ADAPTIVE MODULATION AND CODING WITH FEEDBACK SCHEDULING FOR UNDERWATER ACOUSTIC COMMUNICATION

WU SHUANGSHUANG
(B.E.), Wuhan University (WHU), China

A THESIS SUBMITTED FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
NATIONAL UNIVERSITY OF SINGAPORE

2023

Thesis Advisors:
Associate Professor Mandar Chitre, Main Thesis Advisor
Dr V Prasad Anjangi, Co-Advisor

Examiners:
Professor Biplab Sikdar
Associate Professor Wee-Seng Soh

DECLARATION

I hereby declare that this thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

Wu Shuangshuang

21th September 2023

ACKNOWLEDGEMENTS

Looking back at the Ph.D. journey, which began with the onset of COVID and is culminating as COVID comes to an end this year, I am about to submit my thesis. This milestone would not have been possible without the support of many great people. Upon accomplishing the study, I would like to take this opportunity to convey my appreciation to them in this acknowledgment.

First of all, I would like to express my deepest gratitude to my supervisor, A/Prof. Mandar Chitre who introduced me to the field of underwater acoustics. More than guiding the technical aspects of my research, he imparted invaluable life lessons that fortified me to face challenges resolutely. His enduring support, from his patience during my initial days in Singapore to his encouragement through research obstacles, has been indispensable. My contributions to this domain would have been unattainable without his unwavering belief in me.

I also want to say great thanks to my co-supervisor, Dr. Prasad Angangi, for supporting the research work of my dissertation. Moreover, he is pivotal in training me to conduct simulations and experiments using modems, which laid the foundation for my research. All my papers can not be published without his invaluable assistance and guidance in writing and research.

Special acknowledgment goes to Dr. Kexin Li, who eased my transition to life in Singapore. My gratitude extends to Ms. Luyuan Peng and Dr. Kexin Li for for cheering me up especially during challenging phases of this thesis. Engaging discussions with Dr. Hari Vishnu, Dr. Gabriel Chua, Dr. Rajat Mishra, and Dr. Too Yuen Min, enriched my perspectives on the limitations and

potential of my study. I am thankful to Mr. Kee Boon Leng for his assistance in prepping my experiments, making the research process smoother. To everyone in ARL, your feedback and encouragement during every mock presentation were invaluable. I also want to express gratitude to colleagues from Subnero who generously supplied me with the modems and other necessary equipment for my experiments.

Lastly, heartfelt thanks go to my parents for their unwavering love and support. Their steadfast encouragement and belief in me powered my academic pursuits.

Contents

Abstract

Underwater acoustic channels exhibit significant temporal and spatial variability, making it challenging to design a single communication scheme that works well everywhere and at all times. Adaptive Modulation and Coding (AMC) techniques offer a solution by dynamically selecting the optimal scheme for specific channel conditions. Data-driven models are commonly used in AMC for their simplicity. A key dilemma in AMC is that of exploration versus exploitation. Exploration means trying new Modulation and Coding Schemes (MCSs) for potentially better communication performance under the prevailing channel conditions while exploitation involves utilizing the best MCS known so far, based on past experiences and collected data. Popular policies such as random, greedy, ϵ-greedy, Upper Confidence Bound (UCB) are employed in AMC. We propose a new algorithm framework based on Monte Carlo Tree Search (MCTS), K-MCTS, which builds a K-level look-ahead tree for every simulation in MCTS. Simulation results demonstrate the superiority of the K-MCTS for balancing the exploration and exploitation, but the inherent dependency of data-driven methods on substantial training data makes this alone unsuitable for underwater communication applications.

Recognizing the limitations of data-driven methods, particularly the extensive data requirements, we incorporate channel physics knowledge into the AMC algorithm design. We propose a Bit Error Rate (BER) estimation

model that fuses channel physics knowledge in Orthogonal Frequency Division Multiplexing (OFDM) system. In complex sea conditions, we enhance the reliability of AMC by extending our BER prediction model from a point prediction to an interval predictor. This extension involves incorporating Gaussian Process Regression (GPR) to address the uncertainty in BER estimation. Predictions from such an algorithm are used to drive AMC to maximize communication throughput reliably.

For effective AMC, consistent receiver-to-transmitter feedback is vital for Channel State Information (CSI) collection. However, feedback sent too often can diminish throughput in channels with huge propagation delays, while inadequate feedback can compromise AMC decisions. Addressing this, we propose an algorithm incorporating Tree Search with Deep Q-Network (DQN), namely TS-DQN, to strike an optimal feedback balance, subsequently optimizing communication performance.

We demonstrate the advantages of our algorithm through experiments in a test tank and at sea. Simulations further corroborate its robustness across varied underwater settings. Our TS-DQN framework also offers a generalized solution for any MDP prioritizing long-term rewards, particularly in scenarios with exploration and exploitation challenges and expansive action or state spaces.

List of Tables

List of Figures

List of Acronyms

AMC	Adaptive Modulation and Coding
BER	Bit Error Rate
CSI	Channel State Information
DL	Deep Learning
DNN	Deep Neural Network
DQN	Deep Q-Network
ESNR	Effective Signal-to-Noise Ratio
FEC	Forward Error Correction
FHBFSK	Frequency-Hopping Binary Frequency Shift Keying
FRI	Feedback Report Interval
FSK	Frequency Shift Keying
GPR	Gaussian Process Regression
KPIs	Key Performance Indicators
LDPC	Low-Density Parity Check
MAE	Mean Absolute Error
MCS	Modulation and Coding Scheme
MCTS	Monte Carlo Tree Search
MDP	Markov Decision Process
ML	Machine Learning
OFDM	Orthogonal Frequency Division Multiplexing
PER	Packet Error Rate
PSK	Phase-Shift Keying
QAD	Quantile Absolute Deviation
QAM	Quadrature Amplitude Modulation
SNR	Signal-to-Noise Ratio
RL	Reinforcement Learning
RX	Receiver node

TX	Transmitter node
UAC	Underwater Acoustic Communication
UCB	Upper Confidence Bound

List of Notation

$\boldsymbol{a}$	Modulation scheme
$\boldsymbol{a}^i$	The i^{th} scheme in modulation scheme space
$\mathcal{A}$	Modulation scheme space
$d(\boldsymbol{a})$	Uncoded data rate of $\boldsymbol{a}$
$\boldsymbol{\epsilon_j}(\boldsymbol{a})$	Measured BER during the j^{th} FRI
$\hat{\epsilon}_j(\boldsymbol{a})$	Estimated BER at state $\boldsymbol{s_j}$
$\eta_j(\cdot)$	Regression analysis model for QAD prediction
D_j	Expected reward at state S_j
h_j	Number of frames in the j^{th} FRI
$\mathcal{H}$	Set of possible FRI values
k	Index used for packet step in a planning iteration
k_1	Number of frames for which "test" mode is enabled
k_2	Number of frames for which "test" mode is disabled
$\bar{k}$	Ratio: $\frac{k_1}{k_1+k_2}$
l	Distance between TX node and RX node
m_j^i	number of times scheme $\boldsymbol{a}^i$ has been tried
$\tilde{m}_j^i$	number of times scheme $\boldsymbol{a}^i$ has been successful
n_j'	Percentage of transmitted bits
N	Total number of bits to be transmitted.
N_j'	Amount of transmitted information at state $\boldsymbol{s_j}$
ω_j	Weight parameters of FRI determination model
r_j	Throughput of the j^{th} FRI
$\rho(\hat{\epsilon}_j(\boldsymbol{a}))$	FEC rate selected based on estimated BER $\hat{\epsilon}_j(\boldsymbol{a})$
$\boldsymbol{\varrho}$	Set of available FEC rates
$\boldsymbol{s_j}$	State where the CSI of the j^{th} FRI is updated
$\mathcal{S}$	State space
τ_j	Transmission duration of each frame in the j^{th} FRI
τ_{m}	Duration of frames containing modulation information
τ_{fd}	Duration of frames containing feedback information
τ_{pd}	Propagation delay between the TX and RX nodes

