control, can tolerate increasing current velocities of up to 0.35 m/s. For current values higher than 0.35 m/s, since the AUV cannot reach the RF region, the hybrid system could not accomplish the docking maneuver as well.

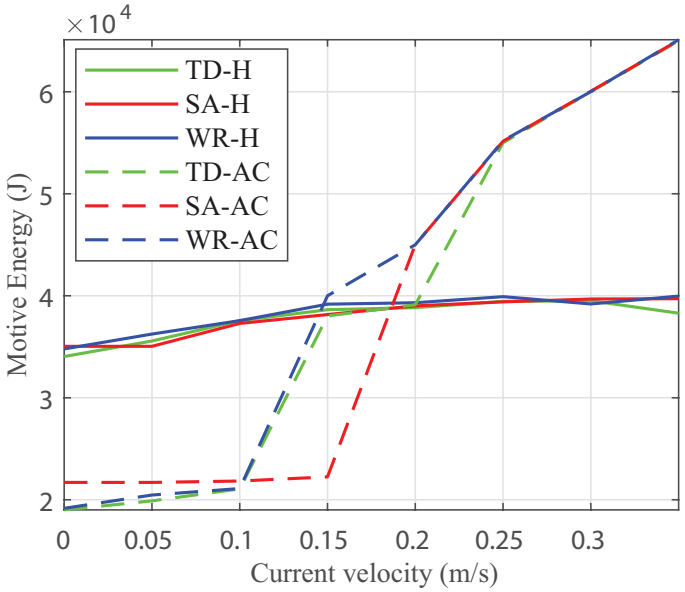
By applying currents to the system, Figure 7.3.(b) shows the motive energy used by the AUV during docking maneuver for the acoustic-only and hybrid system using TDMA, S-ALOHA and WR protocols. Using both acoustic-only and hybrid systems, the motive energy of AUV increases slightly with the increasing velocity of water currents as the controller has to apply a larger force to counter opposing disturbance force. The motive energy of AUV using the hybrid system is 61.48% larger than that of the acoustic-only systems for S-ALOHA, as the controller in the RF region applies a larger force to increase the velocity of the AUV. The higher motive energy of AUV using a hybrid system enables the AUV to have a significantly lower time to

dock and smaller cumulative error as compared to the AUV using the acoustic-only system.

Cumulative position error is defined as the sum of the absolute value of the distance between the reference position and the actual position of the AUV, summed up over every sampling time of the simulation run. Figure 7.4.(a) demonstrates the cumulative error averaged over 10 simulation runs, for hybrid and acoustic-only systems under increased water currents. Initially, an almost linear increase can be

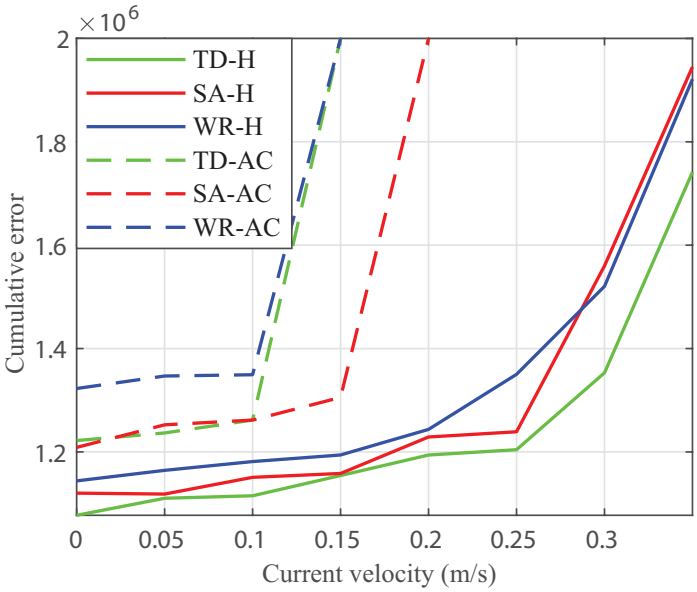


(a) Time to dock

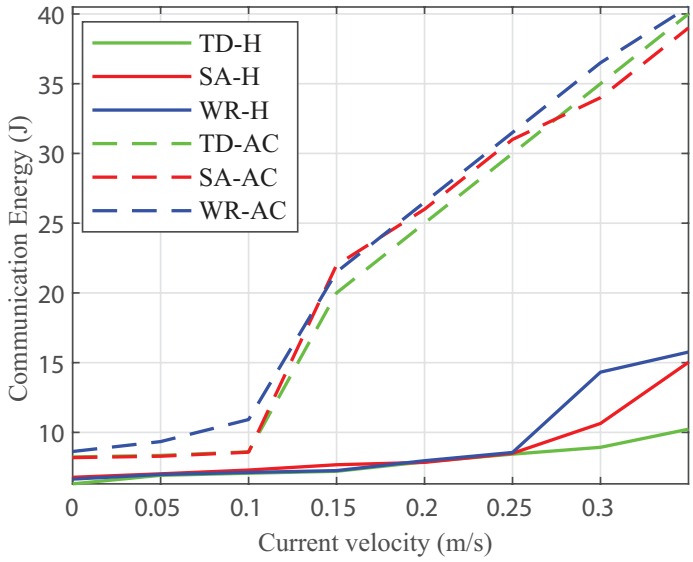


(b) Motive energy

Figure 7.3 Time to dock and Motive energy of AUV w.r.t increasing velocity of the water currents for acoustic-only and hybrid system using TDMA, S-ALOHA and WR protocols.(Simulation was terminated after 300 s.)



(a) Cumulative error



(b) Communication energy

Figure 7.4 Cumulative error and communication energy of AUV w.r.t increasing velocity of the water currents for acoustic-only and hybrid system using TDMA, S-ALOHA and WR protocols.

observed, followed by a sharp increase in the acoustic-only system using TDMA and WR at the velocity of 0.1 m/s, and for the S-ALOHA protocol at velocity 0.2 m/s. This can be explained by the fact that although the AUV can approach the docking station, it cannot complete the docking maneuver with low control gains. As a result, the cumulative error keeps increasing until the allocated simulation time expires at 300 s. In comparison to acoustic-only systems, the hybrid system with three different MAC protocols has smaller cumulative errors. The cumulative error of the S-ALOHA hybrid system is 16.67% less than the S-ALOHA using the acoustic-only system which gives the least cumulative error for an acoustic-only system. Furthermore, the WR and S-ALOHA have about 8.2% and 7.91% less cumulative error respectively, on average. Hence, it can be concluded that TDMA gives the most stable performance under increasing the velocity of water currents.

Although motive power is important, total communication energy also deserves extra attention because communication overhead is introduced to the system.

Figure 7.4.(b) represents the communication energy of AUV with respect to water current velocity. The energy consumption of the AUVs using an acoustic-only system increases marginally with increasing disturbance. However, when the velocity reaches about 0.15 m/s there is a sudden rise in the communication energy of AUV due to its attempt to approach to docking station but failing to make headway due to currents and communication energy consumption keeps increasing. We can observe that S-ALOHA and TDMA in acoustic-only spend about 26% less energy for communication than the WR. There is also a steady rise in communication energy for the hybrid system by increasing the velocity. When velocity reaches 0.25 m/s we can observe a sharp increase in the communication energy, especially in S-ALOHA and the WR which is 33.16% and 39.12% on average respectively. We can conclude that the hybrid system using TDMA protocol has the least communication energy among all.

7.3 FD Hybrid NCS with SMAC Optimized PID

Among the essential benefits of the proposed co-simulation environment is the ability to assess the entire closed loop in a networked control system under various circumstances, which would be impossible or extremely expensive in an actual sea trial. As an example application, the co-simulator could be used to evaluate a system in several situations for reinforcement learning and optimization algorithms. In order to enable this promising application in our co-simulation tool that includes Gazebo and ROS, the execution must be orchestrated so that all ROS nodes are started and terminated promptly. Hence, prior to running the simulation, the Gazebo should

be paused until all ROS nodes have been launched and then unpaused after the simulation has been completed.

In the UUV-Simulator, there is no default setting for simulation termination time. Nevertheless, to realize this, we set an initial timeout in the launch file, and the simulation will be terminated by reaching that timeout. The value for the time out is determined by the maximum amount of time required to complete the docking mission.

In order to conduct parameter evaluation, all events, including messages, generated topics, etc, during simulation runs need to be recorded for later processing. An efficient method for doing this is utilizing ROS bag files to record all the data from the simulation run. The bags are ROS's primary mechanism for logging data, so they can be utilized offline in a variety of ways. It is common to create bags using a tool such as ROSbag, which subscribes to ROS topics and records them in a file upon receiving the serialized message. Additionally, these bag files can reproduce the message and even remap them later to other topics in ROS [109]. Also, it is possible to turn off the Gazebo client, Rviz and other GUI components during iterative simulation in order to avoid additional overhead.

To iteratively run the simulation, it would have been much simpler if all code were in the same language such as Python or C++. However, the running scripts for the automatic iterative simulation must include Matlab code and Simulink blocks since the controller, logic, and networking are implemented in True Time toolbox of Matlab. Hence, it is necessary to further utilize the following MathWorks toolboxes in order to achieve this goal.

- Matlab coder
- Simulink coder
- Embedded coder

The goal was to generate C++ code using these toolboxes and then generate the ROS node utilizing the generated C++ code. Nevertheless, numerous errors were encountered during compilation due to compatibility issues. Further investigation reveals that Matlab code generation considerations and standards were not considered when the TrueTime blocks were developed at Lund University [111], so the TrueTime blocks are not fully compatible with Matlab code generation. Consequently, Simulink coders have difficulty generating code for TrueTime blocks, especially for the TrueTime ttKernel blocks. Considering that eliminating errors was not trivial, we tried a different method of including Matlab code in the iterative scripts.

Parameters	Initials value	Range
$K_{P-Linear}$	7000	[100, 10000]
$K_{D-Linear}$	7000	[100, 10000]
$K_{I-Linear}$	7000	[100, 10000]
$K_{P-Angular}$	0	[0,5000]
$K_{D-Angular}$	0	[0,5000]
$K_{I-Angular}$	0	[0,5000]

Table 7.3 List of parameters that will be optimized using SMAC.

As a second alternative, we can simulate our model programmatically using Matlab Sim and Run commands, allowing us to integrate Matlab code into the iterative scripts more efficiently. To take advantage of this programming interface, installation and configuration of the Matlab Engine API are required. This API supports the reference implementation (CPython) and enables Matlab functions from Python. As a result, it is possible for our iterative script to access Matlab computation power from the Python script. The integrated iterative simulation framework will later be used for training the machine learning algorithms and reinforcement learning techniques.

It is necessary to optimize the controller parameters through SMAC in order to compare the performance of the different acoustic-based MAC protocols fairly. SMAC utilizes a random-forest-based Bayesian optimization algorithm to optimize the parameters. When obtaining cost function evaluations is costly [112], Bayesian optimization approaches are appropriate [28].

To use SMAC, it is necessary to provide the initial values, the desired ranges, the cost function, and the maximum number of simulation runs for each parameter that needs to be optimized. A maximum of 64 simulation runs were conducted in order to evaluate and obtain optimized parameters for each protocol and communication mode. Taking into account the thruster configuration and characteristics, Table 7.3 represents the range and initial values that are used as a starting point for optimization.

For example, based on the initial gain and optimal set, Figure 7.5 illustrates the AUV's trajectory. As shown in the Figure, the AUV cannot follow a smooth trajectory with the initial set of parameters (green trace). On the other hand, using the gains optimized through SMAC (red line), the AUV perfectly matches the desired path.

In this and the following two sections, we describe the results for three communication protocols where the gains are optimized through the SMAC algorithm. Using NCS to implement a docking maneuver, we find that system performance is sensi-

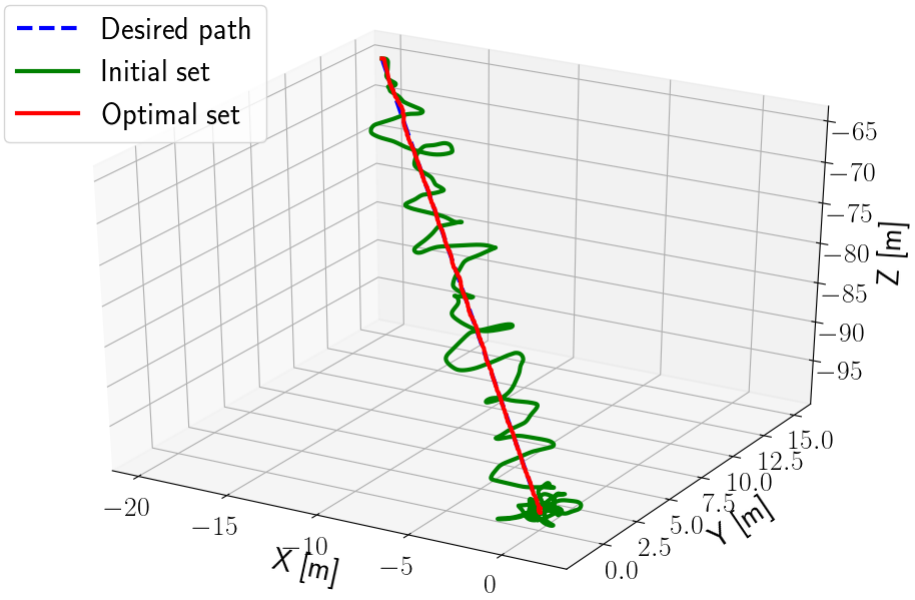


Figure 7.5 Difference between initial set and achieved optimal controller gain for AUV navigation.

tive to end-to-end delay in NCS, which is closely related to the sampling time of the simulation. Different protocols experience different amounts of delay. In fact, as mentioned above, we aimed to utilize SMAC for different communication protocols (such as S-ALOHA, TDMA, and WR) and communication schemes (i.e., half duplex and full duplex) for generating various sets of optimized controller gains. Based on the controller gains calculated for different communication protocols, we can examine the effect of each protocol on the performance of the NCS. It is assumed for the following experiments that the AUV is located at the following arbitrary initial position [-20, 15, -65].

7.3.1 TDMA Protocol

The time spent for AUV to complete the docking maneuver using TDMA protocol is depicted in Figure 7.6. This figure confirms the similar trends that have been observed previously, namely that our proposed FD hybrid has the shortest docking time. Close to this is the HD hybrid, which docks in approximately 108 seconds. Acoustic FD and HD take the longest time to dock, averaging 146 and 162 seconds, respectively. In contrast, the corresponding motive power of an AUV using TDMA is shown in Figure 7.6, which demonstrates that the HD hybrid consumes more power than other modes of communication while docking.

Figure 7.8 illustrates the effect of applying disturbance to the system by increasing water current velocity. In spite of the fact that docking time was improved in TDMA compared with the WR, the results are relatively similar to those in S-ALOHA. As

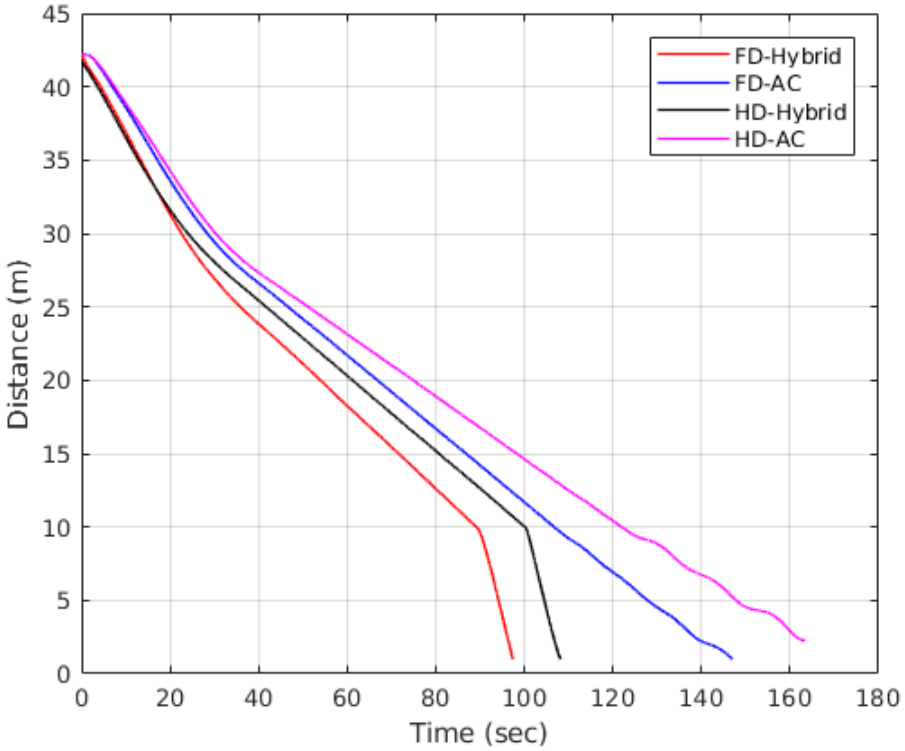


Figure 7.6 Time to dock of different communication modes using SMAC optimized PID employing TDMA protocol.