θ_j	Weight parameters for median prediction from BER distribution
$\zeta(\cdot)$	Model for median prediction from BER distribution

Chapter 1

Introduction

1.1 Background

The oceans, covering 70% of Earth's surface, remain largely unexplored. As interest in underwater research, exploration, and commercial activities grows, there is an increasing demand for more advanced technologies. While electromagnetic waves work well for wireless communication in the air, they face challenges such as limited propagation range and increased signal loss in underwater environments. Acoustic waves, however, can propagate long distances with comparatively low attenuation, making them ideal candidates for maintaining underwater communication links. As a result, Underwater Acoustic communication (UAC) has gained prominence in research, essential for fields like oceanography, marine biology, offshore exploration, and defense. These domains require high-speed data transfer and real-time communication for operational success. However, UAC faces challenges due to limited bandwidth, high signal attenuation, and ambient noise, combined with the dynamic nature of underwater environments influenced by factors like temperature gradients, salinity, and water currents [1]. These factors make real-time, reliable communication and high data-rate transmission in UAC especially complex and necessitate sophisticated communication strategies to ensure robust UAC

systems.

Adaptive Modulation and Coding (AMC) techniques present a viable solution to navigate the dynamic characteristics of UAC channels, promoting efficient data transmission. Channel behaviors vary based on diverse factors, including location, depth, temperature/salinity profiles, modulation techniques, operational frequency, tidal influences, and a myriad of other factors. Therefore, a modulation scheme optimally designed for specific channel conditions can not maintain robust performance in the long-term deployment of UAC systems [2]. When a modulation scheme is defined for ensuring successful frame transmission under adverse channel states, such as those influenced by strong underwater currents leading to water turbulence, communication reliability can be guaranteed. However, this conservative approach compromises spectrum efficiency when channel conditions subsequently improve. Instead, AMC permits the dynamic selection of appropriate Modulation and Coding Schemes (MCSs) based on real-time assessment of channel conditions [3]–[7]. By continuously adapting to changing channel characteristics, AMC strives to attain an optimal trade-off between communication reliability and throughput, thus enhancing UAC system performance. In the wireless communication system, efforts have been directed to optimize the following parameters:

- Modulation schemes (e.g., Phase-Shift Keying (PSK), Frequency Shift Keying (FSK), and Quadrature Amplitude Modulation (QAM)).
- Error correction methodologies and associated coding rates.
- Power levels for transmission.

- Allocated channel bandwidth.
- Frame length.
- Parameters in Orthogonal Frequency Division Multiplexing (OFDM) systems, encompassing cyclic prefix length, number of subcarriers, number of nulls, etc.

Collectively, these parameters form an extensive array of potential modulation configurations. Nonetheless, the absence of a universally accepted UAC channel model presents an obstacle in accurately evaluating communication system performance. Furthermore, there appears to be a research gap concerning modems that automate AMC. In most current works, modulation configurations are typically predetermined and retained for transmission, or tuned manually.

The effectiveness of AMC heavily relies on the precise and prompt acquisition of Channel State Information (CSI) [8]–[10]. Feedback-based mechanisms are commonly employed in AMC to obtain real-time channel data. This feedback facilitates the transmission of channel metrics to the source, enabling informed decisions on MCSs. The speed of sound in water is approximately 1500 m/s, resulting in propagation delays that are 200000× higher than those experienced in terrestrial radio communication networks [11]. These propagation delays are comparable to typical frame duration in UAC. Extensive research has addressed the ill effects of large propagation delays, impacting handshaking protocols and retransmission schemes [12], as well as medium-access control layer protocols preventing data collisions [13]. In a one-to-one communication system, where data frames are exchanged between a transmitter node and a receiver node,

the time taken for data frames to be received is influenced not only by the frame transmission time but also by the distance between the transmitter and receiver [14]. The transmitter awaits CSI feedback before performing AMC and initiating frame transmission. In such scenarios, the introduction of two-way propagation delays can substantially degrade the channel throughput. However, to our knowledge, there is a limited amount of research focused on mitigating the ill effects stemming from propagation delays in one-to-one communication systems.

1.2 Motivation

Consider an UAC system where information frames are exchanged between a transmitter (TX) node and a receiver (RX) node. Our goal is to transmit a large file containing N bits from the TX node to the remote RX node, situated at a distance l, in the shortest possible time, thereby optimizing channel throughput. Modulation and Coding techniques encode these bits onto frames to ensure reliable communication. As detailed in Section 1.1, modulation in UAC systems can be characterized by either individual or combined tunable parameters. For instance, modulation schemes may comprise phase, frequency, or amplitude modulation, such as PSK, FSK, or QAM. In the context of OFDM systems, modulation can be influenced by parameters like the number of subcarriers, cyclic prefix length, the designated channel bandwidth, or diverse modulation orders assigned to subcarriers. After the modulation process, error correction mechanisms, like Forward Error Correction (FEC), embed redundant bits into the modulated frames. This redundancy facilitates the RX node in detecting

and correcting potential transmission errors.

Given the dynamic nature of UAC channels, it is impractical to design a one-size-fits-all modulation scheme that performs optimally across all scenarios. We employ the AMC techniques to enable tuning the MCSs based in line with prevailing channel conditions. Research on AMC in UAC channels predominantly investigates the relationship between channel characteristics, like the Signal-to-Nosie Ratio (SNR) information, and system design. Typically, there is a limited amount of MCSs available for AMC and data-driven approaches are applied to figure out the relationship between the channel characteristic and each MCS. Data-driven methods have gained prominence due to their simple input requirements, such as some basic metrics SNR or Bit Error Rate (BER), capability for various problems without or with limited knowledge about underlying physics, and ability to learn and extract insights from provided datasets. This perspective spurred our development of an innovative data-driven AMC algorithm, as discussed in Chapter 3.

Data-driven methods generally rely on statistical and Machine Learning (ML) analyses, necessitating heavily on data availability. Their inherent demand for extensive training data intensifies when transitioning to real-sea applications. For instance, [15] documents that, to calibrate just four MCSs (JANUS, BPSK, QPSK, OFDM) in a long-range UAC system, approximately 100000$\times$ data points were collected for ML-driven channel classification, supplemented by 900 simulated channels due to a limited number of channels from sea trials. As for operating AMC in an expansive MCS landscape, like tuning the number of subcarriers and cyclic prefix length parameters in OFDM systems, the number

of available MCSs can be 1000× more than that in [15] and hence the dataset required will be a prohibitive size. The time required to collect such data for purely data-driven AMC strategies challenges the goal of transmitting N bits in the shortest possible time. The incorporation of physics-informed methods offers an advantage by leveraging prior channel knowledge, often reducing the reliance on extensive training data typically demanded by purely data-driven approaches. As shown in [16], the inherent physics of the OFDM system can act as an initial filter, narrowing down the feasible MCSs domain. Simultaneously, channel physics helps build correlations among MCSs with similar values, fostering expedited convergence rates for the AMC model when synergized with data-driven techniques. Motivated by these findings, our focus shifts to delving into the underlying channel physics of UAC channels and presents it as an appealing substitute to exclusively data-driven approaches, aiming for a more effective AMC.