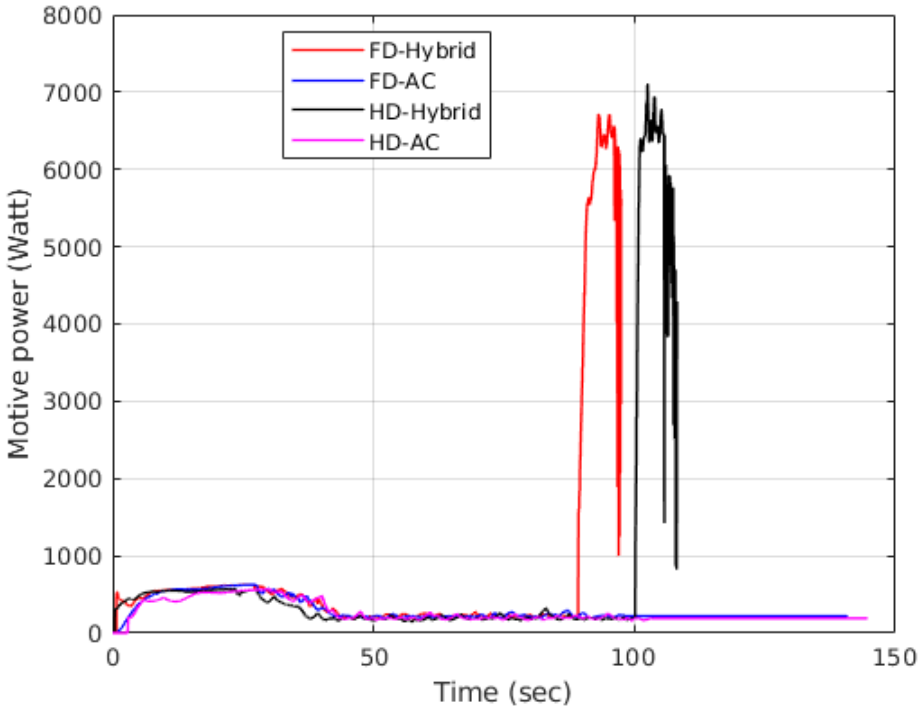


Figure 7.7 Motive power of different communication modes using SMAC optimized PID employing TDMA protocol.

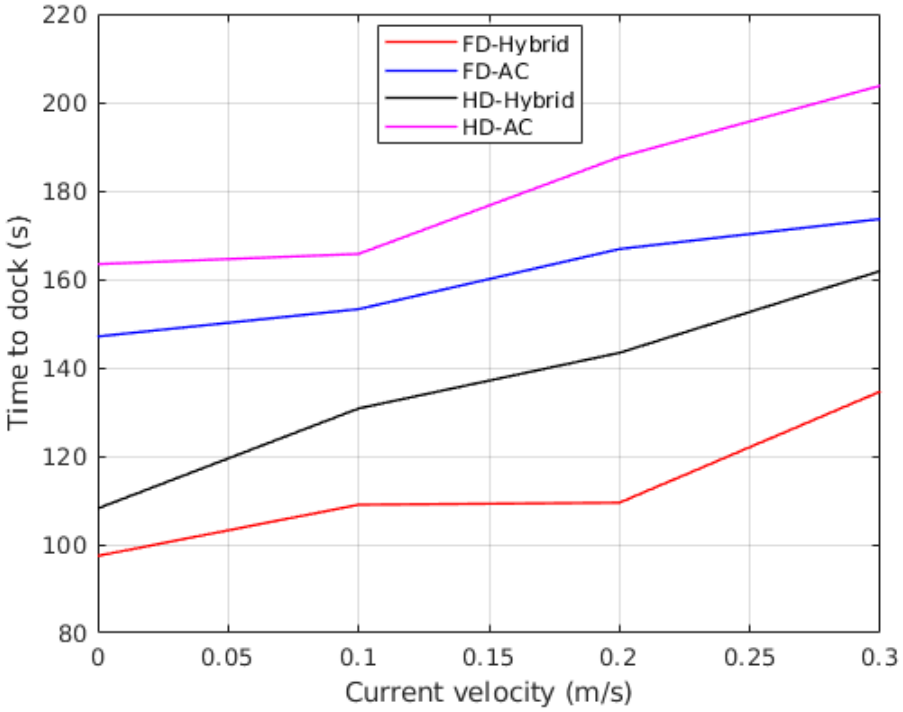


Figure 7.8 Time to dock by increasing current velocity using SMAC optimized PID employing TDMA protocol.

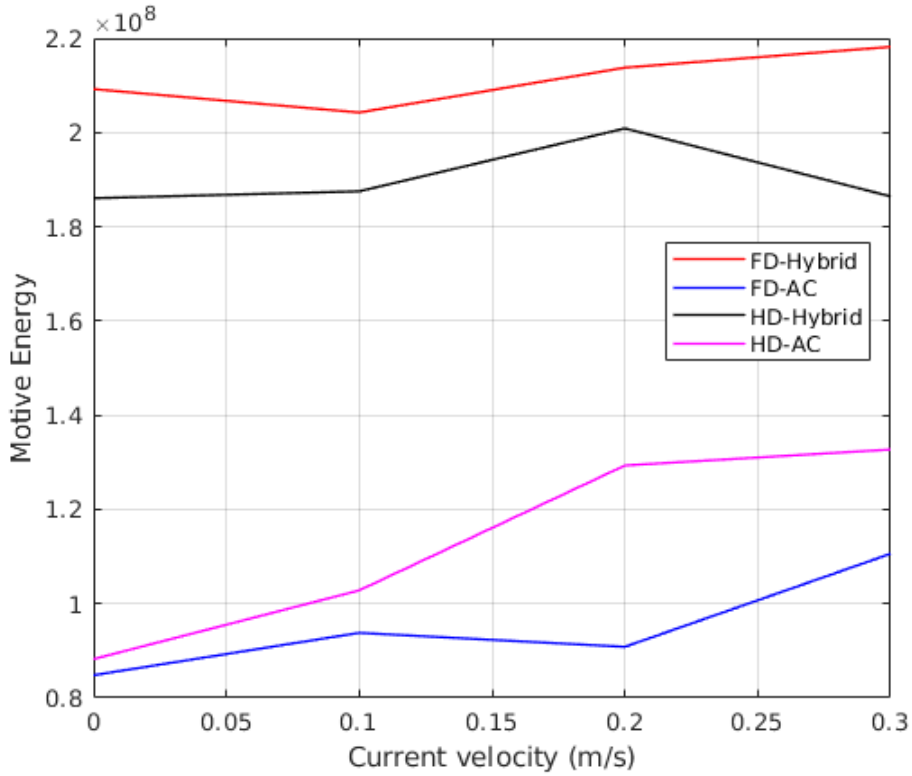


Figure 7.9 Motive energy by increasing current velocity using SMAC optimized PID employing TDMA protocol.

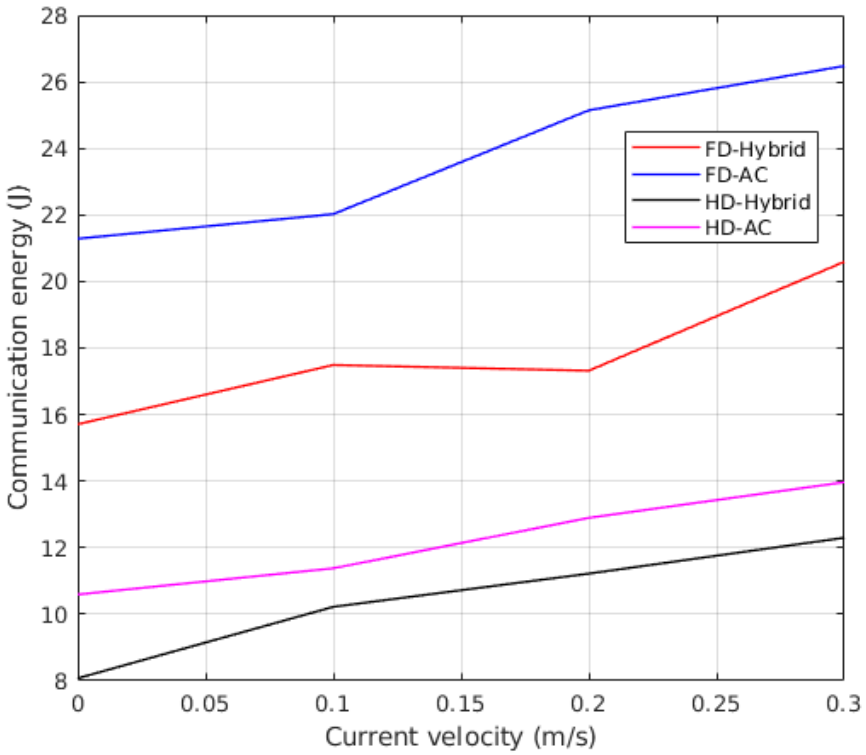


Figure 7.10 Communication energy by increasing current velocity using SMAC optimized PID employing TDMA protocol.

expected, the FD hybrid consumed more power in order to control the vehicle and dock swiftly. Thus, it has the greatest motive power as shown in Figure 7.9.

Receiving more packets in FD mode and hybrid mode increases communication energy, as illustrated in Figure 7.10. However, the more frequent packet reception allows AUV to have less cumulative error as shown in Figure 7.11. In other words, because FD hybrid consumes the most energy in communication, it has the shortest cumulative error.

7.3.2 Slotted ALOHA (S-ALOHA) Protocol

In a similar manner to the TDMA protocol, following the optimization of the controller gains for S-ALOHA, the following experiments were conducted to assess the protocol's performance.

According to Figure 7.12, docking time for an AUV using the S-ALOHA protocol depends on the mode of communication. FD Hybrid takes approximately the same amount of time to dock as WR's; however, for acoustic only, the docking time has been reduced from approximately 170 seconds to roughly 150 seconds. Although hybrid methods have higher motive power than acoustic ones, FD hybrid uses less

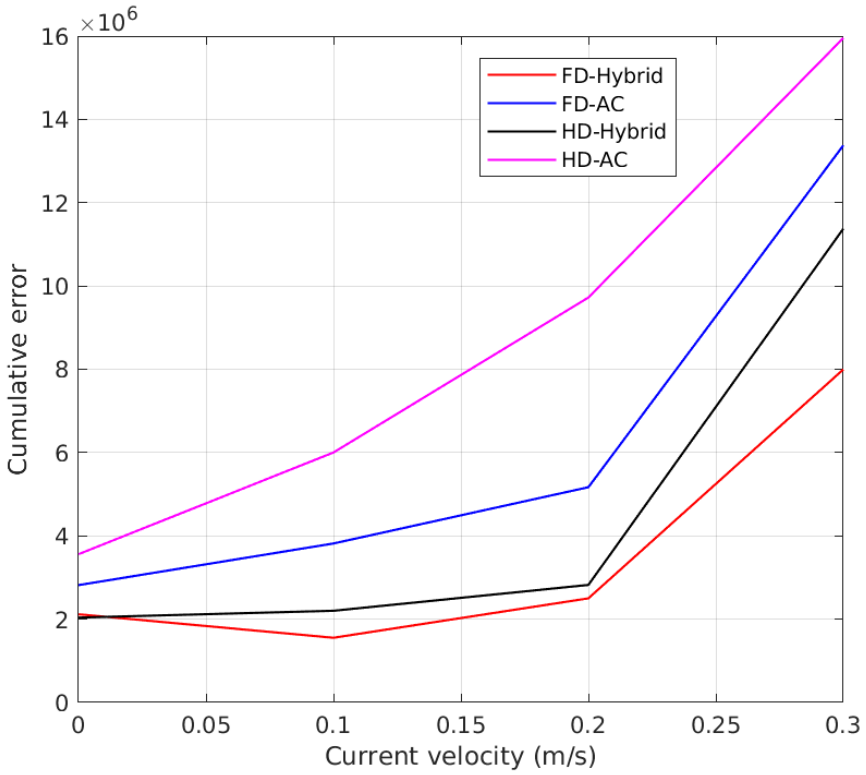


Figure 7.11 Cumulative error by increasing current velocity using SMAC optimized PID employing TDMA protocol.

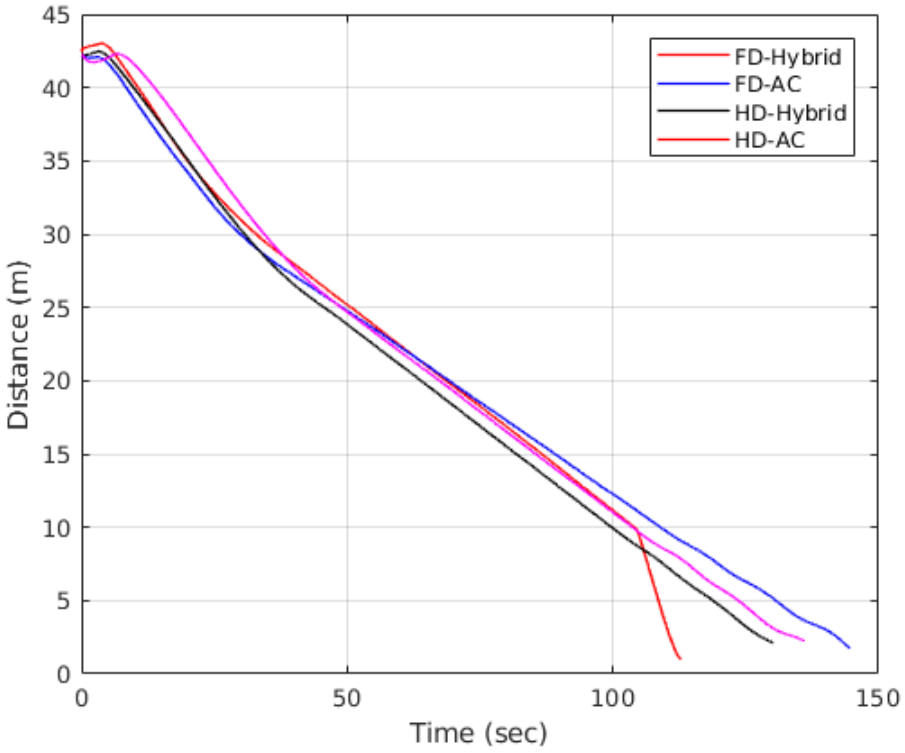


Figure 7.12 Time to dock of different communication modes using SMAC optimized PID employing S-ALOHA protocol.

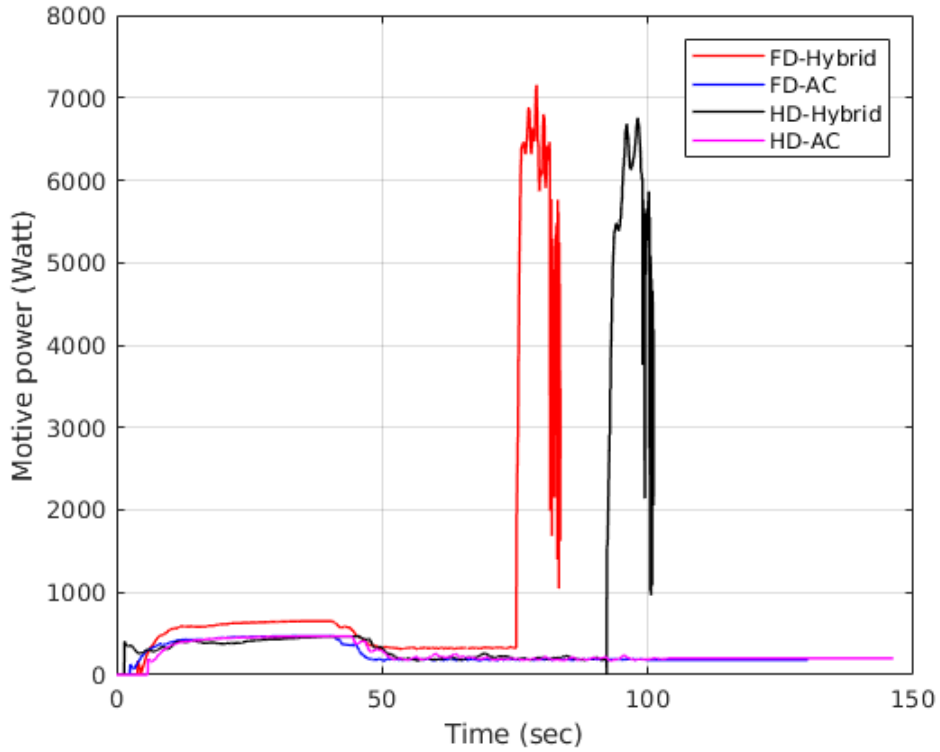


Figure 7.13 Motive Power of different communication modes using SMAC optimized PID employing S-ALOHA protocol.

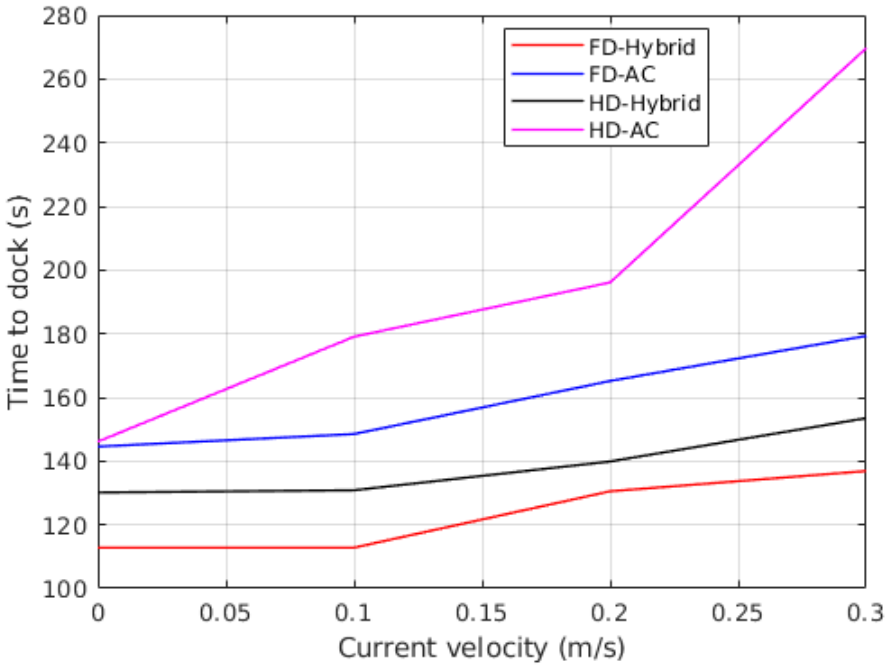


Figure 7.14 Time to dock by increasing current velocity using SMAC optimized PID employing S-ALOHA protocol.