Research involving AMC in wireless communication has predominantly revolved around single-carrier systems [17] as well as multi-carrier systems like OFDM. Nowadays, OFDM has emerged as a preferred alternative to single-carrier transmission, particularly for the forthcoming generation of commercially available UAC modems. This preference for OFDM stems from its straightforwardness and robustness in handling unique UAC characteristics, such as multipath and frequency-selective fading, without the need for intricate equalization procedures [18], [19]. Therefore, we first focus on physics-informed AMC in OFDM systems. The properties of OFDM have been detailed in works such as [16], [20]. Specifically, they underline that the duration of the cyclic prefix

should surpass the channel's delay spread. Furthermore, it is imperative that the channel remains relatively consistent throughout the symbol's duration, implying that this duration should not exceed the channel coherence time. Moreover, to ensure flat fading on each subcarrier, the bandwidth allocated to each subcarrier should not surpass the channel's coherence bandwidth. Armed with these theoretical underpinnings, we propose a physics-informed AMC algorithm in the OFDM system.

Frames are transmitted from the TX node to the RX node after modulation and coding. The CSI is subsequently obtained at the RX node through feedback, as performing AMC heavily relies on obtaining accurate CSI for communication performance evaluation. Given the considerable propagation delays in certain UAC channels, it is impractical to expect feedback after every frame transmission and still achieve high throughput. There is also an inherent trade-off between CSI feedback periodicity and the accuracy of channel estimation for AMC. Increasing the periodicity of CSI feedback reduces the overhead due to the long propagation delays. However, obtaining feedback more often enables more frequent updates of the channel information, providing better tracking of channel variations. This allows the system to gain faster convergence speed of AMC strategy and optimize performance. On the other hand, if the channel variations are relatively slow or the AMC model has been well-trained, increasing the feedback periodicity may be sufficient to capture the relevant changes, thereby conserving resources. The research on dynamically scheduling feedback is still relatively limited which motivates us to propose a feedback scheduling strategy and determine relevant decision parameters to address the trade-off between

communication performance and resource utilization.

Feedback typically comprises metrics like BER, SNR, and throughput to guide the transmitter in assessing communication performance and determining the optimal MCS. Throughput assessment is vital in optimizing data transmission rates among these metrics, especially when accounting for two-way propagation delays. The computation of throughput entails the measurement of successful bit transmission over a specified duration. For a given transmission range with fixed propagation delay, the AMC strategy can opt for MCSs with higher coded data rates for better throughput. The coded data rate comprises uncoded data rate and error correction overhead. BER knowledge of MCSs aids in the selection of appropriate error correction techniques and coding rates. Therefore, accurate BER estimation is indeed crucial for improving AMC for optimal throughput in communication systems. However, the time-varying behavior of UAC channels introduces significant fluctuations in the actual BER. Consequently, we emphasize the need for BER distribution prediction which provides a range of possible BER values given any modulation configuration. Such AMC algorithms select MCSs leveraging this predicted BER distribution, synchronizing data rates and reliability with prevailing channel conditions.

1.3 Thesis Contributions

The objective of this thesis is to develop and implement an AMC algorithm together with the feedback scheduling mechanism, targeted at maximizing channel throughput for the transmission of large files of fixed bit size in the shortest possible time. The key contributions can be outlined as follows:

1. We utilize Markov Decision Processes (MDPs) for AMC and feedback scheduling formulation. Drawing inspiration from prevalent MDP algorithms, we introduce the K-MCTS algorithm in Chapter 3, adeptly balancing exploration and exploitation, particularly when channel information is limited.

2. In light of the extensive data requirements of data-driven algorithms presented in Chapter 3, we introduce an AMC strategy that incorporates channel physics in the OFDM system in Chapter 4. Within the AMC strategy, we formulate a heuristic BER estimation model that aligns well with empirical BER findings, underscoring the augmented efficacy of AMC as evidenced by simulations.

3. We then assess the practical challenges of using our AMC algorithm in real-sea scenarios in Chapter 6. Emphasizing the criticality of robustness and reliability in MCS selection in real-sea experiments, we introduce a BER distribution predictor, which harnesses the power of GPR. This methodology quantifies BER uncertainties, ensuring a reliable AMC performance evaluation and throughput optimization as evidenced by real-sea experiments.

4. With knowing the importance of AMC efficiency and two-way propagation delays in throughput optimization within one-to-one communication systems, heuristic feedback scheduling strategies are previously explored in Chapters 3 and 4. We further introduce an algorithm, TS-DQN, which merges tree search and Deep Q-Network (DQN) in Chapter 5 to schedule

the MCSs tuning and feedback timing. TS-DQN is specifically tailored for long-term throughput optimization in AMC, addressing the challenges posed by high-dimensional action and state spaces in MDPs.

5. We set up a 2-node UAC network in a test tank and at sea. Using our TS-DQN-based AMC algorithm, we observed significant throughput improvements across various experiments. Additionally, we test the algorithm in diverse simulated UAC channels with varying propagation structures and BER profiles in Chapter 6.

6. Previous works addressed the exploration-exploitation trade-offs for MCS selection and feedback scheduling separately within our MDP framework. In Chapter 7, we integrate these trade-offs under the TS-DQN framework and test the enhanced approach in diverse simulated UAC channels with distinct propagation structures and BER profiles.

Chapter 2

Literature Review

Research on AMC in wireless communication systems has highlighted its capability to enhance communication efficiency and reliability. This review summarizes some recent studies on the success of AMC in wireless communication. Numerous studies perform AMC in a data-driven style that depends on the availability and quality of data to figure out the modulation and setup selection based on the channel conditions. However, limited literature has been done on incorporating channel physics to enable the MCS selection in AMC. AMC also enables the feedback loop between the transmitter and receiver in the communication system for updating the AMC strategy iteratively. In UAC systems, papers in the literature pay less attention to the latency introduced by obtaining feedback given the huge propagation delays in the one-to-one system. Since in subsequent chapters of this thesis, we formulate the AMC with feedback scheduling in UAC as a MDP, we will also explore state-of-the-art algorithms employed in MDPs to propose an appropriate solution for the potential challenges in our problem.

2.1 AMC Methodologies

The development of AMC techniques for wireless communication has witnessed significant progress. Existing literature reviews highlight two primary

approaches for implementing AMC: data-driven methods and physics-informed methods.

2.1.1 Data-driven Methods in AMC

Data-driven methods rely on ML techniques and statistical analysis to make adaptive decisions for MCSs based on real-time channel measurements and feedback. These methods utilize large datasets of empirical channel data to train models that can predict channel conditions and optimize the choice of MCSs accordingly. In Data-driven AMC approaches, various branches of ML such as supervised learning, unsupervised learning, and Reinforcement Learning (RL) [21] or other statistical techniques have been applied to achieve AMC.

In the terrestrial wireless communication area, data-driven approaches have shown promising results in AMC. Prior research has struggled to find channel quality parameters and constructed look-up tables to simultaneously provide the mapping to channel performance metrics in a classification fashion. For example, in [22], a supervised learning method, Artificial Neural Network, aided SNR estimation of different MCSs and operated AMC accordingly. In [23], supervised learning helped AMC to exploit past observations of error rate and the associated channel state information to predict the ordered SNR and choose the best MCSs in a Multiple-Input Multiple-Output system. A fast link adaptation algorithm based on a support vector machine aiming to minimize computational time was proposed in [24]. However, AMC algorithms in both [22]–[24] require sufficient training data. Although in [25], the same support vector machine-based

algorithm was employed and it assisted AMC without any external training while this method is only applicable in cognitive radio networks and not generalized. Unsupervised learning attempts to divide inputs into clusters having common factors and to extract frequent patterns. The author of [26] presented a clustering algorithm, the k-means algorithm, to do AMC via modem grouping in different channel conditions in a wired OFDM system. The k-means algorithm was also employed in [27] to split the mobile stations into clusters where the mobile stations were selected to maximize the capacity. Unlike supervised and unsupervised learning, RL trains the AMC strategies in an online style via interaction with the environment and lowers the dependence on the training data [28]. [29] first proposed RL as a possible and practical strategy for solving AMC problems. The work presented in [30] utilized RL in AMC which is able to optimize the channel performance in terms of BER, transmission time, and the energy consumption of the transmitter. Similar works have been done in 5G network [31] and an OFDM system [32]. When dealing with higher dimensional problems, a Neural Network-based extension of an RL scheme, i.e., DQN, is well established for balancing between the convergence time of the algorithm and the dimensions of the search space [33]. [33] employed DQN for link adapting based on mapping the different SNR rate regions to optimal modulation schemes.

The unique propagation characteristics posed by UAC create significant barriers to directly applying AMC technologies developed for air-based communication systems. AMC techniques for UAC must be specifically designed to account for the complexities and uncertainties inherent to underwater communication environments. As a result, the development of AMC in UAC

is far behind its terrestrial-based counterpart. So far, AMC in UAC research has generally focused on data-driven methods. Unlike terrestrial wireless communication, the high attenuation, multipath propagation, variable channel conditions, and long propagation delays in underwater environments pose challenges to performing data-driven AMC algorithms in the UAC channels.

Some existing works of AMC in UAC are summarized as follows. Several works perform AMC selection in UAC as a classification problem for which the mode selection metric is produced by ML models [7], [15], [17], [34]. In [34], the AMC procedure is formulated as a classifier that has been trained by a labeled database which helped map the real-time channel state to the corresponding optimal MCSs. The author of [17] proposed a decision tree that was trained to associate channels with modulation schemes under a target BER and all relevant channel characteristics were extracted from large amounts of transmissions from a PSK modem. [7] performs joint key features selection and extraction of CSI instead of all the measured ones via sparse principal component analysis to obtain a faster convergence speed in the AMC system. Work in [15] classified the channels into different types and identified the best MCSs for each channel type in a long-range UAC. These studies employ unsupervised learning to simplify UAC channel characteristics, albeit with constraints on the amount of available MCSs. In [3], recursive least squares, a ML technique, was used to model the statistical properties of the underlying random process of the channel fading and the obtained CSI aided the adaptation of modulation schemes. Limitations of this method were also addressed in that the proposed scheme required sufficient data to estimate the channel coefficients and predict the CSI. However, collecting

high-quality underwater acoustic channel data can be challenging due to factors like changing environmental conditions, equipment limitations, and the cost of conducting sea trials. RL algorithms have also been attempted in designing AMC strategies in UAC. The advantages of applying RL in operating AMC of UAC are demonstrated in [30], [35], [36]. In [35], an online algorithm in a model-based RL framework was proposed to recursively estimate the model parameters of the channels, track the channel dynamics, and compute the optimal transmission parameters to minimize the long-term system costs. A Dyna-Q algorithm that was based on an UAC AMC strategy was developed in [36], which selected the modulation order based on the feedback CSI from the receiver to maximize the long-term throughput. The Dyna-Q algorithm jointly played two roles: predicting CSI and calculating the communication throughput of each modulation order under different channel states for AMC selection. In [30], the authors proposed an RL-based adaptive MCS that can consider multiple quality of service factors, including information Quality of Service requirements, previous transmission quality, and energy consumption. These works prove the ability of RL algorithms to adapt to the highly dynamic and varying underwater channel conditions, learn directly from experience without requiring a precise model of the underwater channel, and handle complex AMC-related variables. However, inevitably, as a purely data-driven approach, the training speed of RL is relatively slow because of the extensive data samples required to account for the time-varying and unpredictable nature of the underwater environment. Given the long propagation delays in UAC, the feedback (or rewards) associated with specific actions may be considerably postponed, complicating the agent's

ability to associate actions with consequences effectively. These delays, coupled with sparse feedback opportunities, limit the RL agent's data collection rate, hampering swift learning.

It is worth knowing that incorporating unsupervised and supervised learning and RL strategies together is an alternative solution in UAC to improve AMC. In [37], facing the outdated CSI problem in UAC channels, an unsupervised learning algorithm helps extract channel features and a Deep Learning (DL) model is trained to find out the relationship between the channel measurements and BER performance in UAC based on a huge data set, and modulation is switched to satisfy BER requirements in a RL framework. [38] proposed a DQN-based AMC method for UAC given the outdated CSI and a long short-term memory neural network was integrated to mitigate any decision bias that was caused by partial observations of UAC channels. A CSI prediction model that is based on online deep learning has been proposed [10] for UAC adaptive orthogonal frequency division multiple access. Considering the channel correlations in both the time and frequency domains, the authors designed a neural network that integrated a one-dimensional convolutional neural network and a long short-term memory network. However, the substantial requirement for training data is an intrinsic limitation of data-driven approaches, often hindering their practical application in real-world scenarios.

2.1.2 Physics-informed Methods in AMC

Traditional data-driven methods rely on the availability and quality of data as we mentioned in the previous literature works, without fully exploiting the

underlying channel physics. On the contrary, physics-informed methods, which incorporate prior knowledge about the underlying physical processes, can be a useful alternative to data-driven methods in the AMC of wireless communication.

Although data-driven approaches have gained more popularity than physics-driven approaches in air-based wireless communication, there are some works attempting to consider channel physics. The work in [39] discussed channel estimation for OFDM systems, which inherently requires understanding the physical propagation parameters of the air-based communication channel such as path delays, path phases, path frequencies, path angles of arrival, etc. The proposed channel estimation methods could be useful for improving the accuracy of AMC algorithms. Like in [40], authors exploited the channel property from [39] to improve the path delay estimation accuracy and reduced the dependency on plenty of pilots to estimate both path delays and path gains via employing the sparse nature of wireless channels to acquire the path gains by only a very small amount of pilots.

The existing literature demonstrates a limited exploration of incorporating channel physics with AMC specifically in the domain of UAC. UAC channels are characterized by unique propagation characteristics governed by aspects like multipath reflections, delay spread, and the influence of ambient noise. Papers [41]–[43] have spotlighted the growing interest in physics-aware paradigms in underwater communication. By embedding channel physics into the AMC design, the models become significantly more reflective of the real-world behavior of these channels. Some researchers have deftly employed physics-based models in conjunction with AMC to mitigate the effects of channel uncertainties.

For instance, authors of [44] proposed the use of the product of Doppler and multipath spreads as a determinant for the adaptive transition between coherent and non-coherent communication techniques. Meanwhile, another study [45] utilized foundational channel attributes, such as received signal strength and noise power spectral density, to estimate frequency domain SNRs, which subsequently informed the adaptive parameter selection.