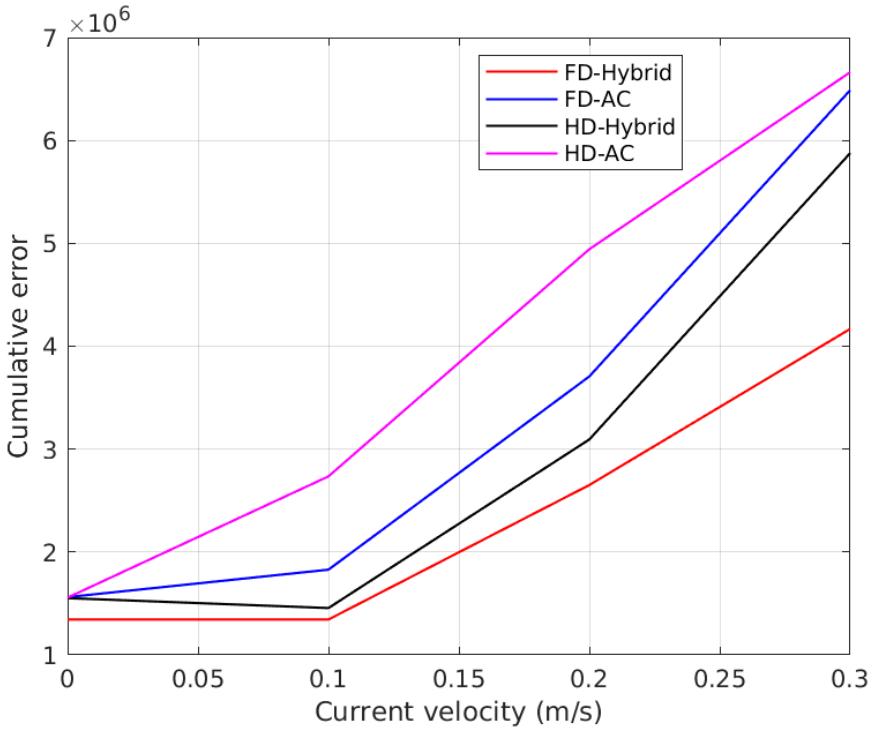


Figure 7.15 Cumulative error by increasing current velocity using SMAC optimized PID employing S-ALOHA protocol.

motive power than HD hybrid, as illustrated in Figure 7.13. In this case, energy is saved because the docking time is shorter, which allows the vehicle to save energy.

Analyzing the results depicted in Figure 7.14 in comparison to the corresponding WR protocol (i.e., Figure 7.19) results in a more favorable performance for S-ALOHA. According to Figure 7.14, when water currents with a speed of $0.3m/s$ are applied, the AUV can reach the docking station in approximately 140 seconds using the FD hybrid method, and this parameter is about 200 seconds using WR protocol. This is also the case for other communication modes using S-ALOHA protocol. Consequently, the cumulative error of S-ALOHA is lower than that of the WR protocol, as illustrated in Figure 7.15, since the docking maneuver is performed more quickly. Moreover, from Figure 7.16, it can be inferred that the FD approach consumes a greater amount of communication energy as compared to the HD one. It is primarily due to the fact that more packets are being transmitted. Hybrid methods, on the other hand, consume less energy during communication than acoustic only methods.

7.3.3 Waiting Room (WR) Protocol

The following plots represent the investigation of various criteria using WR protocols. Based on different communication modes, Figure 7.17 represents the docking time of the vehicle in calm water. Regarding docking time, the traditional acoustic

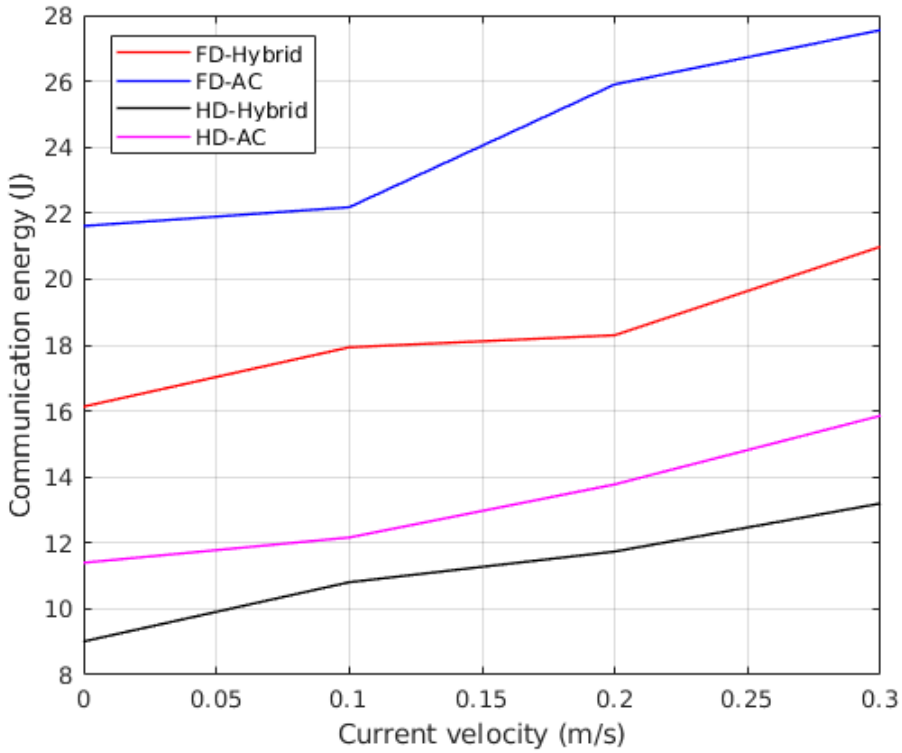


Figure 7.16 Communication energy by increasing current velocity using SMAC optimized PID employing S-ALOHA protocol.

HD method has the longest docking time of about 172 seconds. By enabling FD communication on the links, it is possible to reduce the docking time to approximately 145 seconds. Nevertheless, when using hybrid communication, switching to the RF link improves the slope of reaching the docking station for both hybrid HD and FD schemes.

In this manner, shorter docking times can be achieved using FD hybrid schemes by reducing the delay on NCS by receiving faster navigation data. Thus, a sharp increase in motive power can be observed during the switch from the low gain to the high gain controller, depicted in Figure 7.18.

In order to evaluate the proposed communication framework in the context of a real-life scenario, the proposed docking scenario is evaluated in the presence of water currents. Figure 7.19 illustrates that hybrid FD and HD have the shortest docking time when the current velocity is increased. There are significant differences between an acoustic based scheme and a hybrid counterpart at higher velocities than $0.2m/s$, and for velocities higher than $0.3m/s$ the acoustic based scheme is unable to dock successfully.

In contrast, the motive energy of different schemes becomes a trade-off. Although

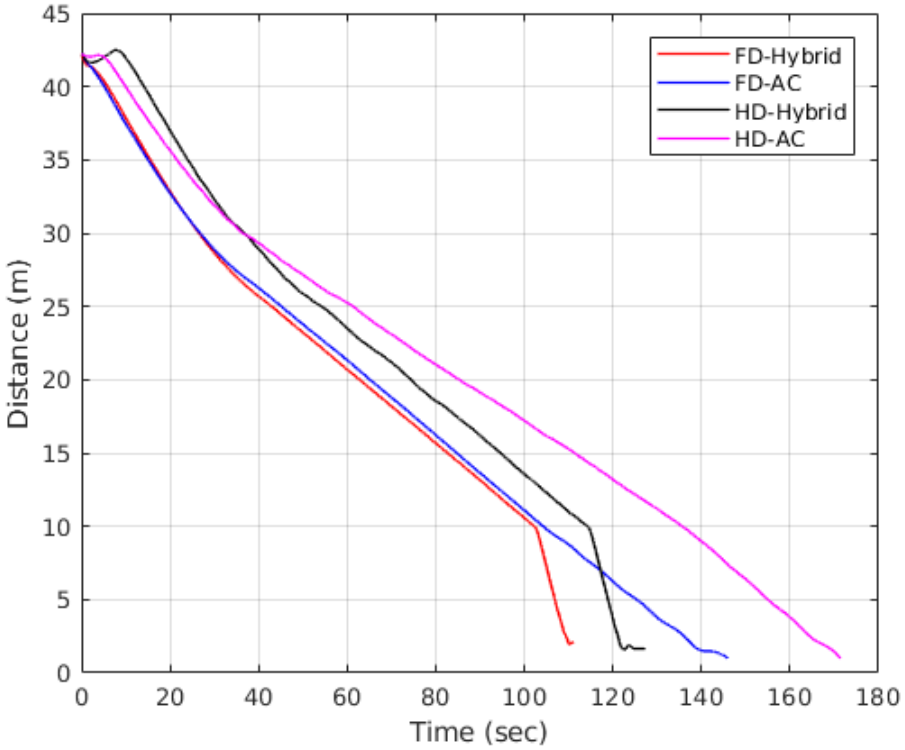


Figure 7.17 Time to dock of different communication modes using SMAC optimized PID employing WR protocol.

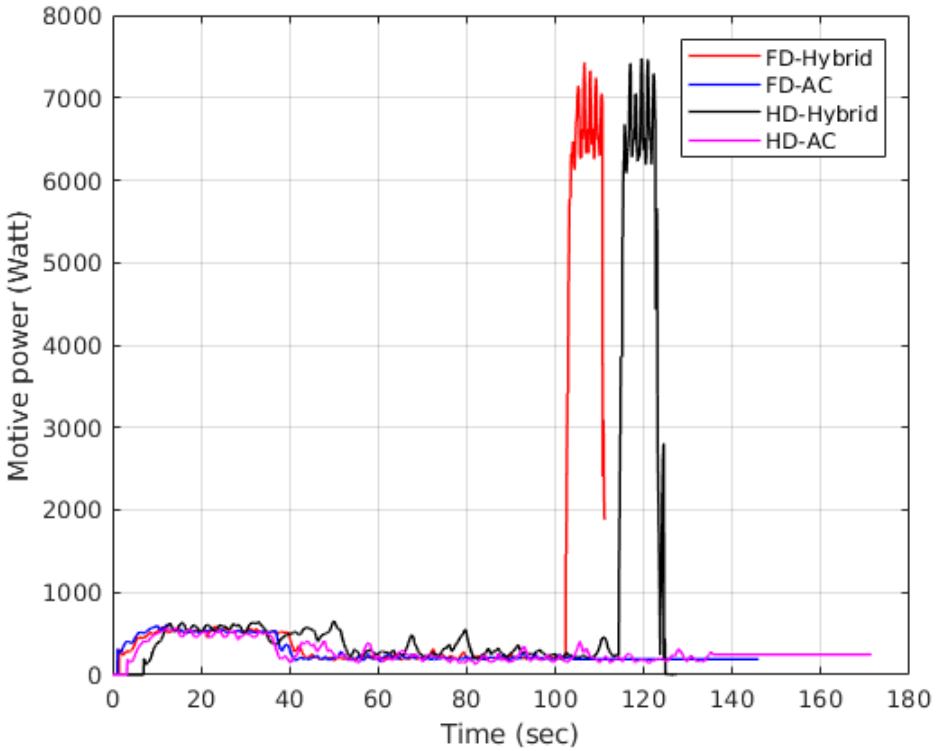


Figure 7.18 Motive Power of different communication modes using SMAC optimized PID employing WR protocol.

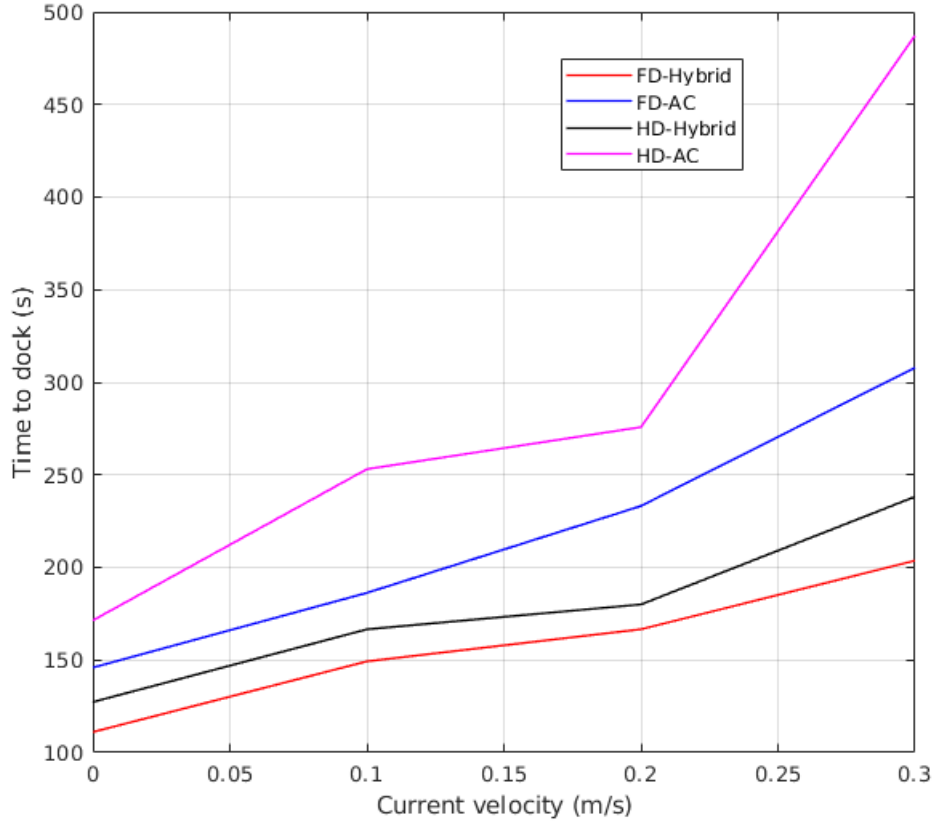


Figure 7.19 Time to dock by increasing current velocity using SMAC optimized PID employing WR protocol.

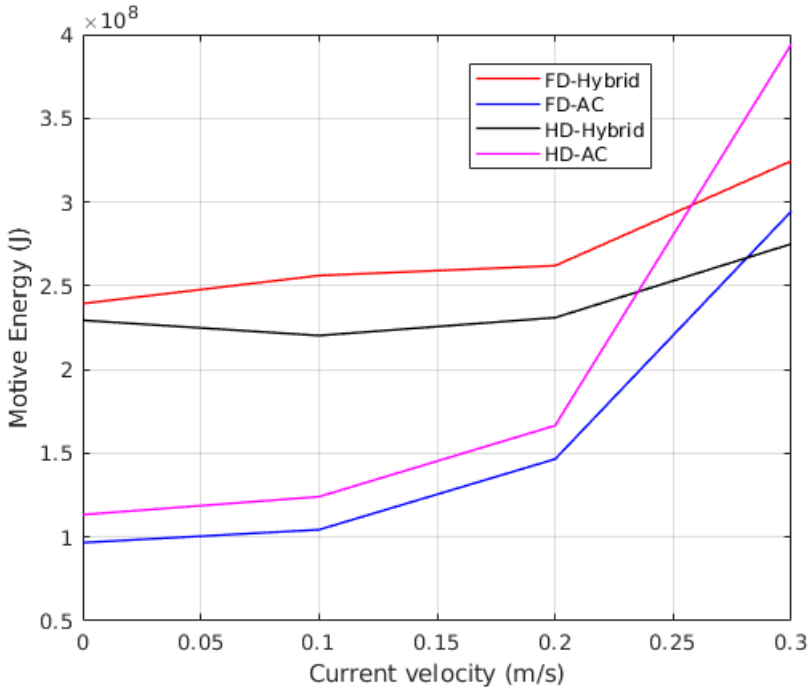


Figure 7.20 Motive energy by increasing current velocity using SMAC optimized PID employing WR protocol.

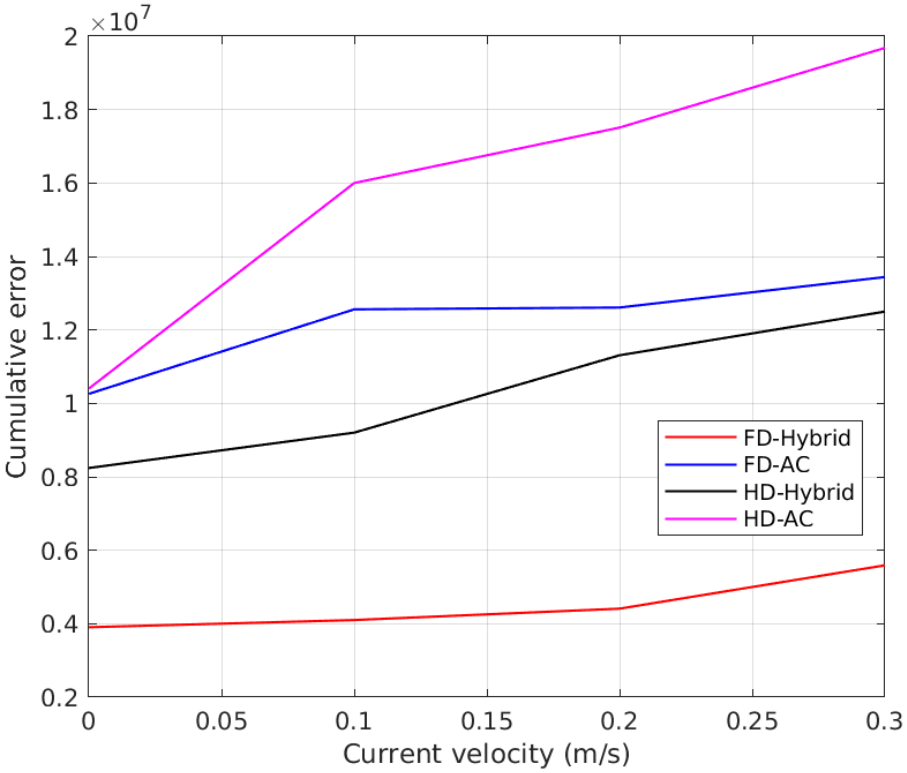


Figure 7.21 Cumulative error by increasing current velocity using SMAC optimized PID employing WR protocol.

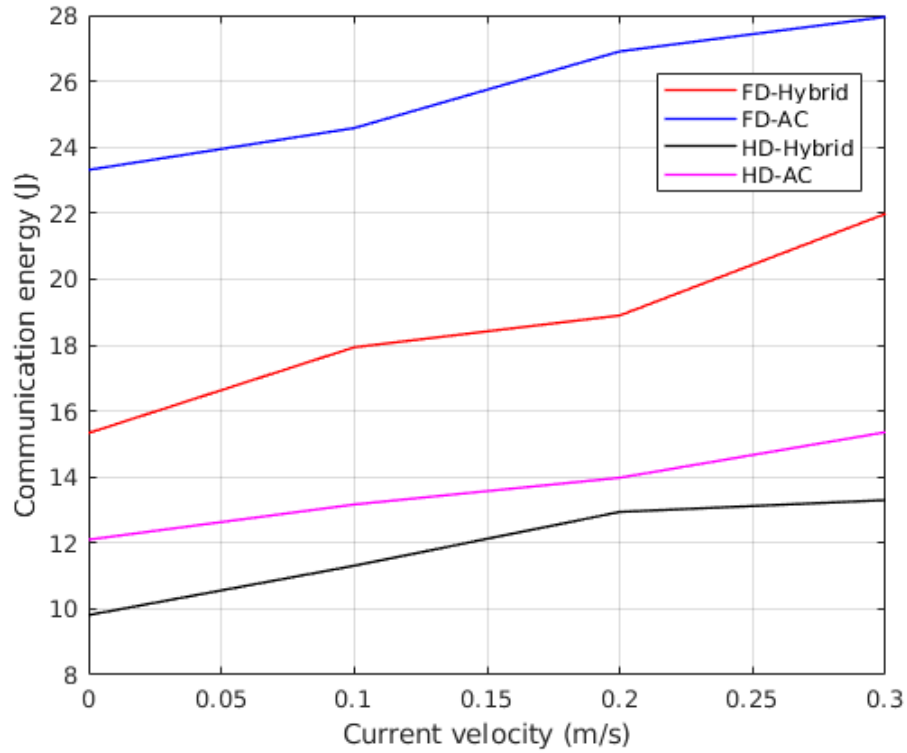


Figure 7.22 Communication energy by increasing current velocity using SMAC optimized PID employing WR protocol.

acoustic-based approaches consume less motive power in low-speed velocities, they take longer to dock and, as a result, have a higher overall motive energy consumption as depicted in Figure 7.20.

The amount of energy AUVs consume during the mission is depicted in Figure 7.22. In contrast to HD acoustic, which consumes the least energy for communication, the other schemes use considerable energy for communication by increasing the speed.

Finally, the cumulative error during the mission is represented in Figure 7.22. The hybrid method is more effective in steering the vehicle in harsh conditions, thus resulting in a lower cumulative error compared to the acoustic approach.

7.4 FD Hybrid NCS with LQR

Previously, we discussed the importance of adjusting a controller to fairly compare the alternative communication protocols to navigate the AUV with minimal tuning. Hence, the LQR controller is designed and implemented into the project to adjust the gains automatically.

Furthermore, according to the experiment conducted in section 7.2, among the three network protocols discussed earlier to utilize in NCS, the TDMA and S-ALOHA protocol provides better results than the WR protocols. Consequently, to evaluate the system performance for the remaining experiments, we have decided to employ TDMA protocol and investigate the effects of changing communication modes, namely half duplex (HD) acoustic only, full duplex acoustic only, HD hybrid, and FD hybrid under LQR control.

Based on our proposed FD hybrid NCS, Figure 7.23 illustrates the results of navigating AUV using the LQR controller. Upon receiving the docking maneuver command, the underwater vehicle employs the embedded LQR controller to steer the vehicle toward the target position. It is observed that the trajectory is smooth during cruise time due to the high control gain of the LQR controller.

Figure 7.24 illustrates the time taken by the AUV to complete the docking maneuver using the LQR controller in calm water. It takes approximately 80 seconds for an acoustic only maneuver to be completed, whereas the hybrid HD maneuver reduces docking time by 13% and reaches the station in roughly 72 seconds. By enabling FD communication between AUVs and docking station, landing times can be shortened to about 62 and 75 seconds for hybrid and acoustic docking stations, respectively. Accordingly, compared to the conventional acoustic only method, the proposed FD hybrid docking method reduces docking time by 25%. This reduction in the FD

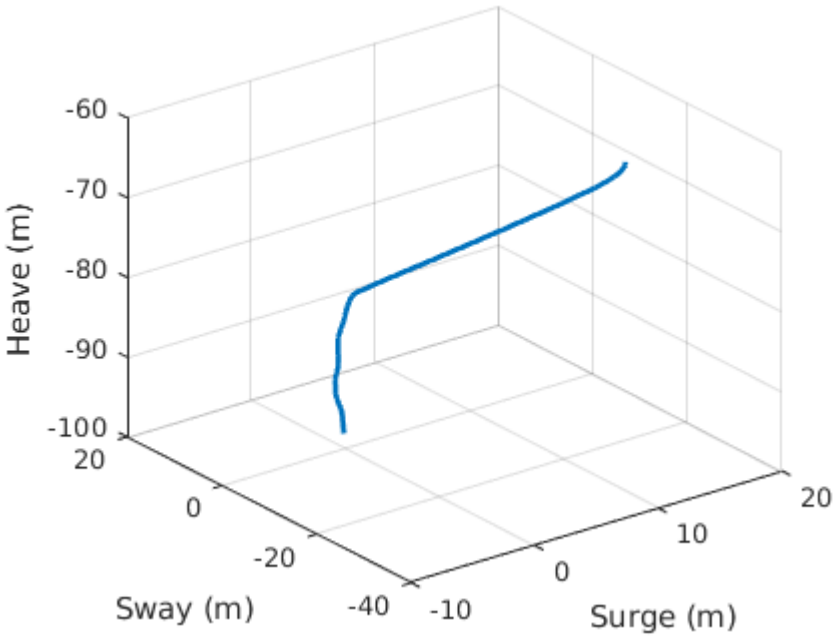


Figure 7.23 The generated trajectory of AUV during docking maneuver using LQR controller.

hybrid method is primarily due to a reduction of sampling time by activating RF and FD simultaneous communication and enabling the AUV to utilize a higher gain controller and provide the required force and torque more quickly. Since the AUV thrusters received faster steering commands, they spent more power to reach the target position. This can be verified by observing the soaring motive power for the FD hybrid and hybrid methods shown in Figure 7.25 which is remarkably higher than the acoustic based and half-duplex methods.

Our previous experiments using the LQR controller assumed that the scenario had been performed in calm water conditions. The effect of water currents and speed on the performance of the system will be examined in subsequent experiments in order to study its performance more realistically. The integration of UUV-Simulator plugin into our co-simulation environment allows water current velocity to be produced using different Gauss-Markov processes explained in Fossen's handbook [29].

In this way, it is possible to specify the magnitude, the horizontal angle, and the vertical angle of the current velocity before starting the simulation and during its running. This can be achieved by calling a ROS service and setting the desired velocity.

Each of the methods in Figure 7.26 shows a linear increase in the trend, and after increasing the velocity to $0.5m/s$, it shows an exponential increase slightly. Among all methods depicted in Figure 7.26 our proposed FD hybrid has the shortest time

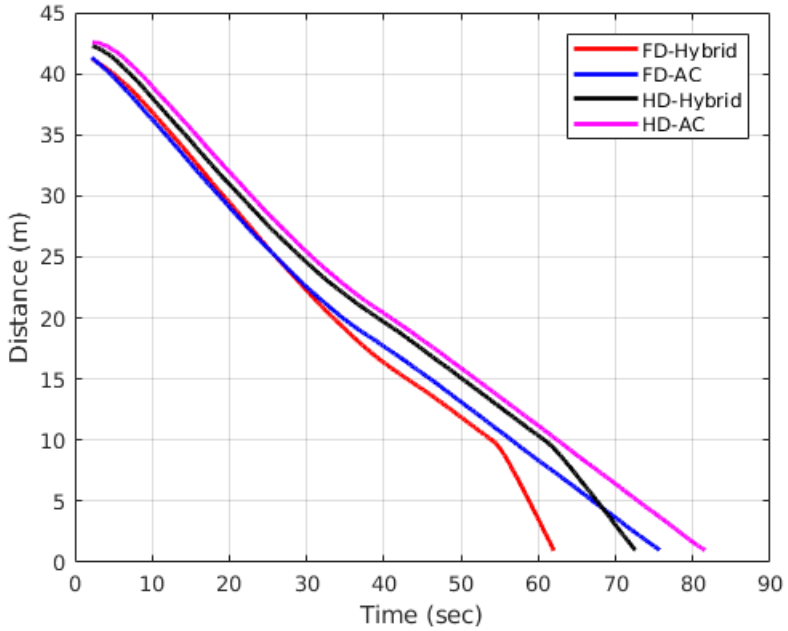


Figure 7.24 Comparison of docking times between different communication modes using LQR controller with TDMA protocol in calm water. (FD: Full duplex, HD: Half duplex)

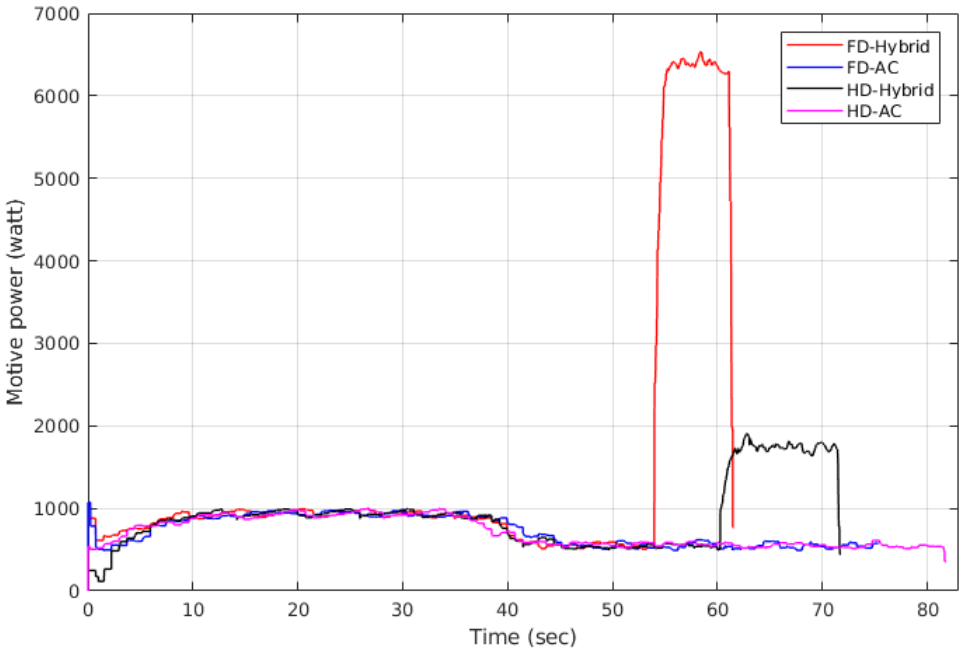


Figure 7.25 Comparison of the motive power of different communication modes using LQR controller with TDMA protocol.

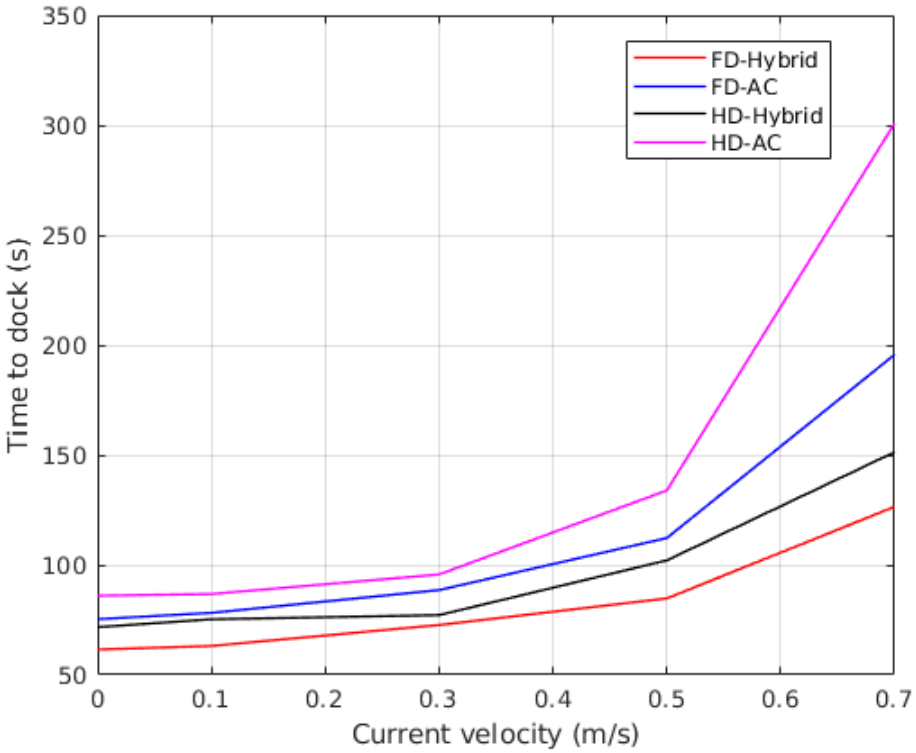


Figure 7.26 Time to dock by increasing current velocity using LQR controller employing TDMA protocol.

to dock. Unlike the conventional acoustic based underwater communication, (i.e. this is referred to as the HD acoustic method only in this thesis), which is unable to land on the docking station at a velocity of $0.7m/s$, other methods are able to land successfully on the station. A further observation that can be made from Figure 7.26 is a comparison of the performance of the LQR controller versus the conventional PID controller when it comes to tolerating the current in the water. According Figure 7.3, the PID-based controller cannot navigate the vehicle when the current velocity exceeds $0.35m/s$, while the LQR-based controller is capable of controlling the vehicle up to $0.7m/s$.

Figure 7.27 illustrates the amount of communication energy spent by different communication modes when the velocity of water currents is increased. The communication energy traces are approximately the same for both HD and FD hybrid methods, whereas the trend keeps rising for acoustic-based approaches. This can be explained by the fact that increasing the velocity of water currents and receiving position information with a delay using acoustic-based methods causes AUVs to struggle a lot to complete docking maneuvers. It involves the exchange of more messages and consequently, AUV spends more energy on high-speed currents.

It is important to note that while the proposed FD hybrid has the fastest per-

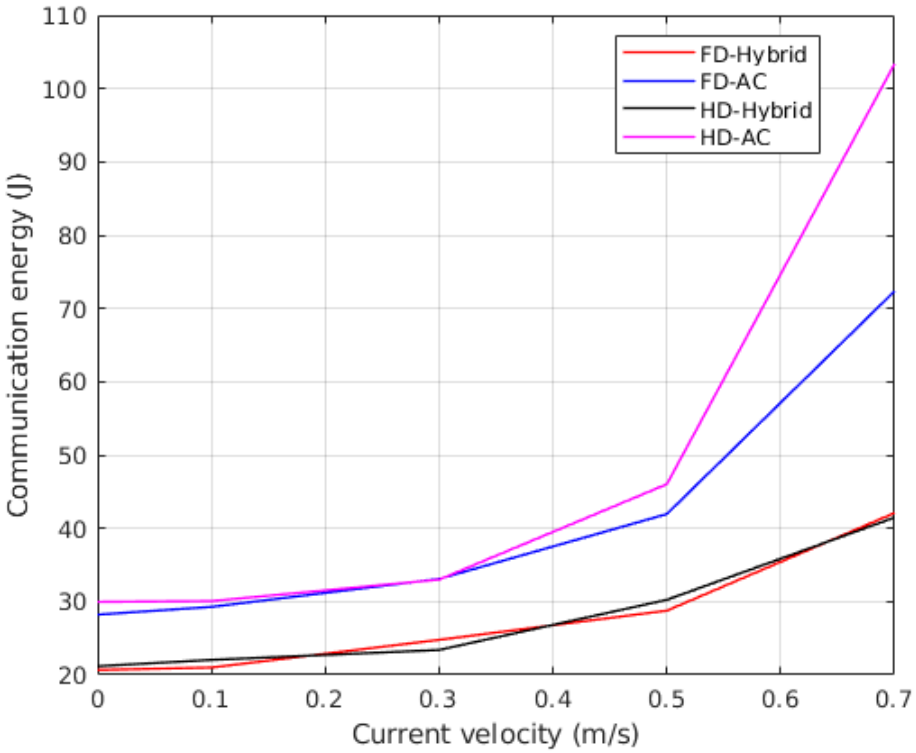


Figure 7.27 Communication energy by increasing current velocity using LQR controller employing TDMA protocol.

formance, it also consumes the most energy during docking operations. A steady increase in motive energy consumption occurs with increasing velocity up to 0.5 m/s regarding all the other methods except for the proposed FD hybrid mode, as shown in Figure 7.28. However, the plot shows a more rapid increase in motive energy usage for higher velocities.

Expanding RF Range using Repeaters

It is evident that the proposed hybrid method outperforms conventional acoustic-based communication for applications requiring precision. However, RF signals are limited in underwater environments due to high-frequency absorption resulting in shorter operation ranges. As a remedy, it is possible to use repeaters to amplify and relay the RF signal over a longer distance. To extend the range of RF communication, repeaters can be placed at regular intervals along the communication path. An AUV that can be controlled precisely during the mission can justify using several repeaters around the docking station.