The synthesis of data-driven methods with foundational channel physics emerges as a potent strategy in the development of AMC techniques. While data-driven methods leverage large datasets and ML algorithms to adapt to varying channel conditions, incorporating channel physics provides a solid foundation rooted in the understanding of how signals propagate in different environments. This union potentially diminishes the AMC design's reliance on expansive training datasets. An illustrative example from [3] revealed how the sparse structure of the channel impulse response can be harnessed to enhance AMC, even with reduced feedback. Here, they put forth a predictor for channel tap coefficients which accounted for channel frequency selectivity and Doppler shifts attributable to the relative motion of transmitters and receivers. This approach noticeably reduced both computational demands and memory overheads. Another compelling case can be drawn from [16], where the authors introduced a hybrid algorithm. It capitalized on channel physics parameters in an OFDM system, including delay spread and channel coherence time, along with methodologies inspired by data-driven algorithms. The primary advantage of integrating channel physics was a noticeable reduction in the number of MCSs needed for AMC, culminating in a faster convergence speed of the AMC strategy.

Collectively, these studies underscore the potential of channel physics-informed approaches, revealing their capacity to offer insights into propagation dynamics. Such approaches undoubtedly bolster the versatility and efficiency of AMC strategies in the intricate domain of UAC.

2.1.3 AMC in OFDM Systems

Recently a great demand for high data rate services has stimulated the development of wideband wireless communication. However, one of the facts that wideband wireless channels always face is frequency selective fading. Therefore multi-carrier modulation technology, especially OFDM has recently emerged as a promising alternative for wideband wireless communications as it can help convert a frequency-selective wideband channel into a set of orthogonal frequency-flat fading channels [46]–[48]. Moreover, OFDM technologies efficiently contrast the ISI [49]. More emphasis is being given to developing efficient coding and modulation schemes in the OFDM system. SNR has been used as the standard measure of the final demodulation signal quality in an OFDM system for a long time. See [49]–[52], a sampling of literature from the field of adaptive modulation, particularly in the cognitive radio domain, employed SNR information which was predicted as the working modem indicator for OFDM systems and helped select the best modulation and coding scheme. Typically, the SNR degradation caused by ISI and ICI due to multipath propagation and channel Doppler spread is often evaluated by the BER or symbol error rate. [53], [54] have shown that BER or SER can characterize the performance degradation more accurately and analytical

approaches to evaluate BER in the OFDM system were implemented to help select the modulation schemes. Similarly, in [55], BER performance comparison of various modulation schemes was proposed and then used to distinguish the SNR ranges matching with different modulation schemes.

2.1.4 Key Performance Indicators in AMC

In the context of AMC, there are multiple Key Performance Indicators (KPIs). These KPIs provide insights into how well the system is adapting its MCSs to the changing channel conditions. Some important KPIs in adaptive modulation include

- BER or Packet Error Rate (PER): BER measures the ratio of incorrectly received bits to the total number of transmitted bits and PER extends the concept of BER to the level of packets. BER or PER are used as the KPI in communication systems that aim to achieve reliable and error-free communication [29], [30], [33]. However, it is worth noting that optimizing solely for low BER might come at the cost of reduced data rates, as higher modulation schemes with lower error rates usually have lower achievable data rates.

- Coded data rate: The coded data rate represents the effective transmission rate after accounting for the error correction coding. The coded data rate can be used as a KPI of the transmission that transfers a large volume of data within a limited time frame [56]–[58].

- Throughput: Throughput refers to the amount of data that can be successfully transmitted over a communication channel within a given

period of time. Currently, maintaining a higher throughput is the main concern in the wireless communication field [36]. In the case of addressing transmission scheduling problems, throughput was often used as the channel performance metric in [3], [59], [60]. Similarly, throughput is provided as the performance metric in a time allocation problem of a wireless-powered communication network [61].

The selection of KPIs must align with the specific aims and prerequisites of a wireless communication system. Given the huge propagation delays in UAC, the time cost to obtain feedback in achieving effective AMC is indispensable. It entails careful transmission and feedback scheduling in the UAC systems. Consequently, throughput is adopted as the assessment metric for our AMC and transmission scheduling. When the propagation delay in the UAC channel is determined, a higher coded data rate usually indicates a potentially higher throughput. Meanwhile, BER facilitates the coding technique, such as the Forward Error Correction (FEC) [62], [63], adaptation [64]–[66] and thereby enables AMC to optimize the channel throughput. However, due to the inherent variability of BER in UAC channels, understanding its distribution instead of relying solely on point predictions is crucial for effective modulation strategy selection under UAC conditions.

Researchers have proposed several models and techniques to estimate the BER in wireless communication systems. For example, [67] proposed an empirical model on BER on the basis of extensive experiments to identify the impact of various parameters, such as the impact of turbo code and

environmental conditions on the BER. Some statistical models proposed in [68]–[70] utilized posterior estimation techniques when no prior knowledge of the channel is available, but need to assume a specific distribution prior or require a huge training set. Specifically, the Monte Carlo error count is regarded as a robust BER estimation strategy [71], [72]. The MC strategy, however, also requires a significant amount of training data to estimate BER [73], [74]. Recently, ML-based approaches have become more popular which employ algorithms like GPR [75], [76], Neural Networks (NN) [77], or support vector machines [78] to estimate BER. They aim to learn the complex relationships between input parameters (such as transmission parameters, channel conditions, and noise levels) and the corresponding BER. Usually, ML is applied in a purely data-driven manner and relies on the availability and quality of data. With channel physics knowledge incorporated, a BER estimation model is proposed in [79] which loosens the demand for the data availability.

2.2 Adaptive Feedback Scheduling

AMC is a prevalent physical layer technique for achieving high throughput over wireless channels. When performing AMC, CSI plays a vital role in facilitating the UAC system to dynamically tune the modulation scheme based on the current channel conditions. Obtaining CSI typically involves feedback from the receiver after decoding frames. In the air-based wireless communication system, conventional AMC systems have the transmitter node requiring CSI from the receiver node in every time slot, causing energy waste. This prompts the emergence of adaptive feedback scheduling algorithms, which determine optimal

instances to adjust MCSs and acquire necessary feedback. In the context of avoiding obtaining CSI feedback very often, there exist some channel-dependent transmission strategies [80]–[82] that exploit temporal correlation in channels to decide on transmission and waiting intervals. However, these adaptive feedback scheduling algorithms were proposed with no channel-dependent variation of MCSs. The authors in [83] use an ML-based feedback scheme that dynamically changes the CSI feedback interval to reduce the feedback overhead, but ignores propagation delays. [84] estimated the interval between consecutive feedback frames along with tuning the modulation orders to optimize energy efficiency.

In UAC channels with larger propagation delays compared with air-based wireless communication, constant feedback incurs more impractical time costs while aiming for high throughput. To this end, the air-based wireless communication protocols that ignore the effects of propagation delays have poor performance in UAC channels. Thus, strategies that balance the need for accurate CSI with the practical constraints of feedback delay are crucial for enhancing the performance and reliability of AMC in UAC systems [85],[86]. The research on dynamically scheduling feedback is still relatively limited. Our goal is to propose a feedback scheduling strategy and determine relevant decision parameters to address the trade-off between communication performance and resource utilization.

2.3 State-of-the-art Algorithms in Markov Decision Processes

Formulating the AMC problem with feedback scheduling as an MDP establishes a structured framework for addressing the intricate decision-making

required to optimize communication performance. MDPs encompass states, actions, transition probabilities, rewards, and policies. States denote system conditions, actions are choices, transition probabilities describe state changes after actions, rewards quantify immediate desirability, and policies dictate action strategies. In MDPs, the cost and transition functions depend solely on the present system state and action. Our aim is to enhance channel throughput in a one-to-one transmission setup, minimizing the time required for transmitting a number of bits. MDPs offer a mathematical representation for sequential decisions, modeling interactions between decision-makers and environments for MCS selection and feedback scheduling. In this section, we explore established algorithms used to find optimal or near-optimal policies in MDPs, particularly from the Tree Search and RL domains. Drawing inspiration from these methodologies, we introduce Tree Search with Deep Q Network (TS-DQN), which merges the planning capabilities of tree search with the generalization potential of DQN.