Hence we aimed to expand the range of RF signals by considering various switching mechanisms which were introduced in section 6.4. By increasing the number of RF transmitters adjacent to the previously established range of RF transmitters, we

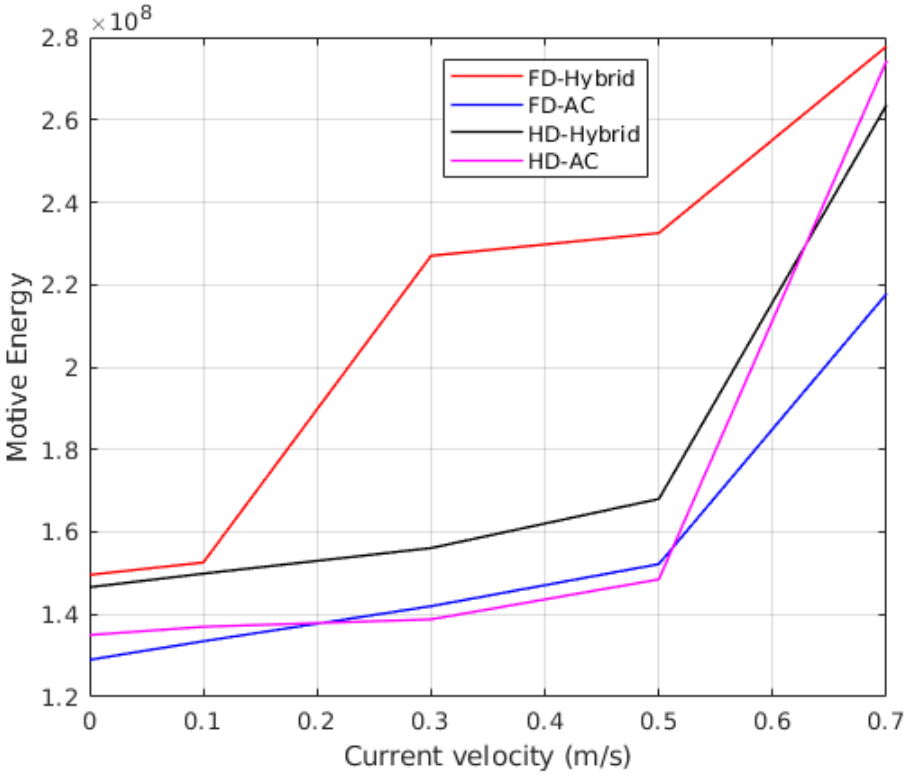


Figure 7.28 Motive energy by increasing current velocity using LQR controller employing TDMA protocol.

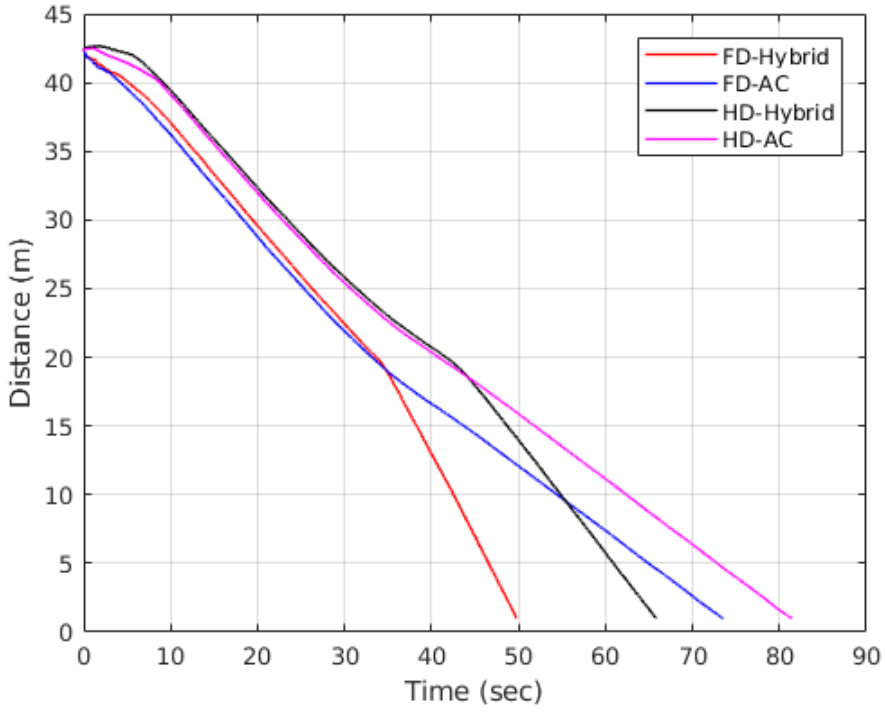


Figure 7.29 Effect of expanding RF range and its impact on docking times.

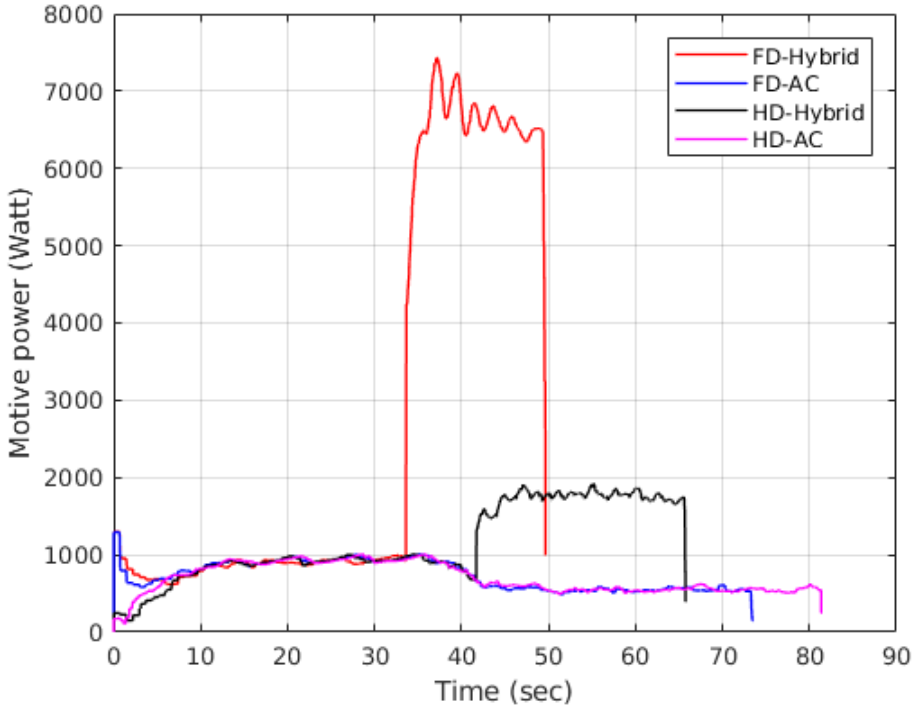


Figure 7.30 Effect of expanding RF range and its impact on motive power.

were able to accomplish our goal.

This will expand the operating range of underwater RF signals, resulting in a wider range of applications. Even though doubling the RF range may not seem comparable to the acoustic range, this modification will still provide ample amounts of operation range for the UUVs. This will enable underwater vehicles to perform their tasks more accurately in order to achieve their mission objectives. Besides, further extending RF ranges in the event of higher water currents helps ensure that these expensive UUVs do not collide with the docking station platform.

As shown in Figure 7.29, the improved switching mode reduces the docking time for our FD hybrid communication mode using the LQR controller by approximately 19%. Furthermore, as expected, docking time for acoustic based approaches has not been affected; however, docking time for HD hybrid communication has been reduced by 10% due to the improved switching algorithm.

In Figure 7.30, the motive power illustrates the cost of reducing docking time in the extended RF range scenario. It appears that the AUV spent slightly more motive power in the expanded version of the diagram than it did in Figure 7.25. However, since the time duration for the expended one is shorter than the other one, the motive energy will be close to each other if the instantaneous power is integrated over time.

Ultimately, the results that we obtained verify that using our proposed FD hybrid communication mode remarkably improves the performance of the system. Besides in cases where there are high water currents the acoustic based solution is not applicable.

7.5 Performance Under Realistic Water Currents

Previous experiments were conducted by applying current, considering horizontal angles, vertical angles and desired velocity. Throughout the simulation, the current was applied in only one direction. Nevertheless, it may not be the case for the oceans and sea currents, where velocity and direction fluctuate over time. By modeling and applying realistic water currents, we have evaluated its impact on the performance of NCS via evaluating different network schemes and SMAC optimized PID and LQR controllers.

In the following experiment, a large current with a velocity of $5m/s$ is generated in the surge direction between 10 to 40 seconds of simulation time. Immediately following this, a current with a velocity of $2m/s$ is applied at a negative sway direction for 20 seconds. The effect of incorporating this realistic current is illustrated in all figures of this section (e.g., Figure 7.31, the blue trend in the plot) at the periods mentioned above.

Following our expectations, applying the realistic current will increase the time to dock for all modes of communication using the SMAC optimized PID controller, as shown in Figures 7.31, 7.33, and 7.35. These figures demonstrate that the time to dock for the proposed FD hybrid mode, using TDMA, S-ALOHA, and WR, is approximately 175s, 180s, and 195s, respectively.

Furthermore, all three protocols exhibited similar motive power levels, approximately 6500 Watts in Figures 7.32, 7.34, and 7.36

In the following experiments, we evaluated the performance of LQR with different network protocols under realistic circumstances, as shown in Figures 7.37. For the acoustic only method, the docking time is affected by 50%, whereas for other methods, the increased time is approximately 35%. When comparing the realistic results of LQR based modes with SMAC optimized PID based modes, it is evident that the time to dock using LQR based modes methods is nearly half that of SMAC optimized PID counterparts.

In consequence, Figure 7.38 depicts the effect of realistic current on motive power and illustrates that FD hybrid vehicles consume the highest amount of motive power.

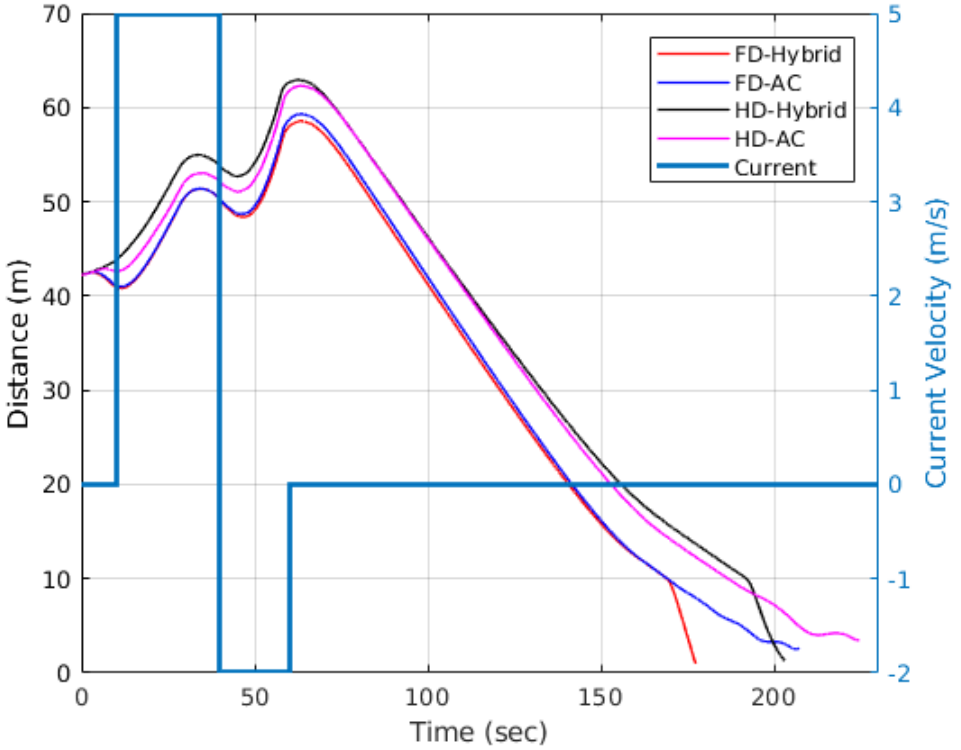


Figure 7.31 Time to dock using SMAC optimized PID employing TDMA protocol for realistic water currents.

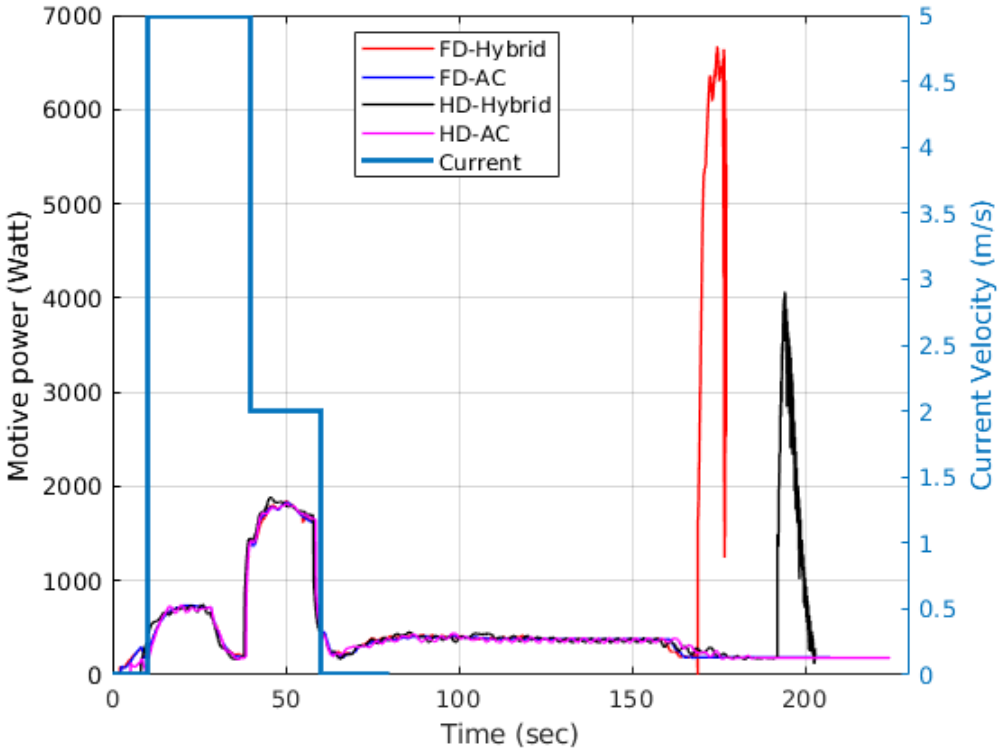


Figure 7.32 Motive power using SMAC optimized PID employing TDMA protocol for realistic water currents.

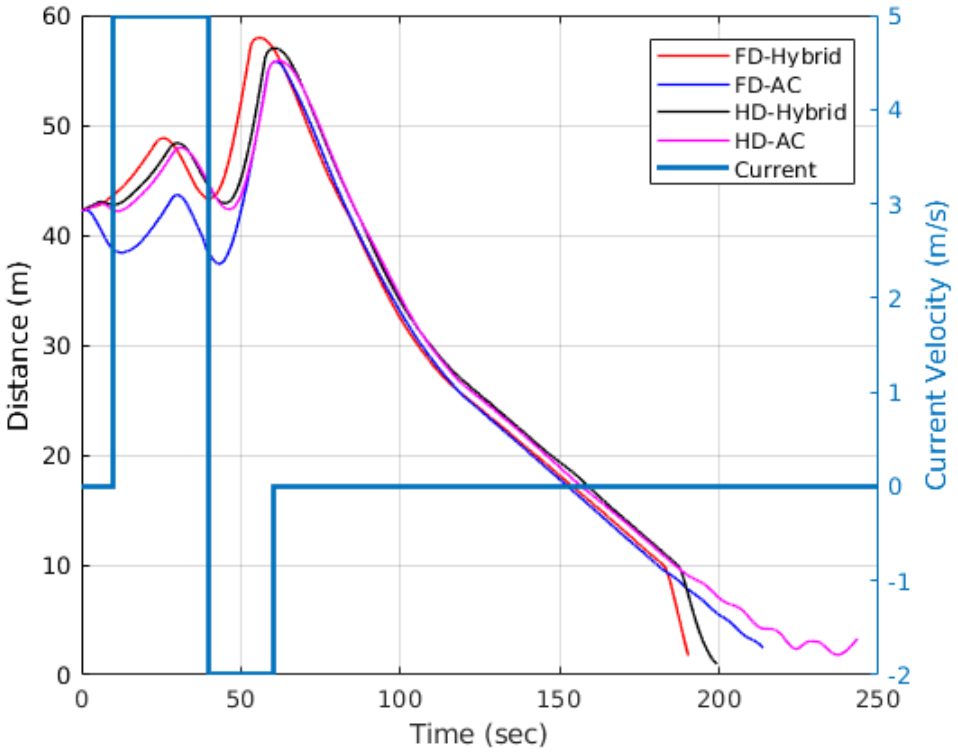


Figure 7.33 Time to dock using SMAC optimized PID and S-ALOHA protocol for realistic water currents.

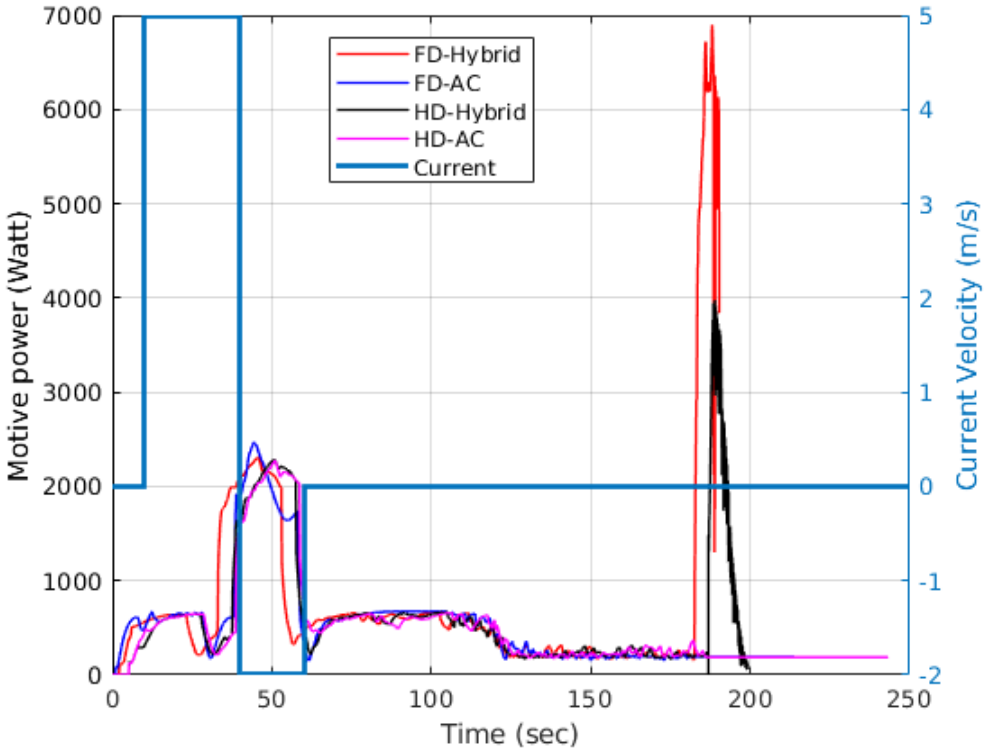


Figure 7.34 Motive power using SMAC optimized PID and S-ALOHA protocol for realistic water currents.

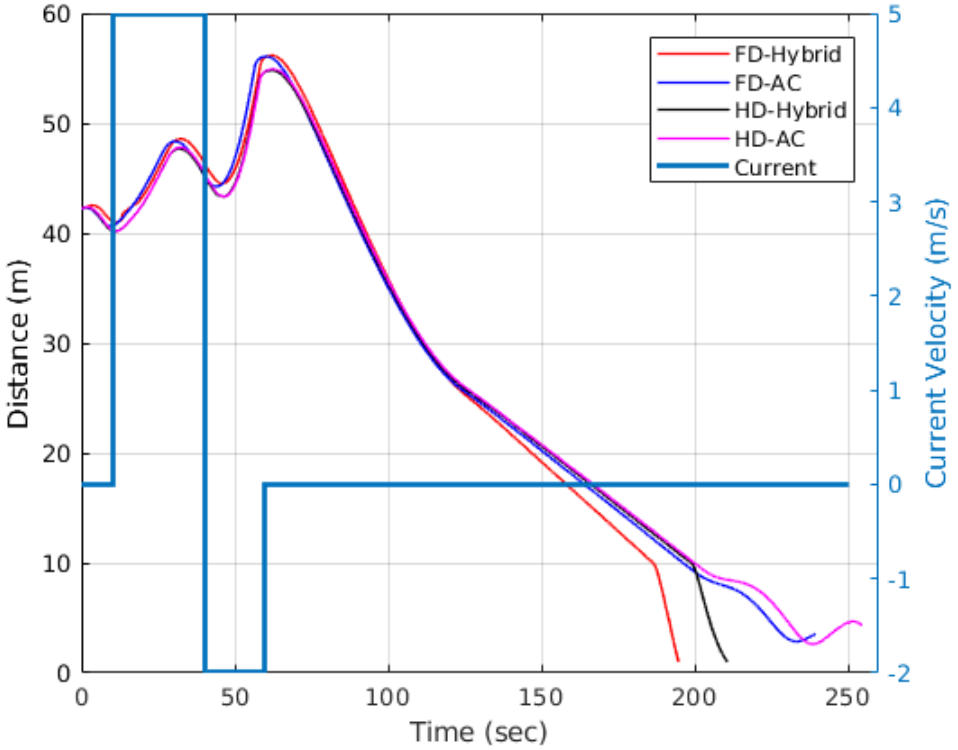


Figure 7.35 Time to dock using SMAC optimized PID and WR protocol for realistic water currents.

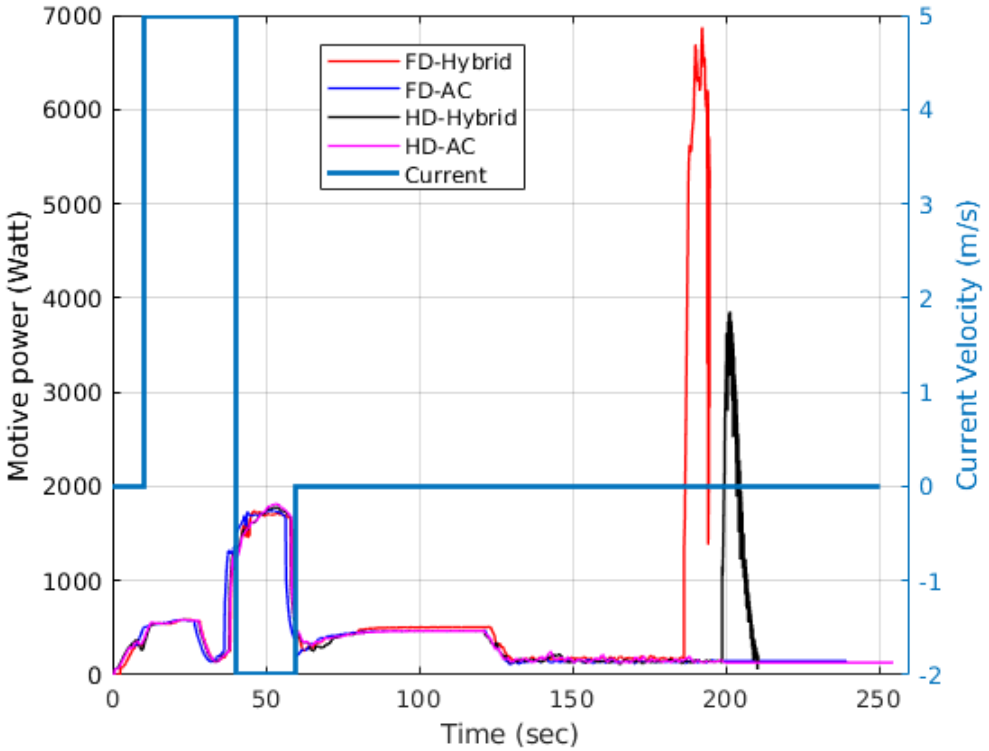


Figure 7.36 Motive power using SMAC optimized PID and WR protocol for realistic water currents.

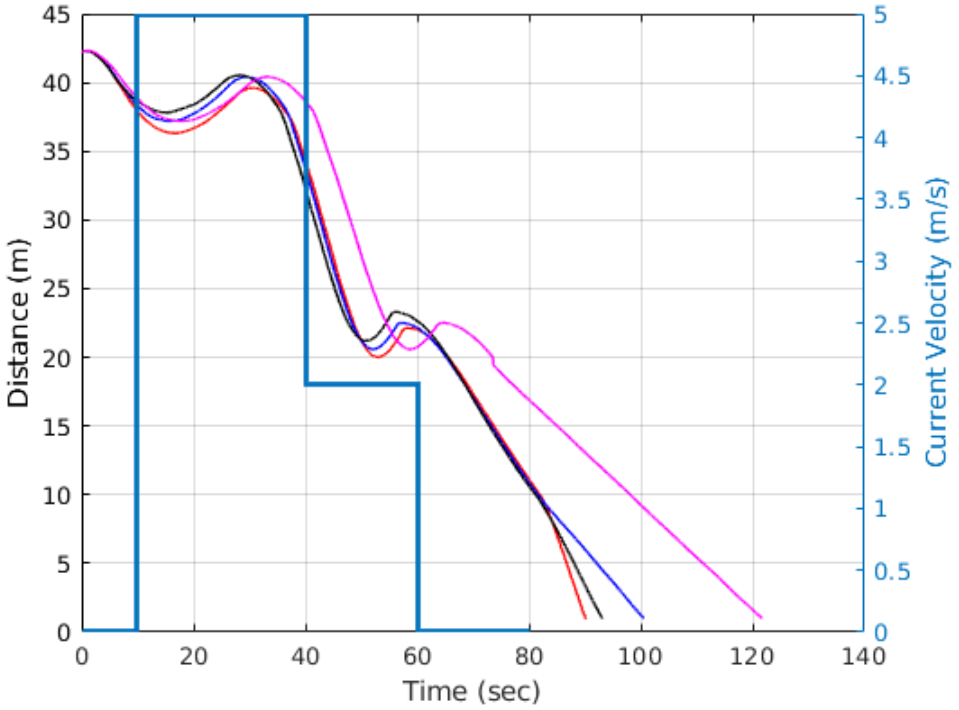


Figure 7.37 LQR time to dock with realistic water currents.

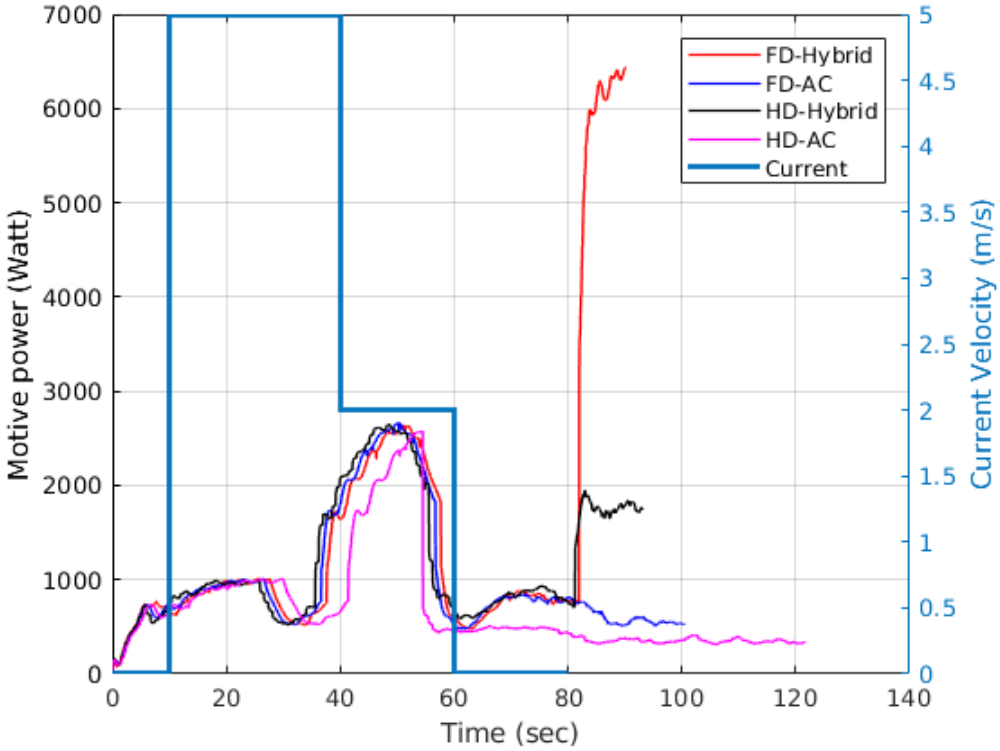


Figure 7.38 LQR motive power with realistic water currents.

It is also notable that comparing realistic motive power results reveal a similarity between LQR-based communication modes and SMAC optimized PID modes. This is mainly due to the trade-off between higher energy consumption over a short period, resulting in similar power consumption for LQR compared to SMAC optimized PID methods. Therefore, this finding can mitigate LQR's drawback of higher energy consumption.

7.6 Comparison of SMAC Optimized PID and LQR controllers

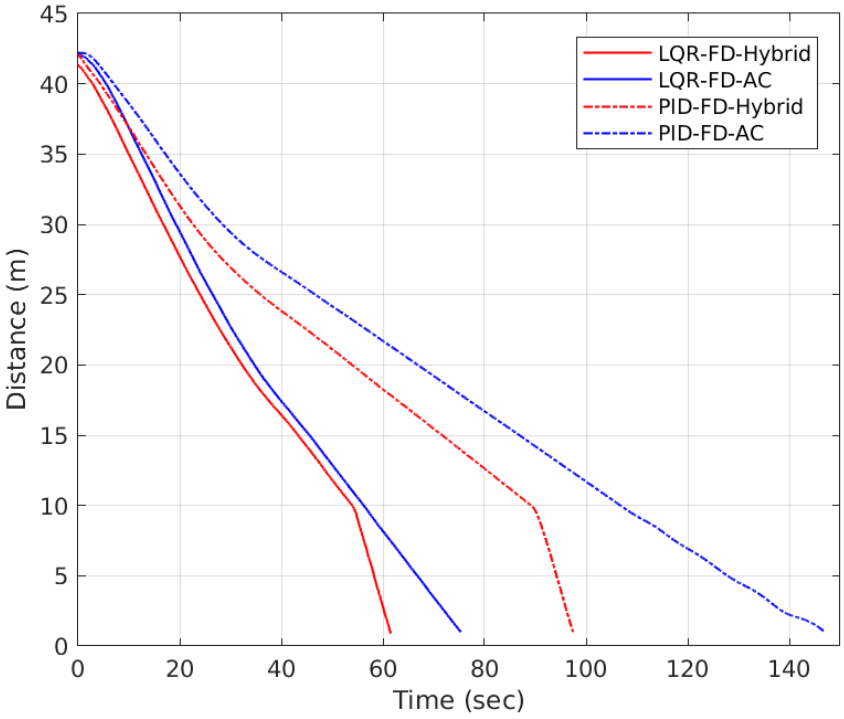
In this section, we evaluate the performance of LQR and SMAC optimized PIDs under calm and increased current velocity scenarios. To achieve this, we compare two controllers with hybrid and acoustic-only approaches using FD and HD communication modes, as depicted in Figure 7.39, parts (a) and (b).

According to part (a)of Figure 7.39, the proposed FD hybrid approach using LQR achieves the shortest docking time of approximately 62 seconds, while the corresponding SMAC optimized PID approach in FD mode takes around 97 seconds. The difference is even more significant when comparing two controllers using acoustics in FD mode. The LQR-based docking time is about 78 seconds, while the PID-based docking time is approximately 148 seconds, representing a 90% increase in docking time.

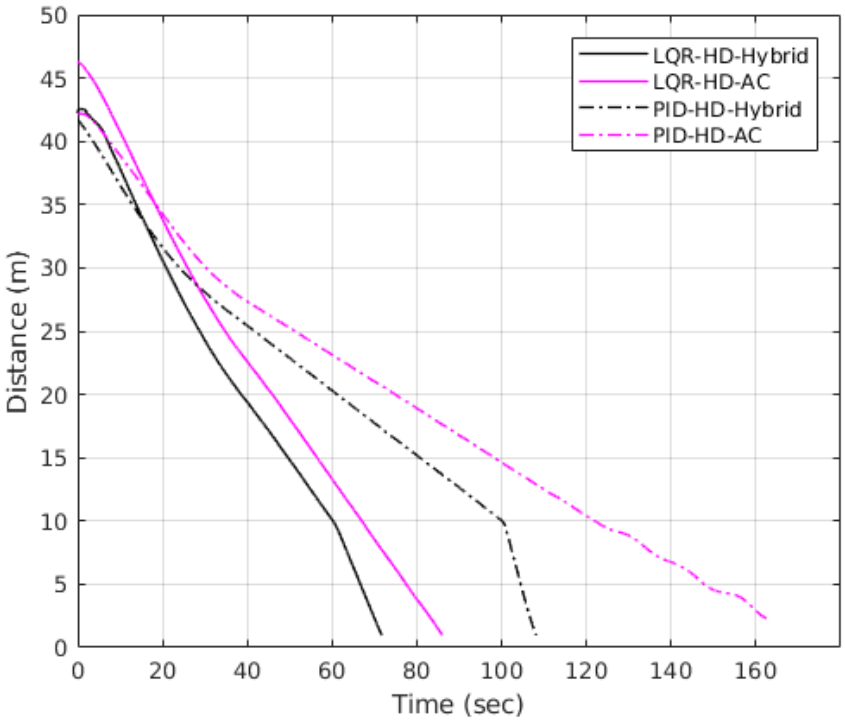
Furthermore, in HD communication, the difference between LQR and SMAC optimized PID using hybrid and acoustic modes is approximately 45% and 90%, respectively, as illustrated in part b of Figure 7.39. Consequently, the LQR controller significantly outperforms the SMAC optimized PID in HD and FD modes.

Figure 7.40, part (a), shows that both LQR and SMAC optimized PID controllers consume almost the same motive power in FD mode. However, when using the HD mode in part (b) of Figure 7.40, the power required for LQR to accomplish the mission is considerably reduced compared to that required for SMAC optimized PID control. Therefore, there is a noticeable difference in the motive power consumption between the two approaches in HD mode but not in FD mode.