In the context of MDP, early research has mostly focused on dynamic programming algorithms, such as value iteration [87] and policy iteration [88], which are optimal but impractical for MDPs with extensive state spaces due to memory limitations. The asynchronous variant of value iteration provides a solution for MDPs with large state spaces by avoiding exhaustive state space exploration [89]. Notably, in the significant state spaces, asynchronous approaches like real-time Dynamic Programming demonstrate successful application [90] but still require offline optimization over a variety of training data sets [91]. Recently, tree search and RL have emerged as popular

alternatives to dynamic programming for MDPs. These methodologies extend their applicability by embracing online learning, adaptability, and exploration strategies, which are crucial for addressing intricate real-world challenges featuring large or uncertain state spaces.

Tree search algorithms are well-suited for solving MDPs due to their ability to optimize long-term rewards and effectively manage the exploration-exploitation trade-offs inherent in decision-making problems. These algorithms construct a tree-like search space with nodes representing states and actions, guiding the search process by iteratively simulating state-action trajectories to explore different paths. This exploration balances with exploiting known information to maximize cumulative rewards [92]. However, traditional tree searches face challenges in high-dimensional action or state spaces, entailing uncertainty and computational complexity [93]. Such limitations hinder real-time applications. Efforts have been made to enhance tree search efficiency, as exemplified by the study of state aggregation to reduce stochastic branching [94]. Additionally, Monte Carlo Tree Search (MCTS), originally proposed in the work [95] and [96], stands as a specific tree search variant that often surpasses traditional counterparts. MCTS finds applications in planning [97],[98] and scheduling [99], [100] domains. Therefore, MCTS is one of the core building blocks of games, such as the AlphaGo algorithm [101] while the integration of a Deep Neural Network (DNN) in the AlphaGo algorithm enhances MCTS simulation performance via estimating the value network given the huge action or state space. Similarly, [102] used a NN to estimate the value network for simulations in MCTS. Combining tree search structures with ML techniques has gained

traction for real-time planning and scheduling optimization [103].

RL provides another set of tools for solving MDP problems. RL agents learn to act over time through interactions with the environment, without explicit knowledge of the environment dynamics [104]. Either when a model is not available, or when an explicit representation of the policy is required, the usual approach to applied RL success has been to use expert-developed task-specific features of a short history of observations in combination with function approximation methods. As a widely used RL method, Q-learning is bedeviled by the curse of dimensionality: The computational complexity grows dramatically with the size of state-action space. To combat this difficulty, an integrated hierarchical Q-learning framework is proposed based on the hybrid MDP using temporal abstraction instead of the simple MDP [105]. [106] adapted Q-learning with UCB-exploration bonus to infinite-horizon MDP with discounted rewards without accessing a generative model. However, Q-learning can struggle when faced with environments that have continuous or large action spaces [107]. To address these limitations and enhance the performance of Q-learning, there has been a trend toward combining Q-learning with DL techniques [108]. This combination, often referred to as DQN aims to leverage the strengths of both Q-learning and DL [79], [109]. The use of a Convolutional Neural Network for state representation learning and function approximation in DQN enhances its ability to generalize to unseen states. The authors in [110] proposed a DQN-based MAC protocol for UAC networks, aiming to maximize the total network throughput. Similarly, in [111], DQN is explored in the MAC protocol for UAC to exploit propagation delays inherent in acoustic

communications to improve the network throughput and packet success rate. However, in DQN, quick but possibly biased action selections without planning the potential consequences and future states may result in short-sighted decisions and suboptimal long-term outcomes [112]. Ignoring long-term rewards can lead to suboptimal decision-making that offers immediate gains but hinders long-term success. A look-ahead tree can be a valuable structure for designing long-term rewards in certain scenarios. This planning aspect complements the learned value estimation of DQN and can lead to more informed and strategic decision-making.

We emphasize the importance of integrating RL techniques into search tree structures to address MDPs, particularly in high-dimensional action or state spaces, with a focus on achieving long-term success. Relevant research includes [113], which employs RL and self-play to train value and policy functions within a search tree. Similarly, [114] utilizes a look-ahead tree for guided exploration in RL within complex manipulation tasks. Additionally, the integration of DQN into tree search structures for solving MDPs has been explored. Initial efforts, such as [115], have demonstrated the potential of training DQN via MCTS.

2.4 Summary

In this chapter, a comprehensive literature review on AMC in wireless communication systems is presented, emphasizing the specific application of AMC with feedback scheduling in UAC, which is modeled as an MDP within this thesis. While data-driven AMC has garnered notable accomplishments in terrestrial wireless systems, the unique characteristics of UAC preclude the direct adoption of these terrestrial strategies. Numerous research studies

predominantly focus on data-driven AMC methodologies across terrestrial and underwater domains, but these approaches are often marked by high computational demands and a dependence on the volume and quality of training data. A subset of the literature accentuates the clear benefits of integrating channel physics into AMC, while the fundamental importance of channel physics is frequently neglected. Feedback scheduling in AMC, integral for optimizing channel throughput, remains sparsely explored in the literature. Consequently, considering the formulation of AMC with feedback scheduling as a MDP, this review delves into potential ML algorithms that have demonstrated success in the MDP domain.

Chapter 3

Data-driven AMC with Heuristic Feedback Scheduling Strategy

In Chapter 2, we presented some unique properties of UAC channels such as limited bandwidth, significant propagation delay, and variability. A modulation scheme optimally designed for specific channel conditions may perform poorly when the channel changes which motivates the development of AMC techniques. There is a trade-off between exploration and exploitation in AMC. Exploration involves actively exploring different MCSs to gain knowledge about their performance in varying channel conditions. Exploitation focuses on leveraging acquired knowledge to make optimal decisions and select the most suitable MCS. Excessive exploration wastes resources and time, while excessive exploitation may lead to suboptimal performance if the system fails to adapt to changing conditions.

In this chapter, we apply AMC to enhance the average data rate within a static UAC channel. Initially, we delve into various data-driven strategies and present a novel data-driven method rooted in MCTS. This method prioritizes the long-term maximization of the data rate when determining the sequence of MCSs. A crucial element of effective AMC is the consistent feedback from the receiver to the transmitter, which offers CSI. Yet, in UAC channels characterized

by prolonged propagation delays, regular feedback expedites the AMC model training but potentially extends transmission waiting periods. On the other hand, sparse feedback can result in suboptimal AMC decisions, impacting throughput adversely while conserving time. We, therefore, also introduced an initial integration of feedback scheduling with AMC, supplemented with an overview of heuristic feedback scheduling techniques. This further demonstrates the impact of obtaining feedback for AMC given the propagation delays in UAC systems.

3.1 Problem Formulation

We begin by focusing on a problem where a transmitter and a receiver are placed at a distance l in a static underwater environment. A total of $|\mathcal{A}|$ MCSs in action space $\mathcal{A}$ are available to transmit N bits of information between the transmitter and receiver and each MCS is denoted by $a^i \in \mathcal{A}$, where $i = 1, \cdots, |\mathcal{A}|$. For the j^{th} transmission frame, scheme a^i associated with data rate d^i is selected to transmit frame within a fixed time duration τ and thus each frame might carry a different number of bits. We consider finite-horizon MDPs (file transfer applications) with state space and MCS space denoted by $\mathcal{S}$ and $\mathcal{A}$ respectively. Total N bits will be transmitted in J frames where J is unknown until N bits are all transmitted and $j = 0, 1, \cdots, J$ denotes the index of a state. The probability of frame success γ^i of each scheme a^i is unknown initially. Frames that are successfully received or not can only be known when the feedback information is collected. However, receiving feedback information from the receiver after every transmission turns out to be expensive due to two-way

propagation delay, and therefore we consider feedback frames to be received only when h frames have been transmitted. The number of frames h is named by Feedback Report Interval (FRI) in the following content.

State $\boldsymbol{s_j} \in \mathcal{S}$ is arrived at when the j^{th} frame is transmitted. Each state $\boldsymbol{s_j} = \{N'_j, G_j\}$ is defined by two parameters N'_j and G_j. Here, N'_j is the total number of bits transmitted till state $\boldsymbol{s_j}$, and $G_j = \{m_j^1, \tilde{m}_j^1, \cdots, m_j^{|\mathcal{A}|}, \tilde{m}_j^{|\mathcal{A}|}\}$ denotes a summary of the knowledge of the channel. m_j^i is the number of times scheme a^i has been tried and $\tilde{m}_j^i$ is the number of times scheme a^i has been successful, i.e., no bits were in error after forward error correction at the receiver. Now, at state $\boldsymbol{s_j}$, the probability of success p_j^i of each scheme a^i is estimated as:

$$p_j^i = \frac{\tilde{m}_j^i}{m_j^i}. \tag{3.1}$$

When scheme a^i is selected at state $\boldsymbol{s_j}$, the immediate reward $r_j = d^i\tau$ bits if transmission is successful and 0 bits otherwise. The expected reward D_j when scheme a^i is selected is approximated by:

$$D_j = p_j^i d^i \tau. \tag{3.2}$$

An agent makes decisions on which scheme to select from MCS space $\mathcal{A}$ available at the current state $\boldsymbol{s_j}$. The policy Π is a function that maps from state space to MCS space $\Pi : \mathcal{S} \to \mathcal{A}$. Guided by different policy functions, a scheme $\Pi(\boldsymbol{s_j}) = a^i$ is selected by the agent for the next h frames. After the h frames are transmitted, the receiver responds with an outcome v and the agent

transitions to a new state $\boldsymbol{s_{j+h}}$. The outcome v records the number of successful transmissions during those h transmitted frames. Therefore, the state transition function is represented as $\Gamma(\boldsymbol{s_j}, a^i, h, v) : \boldsymbol{s_j} \rightarrow \boldsymbol{s_{j+h}}$. The updated parameters of the state $\boldsymbol{s_{j+h}} = \{N'_{j+h}, G_{j+h}\}$ are now represented as:

$$G_{j+h} = \{m_j^1, \tilde{m}_j^1, \cdots, m_j^i + h, \tilde{m}_j^i + v, \cdots, m_j^n, \tilde{m}_j^n\}, \tag{3.3}$$

$$N'_{j+h} = N'_j + v d^i \tau. \tag{3.4}$$

Now, the expected reward D_{j+h} in the new state $\boldsymbol{s_{j+h}}$ is:

$$D_{j+h} = p_{j+h}^i d^i \tau. \tag{3.5}$$

We aim to maximize the average data rate over the entire communication sequence through continuous improvement. Exploitation of the gained knowledge through feedback from the receiver usually means selecting valuable schemes to get a maximal immediate reward while exploration is defined as trying new schemes in the MCS space which may bring a greater benefit at the cost of time. Therefore, the policy to select scheme a^i must balance between exploration and exploitation. As shown in (3.3), only when the transmitter obtains the outcome v, the agent can update the next state $\boldsymbol{s_{j+h}}$ and our estimate of p_j^i gets closer to γ^i.

The cost involved in gathering feedback comprises of the propagation delay τ_{pd} and the feedback duration τ_{fd} as illustrated in Fig. 3.1. Rather than following either an exploration or an exploitation strategy, the objective is to

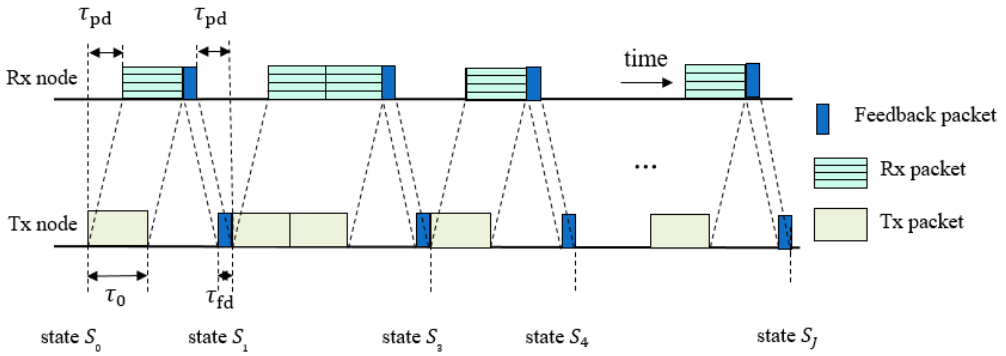


Figure 3.1: An illustration of the delays involved in a typical frame exchange between transmitter and receiver nodes.

investigate policies that target maximizing the long-term average data rate W while transmitting N bits. Now, using a policy function would result in a sequence of MCSs such as $\mathbf{\Pi} = \{\Pi(\boldsymbol{s_0}), \Pi(\boldsymbol{s_1}), \dots\}$. Similarly, an outcome sequence $\mathbf{V}$ is also generated. The outcome sequence $\mathbf{V}$ consists of 1 or 0 indicating either a frame success or a failure. The corresponding data rate sequence is denoted by $\mathbf{d}$. In transmitting N bits of information, it takes a total of J data frames and H feedback frames (both of which are unknown). Therefore, the objective function is formulated as minimizing the total time $T = J\tau + H(\tau_{\text{fd}} + 2\tau_{\text{pd}})$ in transmitting N bits of information:

$$\begin{aligned} &\min \sum_{J} \tau + H(\tau_{\text{fd}} + 2\tau_{\text{pd}}), \\ &s.t. \sum_{q=0}^{J} \mathbf{V}[q]\mathbf{d}[q]\tau = N. \end{aligned} \tag{3.6}$$

3.2 Comparison of Data-driven AMC Strategies

We describe and compare a few well-known strategies here:

3.2.1 Random

An MCS a^i is randomly selected from MCS space $\mathcal{A}$.

$$\Pi(\boldsymbol{s_j}) = a^i, \text{with probability } \tfrac{1}{n}. \tag{3.7}$$

For this policy, if a sufficient number of frames are transmitted, the average data rate W tends to be the mean of n scheme data rates:

$$W = \frac{\sum_{i=1}^{n} d^i}{n}. \tag{3.8}$$

3.2.2 Greedy

A greedy approach always selects a scheme with maximal expected reward D_j, therefore

$$\Pi(\boldsymbol{s_j}) = \underset{a^i \in \mathcal{A}}{\operatorname{argmax}}\, D_j. \tag{3.9}$$

With this approach, when a good scheme fails, the agent is easy to be deceived and it is possible that sub-optimal schemes are recommended.