Figure 7.41 illustrates the evaluation of two controllers for docking maneuvers in varying water current velocities. The LQR based method is found to have a shorter docking time than the SMAC optimized PID method, up to a current velocity of $0.3m/s$. Additionally, using both controllers, our proposed FD hybrid controller has the fastest docking time. Moreover, the proposed HD hybrid approach came in

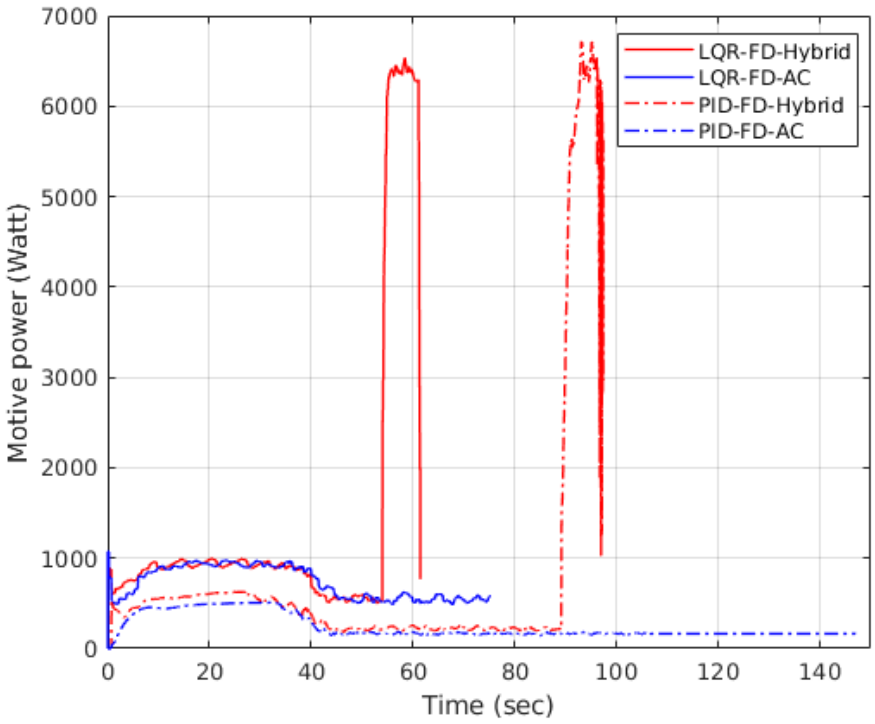


(a) Full duplex

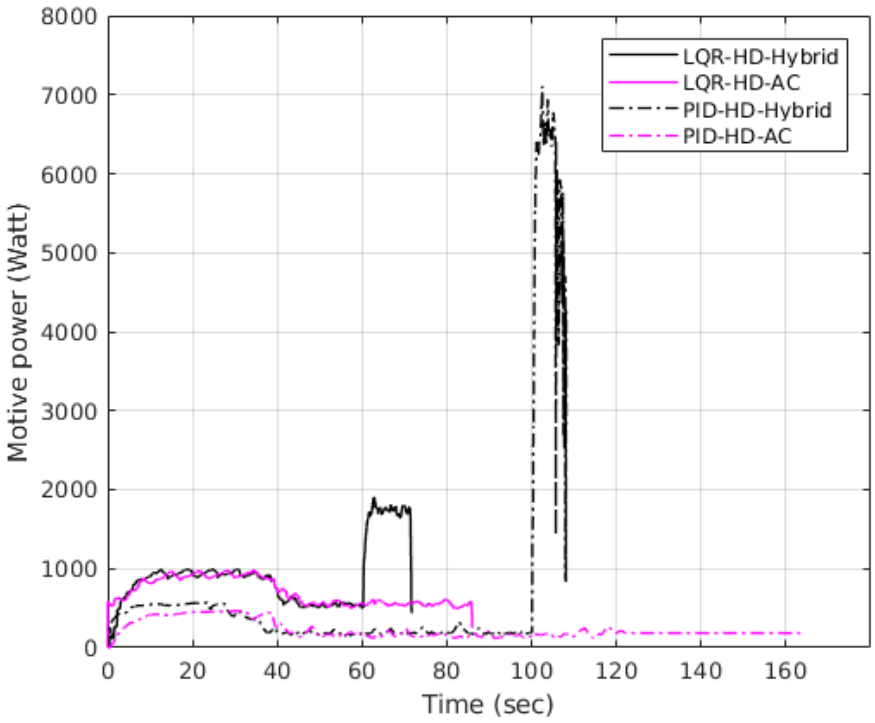


(b) Half duplex

Figure 7.39 A comparison of the time to dock of the AUV using LQR and SMAC optimized PID controllers using the TDMA protocol with FD and HD communication modes. Calm water.



(a) Full duplex



(b) Half duplex

Figure 7.40 A comparison of the AUV motive power consumption using LQR and SMAC optimized PID controllers using the TDMA protocol with FD and HD communication modes. Calm water.

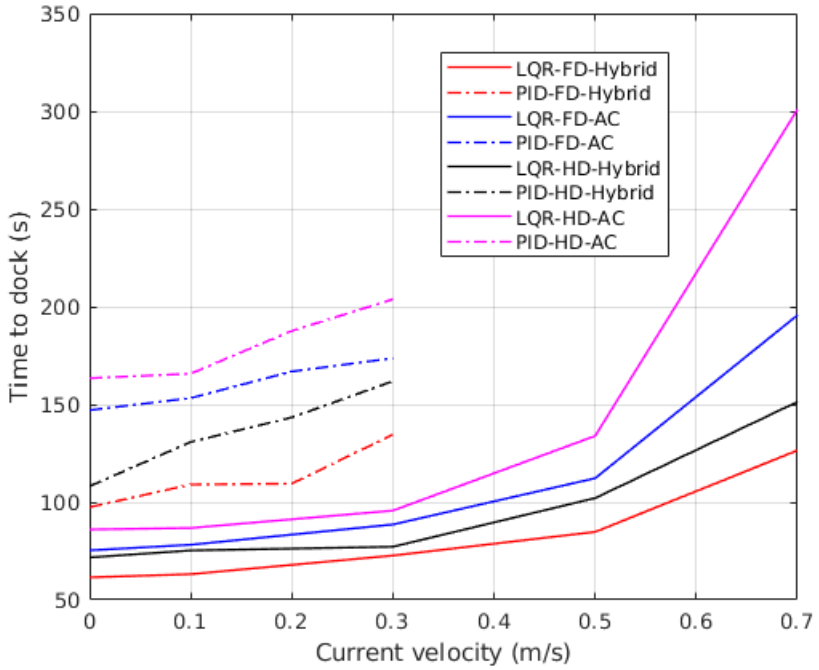


Figure 7.41 A comparison of the AUV time to dock using LQR and SMAC optimized PID controllers using the TDMA protocol with FD and HD communication modes by increasing water current velocity.

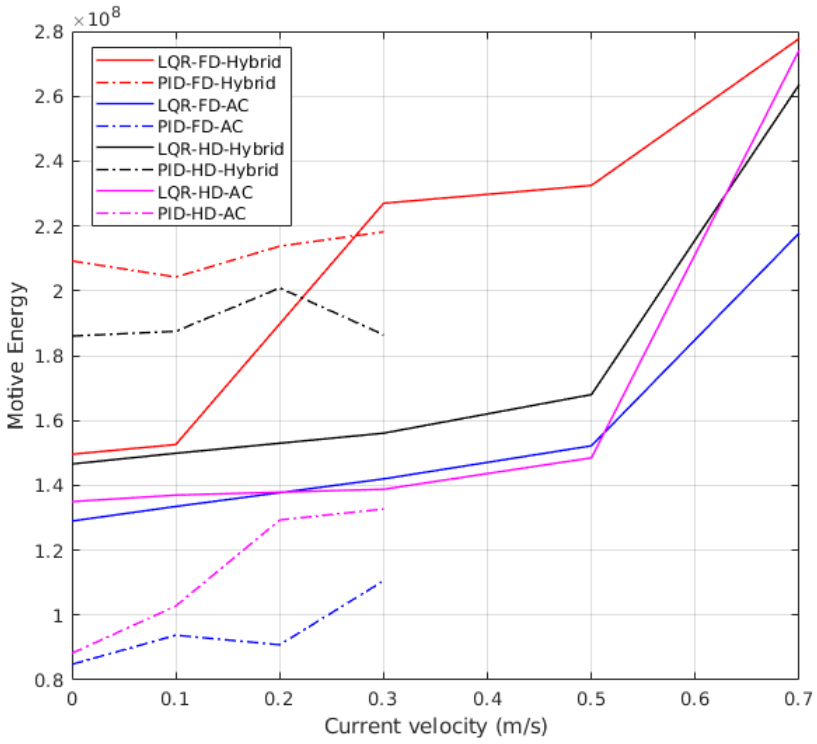


Figure 7.42 A comparison of the AUV motive energy consumption using LQR and SMAC optimized PID controllers using the TDMA protocol with FD and HD communication modes by increasing water current velocity.

second place.

It is also noted that the SMAC optimized PID methods are unable to complete docking maneuvers when the current speed is increased, while LQR methods can handle current strengths up to a velocity of $0.7m/s$.

On the other hand, the cost associated with the improvement of docking time is a higher motive energy among all vehicles, as illustrated in Figure 7.42. When comparing the SMAC optimized PID to the LQR, there is a slight difference in motive energy, with the acoustic based PID having a shorter motive energy than LQR based methods, while the hybrid PID has a higher motive energy.

Accordingly, this shows the importance of choosing the right controller for docking maneuvers in varying water current velocities and the associated trade-offs in performance and cost.

Chapter 8

CONCLUSIONS

In this thesis, we have developed a novel underwater networked control system (NCS) using hybrid acoustic and RF links to satisfy the requirements of docking maneuver application for AUVs. The acoustic link is utilized when the AUV is far away from the docking station. The large communication delay requires low sampling rates and low control gains in the NCS, resulting in high navigation error. However, at such distances precision is not required. When the AUV is closer to the docking station, the communication switches to the RF link, with low latency affording a faster sampling time for control. This allows higher control gains and more precise navigation accuracy. The proposed framework greatly improves the performance of the docking maneuver and can be used for any other underwater vehicle based NCS application that needs to operate autonomously with high precision.

To maximize spectrum efficiency, we have proposed to adopt full duplex communication for both underwater acoustic and RF links. Through simultaneous transmission and reception in FD communication, navigation data can be rapidly transmitted to AUVs using NCS and docking can be made more precise and faster. This is particularly critical in scenarios where water currents vary rapidly. By utilizing full duplex communication, we can ensure the system operates at maximum efficiency and provide the necessary navigational data to perform docking maneuvers.

The performance of the NCS may differ depending on the phase delay and the delays introduced by the communication modes and MAC protocols. Therefore, it is important to optimize the controller gains in order to compare different communication frameworks. In the absence of such an optimization, the comparison results of the systems may be skewed and unfair. Hence, via SMAC method, we optimized the PID gains of NCS for different communication frameworks by iteratively tuning the gains until specific performance criteria were satisfied. LQR controller implements

similar optimization for the NCS to ensure that the results are fair and unbiased and allows a more reliable evaluation of the performance of different communication frameworks under the same conditions. This leads to selecting the suitable control method and communication framework for use in underwater hybrid NCS.

For evaluating the performance of the underwater hybrid NCS for the docking maneuver application, we have implemented detailed and accurate physical models for the simulation of the system components including acoustic and RF underwater channel characteristics, MAC protocols, AUV hydrodynamics, real-time computers and water currents. We have also integrated a novel co-simulation environment modeling the networked control of AUVs using real-time embedded computers with realistic online communication. Our simulations also involve a realistic hydrodynamics model, essential to accurately model the underwater environment and the AUV trajectory. In addition, we have implemented all considered MAC protocols, including TDMA, Slotted ALOHA, and Waiting Room, as well as optimized PID and LQR controllers that can effectively respond to varying data rates.

In the integrated co-simulation environment, we have conducted various experiments to verify the disturbing effects of water currents on the performance of the AUVs. We have shown that employing hybrid acoustic and RF communication rather than acoustic-only is not only feasible for controlling underwater vehicles but also more beneficial. We also found that under calm water conditions, the hybrid system reduces the docking time by 33% due to the higher control gains that RF controller affords. Furthermore, subject to depth dependent disturbance, the acoustic-only system fails to approach the docking station, while our proposed hybrid system can withstand disturbance forces and still complete the docking maneuver. Although the hybrid system's communication energy at low water current speeds is smaller than that of the acoustic-only system by up to 34%, this amount grows with the increased speed of the water current. However, to improve the docking time, the hybrid system consumes up to 78% higher motive energy as compared to the acoustic-only system. The higher power consumption allows the hybrid system to succeed, whereas the acoustic-only system cannot complete the docking maneuver in the realistic scenario with water currents faster than 0.15 m/s. In comparison, our proposed hybrid system working in HD mode, can cope with increasing velocity up to 0.3 m/s and perform docking maneuver successfully. We have also observed that the full duplex communication protocols further improve the time to dock significantly and should be considered for actual implementations.

As an alternative to SMAC optimized PID, we have also implemented an LQR controller by mathematically modeling the hydrodynamics of an AUV. Despite the

convenience of PID's implementation and use, it is sensitive to changes in the gains. Nevertheless, LQR, can provide a better performance in complex implementations, particularly when it comes to controlling nonlinear systems and disturbances. Additionally, a significant advantage of LQR over PID control is that AUV requires separate PID controllers for each axis of movement control, whereas a single LQR can control 6 DOF movements.

The simulation results show that our proposed FD hybrid using LQR achieves the shortest docking time of approximately 62 seconds, while the corresponding SMAC optimized approach in FD mode takes around 97 seconds. Furthermore, using the FD mode on an acoustic only link with the LQR controller reduces the docking time by about 78 seconds, whereas for the PID-based method, the docking time is approximately 148 seconds, representing a 90% increase. Similarly, employing HD communication mode, the differences between LQR and SMAC optimized PID using hybrid and acoustic modes are approximately 45% and 90%, respectively. This indicates that the hybrid method using the LQR controller outperforms SMAC optimized PID with acoustic-based NCS in both HD and FD.

A significant finding in these experiments reveals acoustic based controllers have difficulty completing docking maneuver when the current velocity is increased. In fact, the proposed NCS with FD and HD hybrid modes can cope with growing currents up to a speed of $0.3m/s$ using optimized PID. In the case of speeds exceeding due to the sensitivity of the PID and the rapid change in the environment, the AUV could not approach the docking station. However, the FD and HD hybrid methods and the LQR controller can handle current strengths up to $0.7m/s$. The penalty to achieve this remarkable performance using FD hybrid is increased consumption of motive energy compared to the acoustic only system. This can be alleviated later by optimizing energy during the mission execution.

Finally, we have evaluated the system's performance under realistic water currents that fluctuated during the simulation, using two controllers, different MAC protocols, and FD and HD communication modes. Results indicate that if realistic currents are present, the docking time of acoustic-only approaches is increased by 50%, whereas for hybrid-based approaches, it increases by approximately 35%. These performance gains are the direct result of decreased communication delay in the RF region via the hybrid scheme and using FD communication mode, which allows high control gains.

This study has shown through detailed models and simulations that the hybrid networked control system for AUVs cooperating is possible. It is challenging to conduct underwater experiments due to significant budget and labor requirements. Because

of that, it has not been possible to test our proposed system in sea trials; however, we have presented an accurate environment that can simulate it as realistically as possible. The next phase in this study is to experimentally implement the proposed methods in underwater vehicles. Also, the docking maneuver that has been presented as an ocean floor scenario, will also be equally beneficial for docking an AUV to the hull of a supporting ship, for example.

BIBLIOGRAPHY

[1] X.-M. Zhang, Q.-L. Han, and X. Yu, "Survey on recent advances in networked control systems," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 5, pp. 1740–1752, 2016.

[2] J. Heidemann, M. Stojanovic, and M. Zorzi, "Underwater sensor networks: applications, advances and challenges," *Phil. Trans. R. Soc. A*, vol. 370, no. 1958, pp. 158–175, 2012.

[3] M. Chitre, J. Potter, and O. S. Heng, "Underwater acoustic channel characterisation for medium-range shallow water communications," in *OCEANS'04. MTTS/IEEE TECHNO-OCEAN'04*, vol. 1, pp. 40–45, IEEE, 2004.

[4] W. Farooq, T. Ali, A. Shaf, M. UMAR, and S. Yasin, "Atomic-shaped efficient delay and data gathering routing protocol for underwater wireless sensor networks," *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 27, no. 5, pp. 3454–3469, 2019.

[5] I. V. Zhilin, O. M. Bushnaq, G. De Masi, E. Natalizio, and I. F. Akyildiz, "A universal multimode (acoustic, magnetic induction, optical, rf) software defined radio architecture for underwater communication," in *The 15th International Conference on Underwater Networks & Systems*, pp. 1–6, 2021.

[6] I. F. Akyildiz, D. Pompili, and T. Melodia, "State-of-the-art in protocol research for underwater acoustic sensor networks," in *Proceedings of the 1st ACM international workshop on Underwater networks*, pp. 7–16, ACM, 2006.

[7] X. Che, I. Wells, G. Dickers, P. Kear, and X. Gong, "Re-evaluation of rf electromagnetic communication in underwater sensor networks," *IEEE Communications Magazine*, vol. 48, no. 12, pp. 143–151, 2010.

[8] U. Qureshi, F. Shaikh, Z. Aziz, S. M. Z. S. Shah, A. A. Sheikh, E. Felemban, and S. B. Qaisar, "RF path and absorption loss estimation for underwater

wireless sensor networks in different water environments," *Sensors*, vol. 16, no. 6, p. 890, 2016.

[9] S. Jiang and S. Georgakopoulos, "Electromagnetic wave propagation into fresh water," *Journal of Electromagnetic Analysis and Applications*, vol. 3, no. 07, p. 261, 2011.

[10] J. Li, M. Toulgoat, M. Déziel, F. R. Yu, and S. Perras, "Propagation modeling and MAC-layer performance in EM-based underwater sensor networks," in *Proceedings of the fourth ACM international symposium on Development and analysis of intelligent vehicular networks and applications*, pp. 111–117, ACM, 2014.

[11] M. C. Domingo, "Magnetic induction for underwater wireless communication networks," *IEEE transactions on antennas and propagation*, vol. 60, no. 6, pp. 2929–2939, 2012.

[12] B. Gulbahar and O. B. Akan, "A communication theoretical modeling and analysis of underwater magneto-inductive wireless channels," *IEEE Transactions on Wireless Communications*, vol. 11, no. 9, pp. 3326–3334, 2012.

[13] A. Morel, B. Gentili, H. Claustre, M. Babin, A. Bricaud, J. Ras, and F. Tieche, "Optical properties of the "clearest" natural waters," *Limnology and oceanography*, vol. 52, no. 1, pp. 217–229, 2007.

[14] Sonardyne, "Sonardyne bluecomm 200 uv underwater optical communications and data transfer modem." https://www.sonardyne.com/products/bluecomm-200-wireless-underwater-link/, October 2021.

[15] Hydromea, "Luma — high-speed, long range wireless optical modem." https://www.hydromea.com/underwater-wireless-communication, Ocotber 2021.

[16] Y. Chen, M. Kong, T. Ali, J. Wang, R. Sarwar, J. Han, C. Guo, B. Sun, N. Deng, and J. Xu, "26 m/5.5 gbps air-water optical wireless communication based on an ofdm-modulated 520-nm laser diode," *Optics express*, vol. 25, no. 13, pp. 14760–14765, 2017.

[17] I. V. Zhilin, O. M. Bushnaq, G. De Masi, E. Natalizio, and I. F. Akyildiz, "A universal multimode (acoustic, magnetic induction, optical, rf) software defined modem architecture for underwater communication," *IEEE Transactions on Wireless Communications*, 2023.

[18] F. Campagnaro, F. Guerra, F. Favaro, V. S. Calzado, P. Forero, M. Zorzi, and P. Casari, "Simulation of a multimodal wireless remote control system for

underwater vehicles," in *Proceedings of the 10th International Conference on Underwater Networks & Systems*, pp. 1–8, 2015.

[19] F. Campagnaro, A. Signori, and M. Zorzi, "Wireless remote control for underwater vehicles," *Journal of Marine Science and Engineering*, vol. 8, no. 10, p. 736, 2020.

[20] M. O'Rourke, E. Basha, and C. Detweiler, "Multi-modal communications in underwater sensor networks using depth adjustment," in *Proceedings of the Seventh ACM International Conference on Underwater Networks and Systems*, pp. 1–5, 2012.

[21] M. Soomro, S. N. Azar, O. Gurbuz, and A. Onat, "Work-in-progress: Networked control of autonomous underwater vehicles with acoustic and radio frequency hybrid communication," in *2017 IEEE Real-Time Systems Symposium (RTSS)*, pp. 366–368, IEEE, 2017.

[22] S. N. Azar, O. Erdemir, M. Soomro, Ö. G. ÜNLÜYURT, and A. Onat, "A hybrid acoustic-rf communication framework for networked control ofautonomous underwater vehicles: design and cosimulation," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 30, no. 4, pp. 1475–1491, 2022.

[23] A. J. Healey and D. Lienard, "Multivariable sliding mode control for autonomous diving and steering of unmanned underwater vehicles," *IEEE journal of Oceanic Engineering*, vol. 18, no. 3, pp. 327–339, 1993.

[24] K. Ogata and Y. Yang, *Modern control engineering*, vol. 4. Prentice hall India, 2002.

[25] M. Santhakumar and T. Asokan, "A self-tuning proportional-integral-derivative controller for an autonomous underwater vehicle, based on taguchi method," *Journal of Computer Science*, vol. 6, no. 8, p. 862, 2010.

[26] F. Hutter, H. H. Hoos, and K. Leyton-Brown, "Sequential model-based optimization for general algorithm configuration," in *International conference on learning and intelligent optimization*, pp. 507–523, Springer, 2011.

[27] G. F. Franklin, J. D. Powell, A. Emami-Naeini, and H. Sanjay, *Feedback control of dynamic systems*. Pearson London, 2015.

[28] M. M. Marcusso Manhães, S. A. Scherer, L. R. Douat, M. Voss, and T. Rauschenbach, "Framework for fair comparisons of underwater vehicle con-

trollers," in *Proceedings of the 7th International Conference on Simulation and Modeling Methodologies, Technologies and Applications*, pp. 102–113, 2017.

[29] T. I. Fossen, *Handbook of marine craft hydrodynamics and motion control*. John Wiley & Sons, 2011.

[30] C. S. Chin, *Computer-Aided Control Systems Design: Practical Applications Using MATLAB® and Simulink®*. CRC Press, 2012.

[31] F. Geridonmez, "Simulation of motion of an underwater vehicle," Master's thesis, Middle East Technical University, 2007.

[32] SNAME, "Nomenclature for treating the motion of a submerged body through a fluid," *The Society of Naval Architects and Marine Engineers, Technical and Research Bulletin No*, pp. 1–5, 1950.

[33] T. I. Fossen and O.-E. Fjellstad, "Nonlinear modelling of marine vehicles in 6 degrees of freedom," *Mathematical Modelling of Systems*, vol. 1, no. 1, pp. 17–27, 1995.

[34] M. Kirkeby, "Comparison of controllers for dynamic positioning and tracking of rov minerva," Master's thesis, NTNU: Norwegian University of Science and Technology, 2010.

[35] O. Yildiz, A. E. Yilmaz, and B. Gokalp, "State-of-the-art system solutions for unmanned underwater vehicles," *sensors*, vol. 1, no. 2, 2009.

[36] L. Stutters, H. Liu, C. Tiltman, and D. J. Brown, "Navigation technologies for autonomous underwater vehicles," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 4, pp. 581–589, 2008.

[37] J. J. Leonard and A. Bahr, "Autonomous underwater vehicle navigation," *Springer handbook of ocean engineering*, pp. 341–358, 2016.

[38] E. Bovio, D. Cecchi, and F. Baralli, "Autonomous underwater vehicles for scientific and naval operations," *Annual Reviews in Control*, vol. 30, no. 2, pp. 117–130, 2006.

[39] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, pp. 35–45, 1960.

[40] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman filter: Particle filters for tracking applications*. Artech house, 2003.

[41] M. G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (slam) problem," *IEEE Transactions on robotics and automation*, vol. 17, no. 3, pp. 229–241, 2001.

[42] C. M. Gussen, P. S. Diniz, M. L. Campos, W. A. Martins, F. M. Costa, and J. N. Gois, "A survey of underwater wireless communication technologies," *J. Commun. Inf. Sys*, vol. 31, no. 1, pp. 242–255, 2016.

[43] T. Melodia, H. Kulhandjian, L.-C. Kuo, and E. Demirors, "Advances in underwater acoustic networking," *Mobile ad hoc networking: Cutting edge directions*, pp. 804–852, 2013.

[44] G. Burrowes and J. Y. Khan, "Short-range underwater acoustic communication networks," in *Autonomous Underwater Vehicles*, pp. 100–105, InTech, 2011.

[45] G. Hattab, M. El-Tarhuni, M. Al-Ali, T. Joudeh, and N. Qaddoumi, "An underwater wireless sensor network with realistic radio frequency path loss model," *International Journal of Distributed Sensor Networks*, vol. 9, no. 3, p. 508708, 2013.

[46] A. Goldsmith, *Wireless communications*. Cambridge university press, 2005.

[47] T. Le-Ngoc and A. Masmoudi, "Full-duplex wireless communications systems," in *Wireless Networks*, Springer, 2017.

[48] D. Bharadia, E. McMilin, and S. Katti, "Full duplex radios," in *Proceedings of the ACM SIGCOMM 2013 conference on SIGCOMM*, pp. 375–386, 2013.

[49] A. Sahai, G. Patel, and A. Sabharwal, "Pushing the limits of full-duplex: Design and real-time implementation," *arXiv preprint arXiv:1107.0607*, 2011.

[50] H. Alves, T. Riihonen, and H. A. Suraweera, *Full-Duplex Communications for Future Wireless Networks*. Springer, 2020.

[51] L. Shen, B. Henson, Y. Zakharov, and P. Mitchell, "Digital self-interference cancellation for full-duplex underwater acoustic systems," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 67, no. 1, pp. 192–196, 2019.

[52] G. Qiao, S. Liu, Z. Sun, and F. Zhou, "Full-duplex, multi-user and parameter reconfigurable underwater acoustic communication modem. in 2013 oceans-san diego, 2013 (pp. 1–8)," *IEEE. https://doi. org/10.23919/oceans*, 2013.

[53] D. Bliss, P. Parker, and A. Margetts, "Simultaneous transmission and reception for improved wireless network performance," in *2007 IEEE/SP 14th Workshop on Statistical Signal Processing*, pp. 478–482, IEEE, 2007.

[54] S. Hong, J. Brand, J. I. Choi, M. Jain, J. Mehlman, S. Katti, and P. Levis, "Applications of self-interference cancellation in 5g and beyond," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 114–121, 2014.

[55] M. Erdem, O. Gurbuz, and H. Ozkan, "Integrated linear and nonlinear digital cancellation for full duplex communication," *IEEE Wireless Communications*, vol. 28, no. 1, pp. 20–27, 2021.

[56] T. S. Rappaport *et al.*, *Wireless communications: principles and practice*, vol. 2. prentice hall PTR New Jersey, 2002.

[57] J. Ahn, A. Syed, B. Krishnamachari, and J. Heidemann, "Design and analysis of a propagation delay tolerant aloha protocol for underwater networks," *Ad Hoc Networks*, vol. 9, no. 5, pp. 752–766, 2011.

[58] A. Munari, P. Mähönen, and M. Petrova, "A stochastic geometry approach to asynchronous aloha full-duplex networks," *IEEE/ACM Transactions on Networking*, vol. 25, no. 6, pp. 3695–3708, 2017.

[59] X. Wang, H. Huang, and T. Hwang, "On the capacity gain from full duplex communications in a large scale wireless network," *IEEE Transactions on Mobile Computing*, vol. 15, no. 9, pp. 2290–2303, 2015.

[60] X. Xie and X. Zhang, "Does full-duplex double the capacity of wireless networks?," in *IEEE INFOCOM 2014-IEEE Conference on Computer Communications*, pp. 253–261, IEEE, 2014.

[61] D. Marlali and O. Gurbuz, "Design and performance analysis of a full-duplex mac protocol for wireless local area networks," *Ad Hoc Networks*, vol. 67, pp. 53–67, 2017.

[62] K. Akcapinar and O. Gurbuz, "Full-duplex bidirectional communication under self-interference," in *2015 13th International Conference on Telecommunications (ConTEL)*, pp. 1–7, IEEE, 2015.

[63] F. Birlik, O. Gurbuz, and O. Ercetin, "Iptvhome networking via 802.11 wireless mesh networks: an implementation experience," *IEEE Transactions on Consumer Electronics*, vol. 55, no. 3, pp. 1192–1199, 2009.

[64] X. Xie and X. Zhang, "Semi-synchronous channel access for full-duplex wireless networks," in *2014 IEEE 22nd International Conference on Network Protocols*, pp. 209–214, IEEE, 2014.

[65] K. J. Åström and B. Wittenmark, *Computer-controlled systems: theory and design*. Courier Corporation, 2013.

[66] T. Salgado-Jimenez, J.-M. Spiewak, P. Fraisse, and B. Jouvencel, "A robust control algorithm for auv: based on a high order sliding mode," in *Oceans' 04 MTS/IEEE Techno-Ocean'04 (IEEE Cat. No. 04CH37600)*, vol. 1, pp. 276–281, IEEE, 2004.

[67] P. Herman, "Decoupled pd set-point controller for underwater vehicles," *Ocean Engineering*, vol. 36, no. 6-7, pp. 529–534, 2009.

[68] C. Field A, Cherches D, "Optimal control of an autonomous underwater vehicle," in *Proceedings of the World Automatic Congress. Hawaii (USA)*, 2000.

[69] R. Cristi, F. A. Papoulias, and A. J. Healey, "Adaptive sliding mode control of autonomous underwater vehicles in the dive plane," *IEEE journal of Oceanic Engineering*, vol. 15, no. 3, pp. 152–160, 1990.

[70] L. Rodrigues, P. Tavares, and M. de Sousa Prado, "Sliding mode control of an auv in the diving and steering planes," in *OCEANS 96 MTS/IEEE Conference Proceedings. The Coastal Ocean-Prospects for the 21st Century*, vol. 2, pp. 576–583, IEEE, 1996.

[71] S. Soylu, B. J. Buckham, and R. P. Podhorodeski, "A chattering-free sliding-mode controller for underwater vehicles with fault-tolerant infinity-norm thrust allocation," *Ocean Engineering*, vol. 35, no. 16, pp. 1647–1659, 2008.

[72] V. Utkin, "Discussion aspects of high-order sliding mode control," *IEEE Transactions on Automatic Control*, vol. 61, no. 3, pp. 829–833, 2015.

[73] M. Caccia and G. Veruggio, "Guidance and control of a reconfigurable unmanned underwater vehicle," *Control engineering practice*, vol. 8, no. 1, pp. 21–37, 2000.

[74] J. Li and P. Lee, "Design of an adaptive nonlinear controller for depth control of an autonomous underwater vehicle," *Ocean engineering*, vol. 32, no. 17-18, pp. 2165–2181, 2005.

[75] A. Pisano and E. Usai, "Output-feedback control of an underwater vehicle prototype by higher-order sliding modes," *Automatica*, vol. 40, no. 9, pp. 1525–1531, 2004.

[76] J. Lorentz and J. Yuh, "A survey and experimental study of neural network auv control," in *Proceedings of Symposium on Autonomous Underwater Vehicle Technology*, pp. 109–116, IEEE, 1996.

[77] V. Porto and D. Fogel, "Neural networks for auv guidance control: initial efforts encouraging in design testing of a tivo-dimensional controller for accurate navigation," *Sea Technology*, 1992.

[78] R. M. Sanner and D. L. Akin, "Neuromorphic pitch attitude regulation of an underwater telerobot," *IEEE Control Systems Magazine*, vol. 10, no. 3, pp. 62–68, 1990.

[79] I. S. Akkizidis, G. Roberts, P. Ridao, and J. Batlle, "Designing a fuzzy-like pd controller for an underwater robot," *Control Engineering Practice*, vol. 11, no. 4, pp. 471–480, 2003.

[80] D. T. Bill Messner, "Control tutorials for matlab and simulink." https://ctms.engin.umich.edu/CTMS/index.php?aux=Home, September 2020.

[81] M. Claesen and B. De Moor, "Hyperparameter search in machine learning," *arXiv preprint arXiv:1502.02127*, 2015.

[82] C.-W. Hsu, C.-C. Chang, C.-J. Lin, *et al.*, "A practical guide to support vector classification," 2003.

[83] J. Bergstra and Y. Bengio, "Random search for hyper-parameter optimization.," *Journal of machine learning research*, vol. 13, no. 2, 2012.

[84] J. Y. Hesterman, L. Caucci, M. A. Kupinski, H. H. Barrett, and L. R. Furenlid, "Maximum-likelihood estimation with a contracting-grid search algorithm," *IEEE transactions on nuclear science*, vol. 57, no. 3, pp. 1077–1084, 2010.

[85] X. He, K. Zhao, and X. Chu, "Automl: A survey of the state-of-the-art," *Knowledge-Based Systems*, vol. 212, p. 106622, 2021.

[86] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas, "Taking the human out of the loop: A review of bayesian optimization," *Proceedings of the IEEE*, vol. 104, no. 1, pp. 148–175, 2015.

[87] C. E. Rasmussen, "Gaussian processes in machine learning," in *Summer school on machine learning*, pp. 63–71, Springer, 2003.

[88] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," *Advances in neural information processing systems*, vol. 24, 2011.

[89] Y. Bengio, "Gradient-based optimization of hyperparameters," *Neural computation*, vol. 12, no. 8, pp. 1889–1900, 2000.

[90] D. Maclaurin, D. Duvenaud, and R. Adams, "Gradient-based hyperparameter optimization through reversible learning," in *International conference on machine learning*, pp. 2113–2122, PMLR, 2015.

[91] B. Adenso-Diaz and M. Laguna, "Fine-tuning of algorithms using fractional experimental designs and local search," *Operations research*, vol. 54, no. 1, pp. 99–114, 2006.

[92] M. Birattari, Z. Yuan, P. Balaprakash, and T. Stützle, "F-race and iterated f-race: An overview," *Experimental methods for the analysis of optimization algorithms*, pp. 311–336, 2010.

[93] F. Hutter, H. H. Hoos, K. Leyton-Brown, and T. Stützle, "Paramils: an automatic algorithm configuration framework," *Journal of Artificial Intelligence Research*, vol. 36, pp. 267–306, 2009.

[94] "SMAC: Sequential Model-based Algorithm Configuration." https://automl.github.io/SMAC3/master/, June 2021.

[95] D. Cook, A. Vardy, and R. Lewis, "A survey of auv and robot simulators for multi-vehicle operations," in *2014 IEEE/OES Autonomous Underwater Vehicles (AUV)*, pp. 1–8, IEEE, 2014.

[96] M. M. M. Manhaes, S. A. Scherer, L. R. Douat, M. Voss, and T. Rauschenbach, "Use of simulation-based performance metrics on the evaluation of dynamic positioning controllers," in *OCEANS 2017-Aberdeen*, pp. 1–8, IEEE, 2017.

[97] "Gazebo, Robot simulation Environment. Available online." http://www.gazebosim.org, August 2020.

[98] M. M. M. Manhães, S. A. Scherer, M. Voss, L. R. Douat, and T. Rauschenbach, "UUV simulator: A gazebo-based package for underwater intervention and multi-robot simulation," in *OCEANS 2016 MTS/IEEE*, IEEE, 2016.

[99] V. Berg, "Development and commissioning of a dp system for rov sf 30k," Master's thesis, NTNU: Norwegian University of Science and Technology, 2012.

[100] M. M. M. Manhães, S. A. Scherer, M. Voss, L. R. Douat, and T. Rauschenbach, "Uuv simulator: A gazebo-based package for underwater intervention and multi-robot simulation," in *OCEANS 2016 MTS/IEEE Monterey*, pp. 1–8, IEEE, 2016.

[101] E. Ege, *Feedback motion planning of unmanned underwater vehicles via random sequential composition*. PhD thesis, Middle East Technical University, Middle East Technical University Library, 9 2019.

[102] "Open Source 3D Graphics Engine." http://www.ogre3d.org, September 2021.

[103] "Real-Time Physics Simulator." http://bulletphysics.org, September 2021.

[104] "Open Dynamics Engine." http://www.ode.org/, September 2021.

[105] D. Henriksson, A. Cervin, and K.-E. Årzén, "Truetime: Real-time control system simulation with matlab/simulink," in *Proceedings of the Nordic MATLAB Conference*, Copenhagen, Denmark, 2003.

[106] J. W. Liu, *Real-time systems*. Prentice Hall PTR, 2000.

[107] M. Prats, J. Perez, J. J. Fernandez, and P. J. Sanz, "An open source tool for simulation and supervision of underwater intervention missions," in *2012 IEEE/RSJ international conference on Intelligent Robots and Systems*, pp. 2577–2582, IEEE, 2012.

[108] "Components of sdf model." http://gazebosim.org/tutorials, September 2020.

[109] "Robot Operating System. Available online." http://www.ros.org, April 2020.

[110] M. Morgado, P. Oliveira, and C. Silvestre, "Design and experimental evaluation of an integrated USBL/INS system for auvs," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pp. 4264–4269, IEEE, 2010.

[111] A. Cervin, D. Henriksson, B. Lincoln, J. Eker, and K.-E. Arzen, "How does control timing affect performance? analysis and simulation of timing using jitterbug and truetime," *IEEE control systems*, vol. 23, no. 3, pp. 16–30, 2003.

[112] E. Brochu, V. M. Cora, and N. De Freitas, "A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," *arXiv preprint arXiv:1012.2599*, 2010.