3.2.3 ϵ-Greedy

ϵ-Greedy policy is more proficient in dealing with the exploration-exploitation dilemma via:

- exploring schemes randomly to avoid missing better choices with probability ϵ;
- adopting greedy policy to help the agent select a scheme with the maximal immediate estimated reward with probability $1 - \epsilon$.

$$\Pi(\boldsymbol{s_j}) = \begin{cases} \underset{a^i \in \mathcal{A}}{\operatorname{argmax}} D_j, & \text{with probability 1-}\epsilon \\ \text{Random}, & \text{with probability } \epsilon \end{cases}. \tag{3.10}$$

3.2.4 Upper Confidence Bound

Upper Confidence Bound (UCB) is the most widely used solution for the exploration-exploitation dilemma in MDPs. The series of transmission successes and failures is formulated as a Bernoulli process. UCB is a family of algorithms and the Wilson score interval developed by Edwin Bidwell Wilson [116] has the asymmetric analytical representation which avoids the *overshoot* and *zero-width interval* problems. Therefore, the Wilson score interval can be safely employed with small samples and skewed observation in our initial transmission phase [117]:

$$\Pi(\boldsymbol{s_j}) = \underset{a^i \in \mathcal{A}}{\operatorname{argmax}} \left(\frac{\tilde{m}_j^i + \frac{1}{2}z^2}{m_j^i + z^2} + \frac{z}{m_j^i + z^2} \sqrt{\frac{\tilde{m}_j^i (1 - \tilde{m}_j^i)}{m_j^i} + \frac{z^2}{4}} \right) d^i \tau, \tag{3.11}$$

where $z = 1.96$ for 95% confidence. The second term inside the bracket is for confidence or used as a measure of the knowledge of every scheme, i.e., for each scheme, the less we understand, the greater the second term. Therefore, this policy selects schemes that have been tried less and continually tends to select schemes with higher estimated rewards. Therefore, UCB policy balances the exploration and exploitation and eventually leads to the optimal scheme.

3.2.5 K-levels Look-ahead in Monte Carlo Tree Search

MCTS is a powerful approach to designing game-playing bots or solving sequential decision problems. Based on the roll-out-based Monte-Carlo planning algorithms [118], we propose a new K-MCTS algorithm that builds its K-level look-ahead tree by repeatedly sampling a sequence with length K of state-action-cost triplets from the current state by Bellman equation (3.13). In such trees, every node denotes one state and at one state, a pair of edges represent successful and failed outcomes with an action selected. Our generic scheme K-MCTS is shown in Fig. 3.2. An action at each state is selected using (3.12) to minimize the cost function:

$$\Pi(S_j) = \underset{a^i \in \mathcal{A}}{\operatorname{argmin}} C(S_j), \tag{3.12}$$

where $C(S_j)$ is the cost involved in a state to adjust the estimated remaining transmission time. During the K-level look-ahead tree, the cost of $\boldsymbol{s_j}$ to S_{j+K-1} is calculated using (3.13), i.e.,

$$\begin{aligned} C(\boldsymbol{s_j}) &= \min_{a^i \in \hat{\mathcal{X}}} \tau + C(S_{j+1}) \\ &= \min_{a^i \in \hat{\mathcal{X}}} \tau + p_j^i C(\Gamma(\boldsymbol{s_j}, a^i, h=1, v=1)) + (1-p_j^i) C(\Gamma(\boldsymbol{s_j}, a^i, h=1, v=0)), \end{aligned} \tag{3.13}$$

but in S_{j+K}, unless the terminal state has arrived the cost of which is 0, the cost is approximated by the average remaining transmitted time with the selected schemes, i.e.,

$$C(S_{j+K}) = \frac{N - N'_{j+K}}{p_j^i d^i}. \tag{3.14}$$

The success probability is estimated by UCB method to avoid *zero* estimation if the selected $\tilde{m}_j^i = 0$ using (3.1) when evaluating the cost using (3.13) and (3.14):

$$p_j^i = \frac{\tilde{m}_j^i + \frac{1}{2}z^2}{m_j^i + z^2} + \frac{z}{m_j^i + z^2}\sqrt{\frac{\tilde{m}_j^i(1-\tilde{m}_j^i)}{m_j^i} + \frac{z^2}{4}}. \tag{3.15}$$

For each K-level look-ahead tree, there are four phases shown in Fig. 3.2:

- *Selection* - A scheme is selected according to (3.12) at each depth. This phase terminates when a terminal state of the problem has been reached.
- *Expansion* - Before reaching a terminal state, *expansion* determines all possible child nodes (states) in K levels. When *expansion* comes to the terminal state (all N has been transmitted), it skips to *backpropagation* phase.
- *Simulation* - Compared with traditional MCTS, our K-MCTS follows an approximate way to implement iterative deepening within K look-ahead levels using (3.13) and (3.14). However, with a larger action space n, the computational complexity increases since all possible actions are considered at each state. Consequently, to narrow down the searching space and decide which ones to throw away, we used the 70^{th} percentile in our simulation, which means that we only sample the 30% with higher results in (3.15) than 70% of the others for $k = 0, \cdots, K$. Then, a smaller set of actions $\hat{\mathcal{X}}$ is used for expansion.
- *Backpropagation* - Propagate the costs back to all states along the path.

Following these 4 phases, a^i is selected and employed for the transmission of

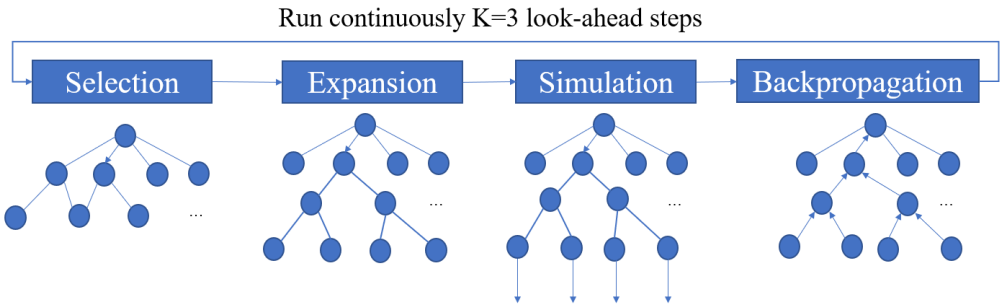


Figure 3.2: Monte Carlo Tree Search phases.

h frames. With feedback, the agent transfers to a new state and backpropagates costs to the initial state. Since the K-MCTS algorithm adjusts between exploration and exploitation by comparing the expected rewards of each scheme, its strategy favors exploration initially and gradually switches to a pure exploitation mode. Algorithm 1 presents a step-by-step procedure to determine an optimal scheme using K-MCTS policy.

Algorithm 1 K-MCTS algorithm

INITIALIZATION: $k = 0$, $j = 0$.
while N bits not finished, **do**
  if $\boldsymbol{s_j}$ is the terminal state **then**
    cost($\boldsymbol{s_j}$) $= 0$.
  end if
  if $k = K$ **then**
    cost($\boldsymbol{s_j}$) according to (3.14).
  else
    cost($\boldsymbol{s_j}$) $= \tau$ + search(S_{j+1}, k+1).
  end if
  $C(\boldsymbol{s_j})$ =cost($\boldsymbol{s_j}$).
  Action $\Pi(S_j) = \underset{a^i \in \hat{\mathcal{X}}}{\operatorname{argmin}} C(S_j)$.
  Calculate FRI h.
  Transmit h frames by $\Pi(S_j)$,
  Feedback is obtained and new state $\boldsymbol{s_{j+h}}$ is arrived;
end while

3.3 Heuristic Feedback Strategies Comparison

The frequency and reliability of feedback information are crucial to perform adaptive modulation efficiently in UAC systems. Due to the large propagation and feedback delay, it is unrealistic to obtain feedback information for every frame that is transmitted. Therefore, an adaptive delay-aware feedback strategy to determine the appropriate number of transmission frames h to be transmitted until the next feedback is necessary. Intuitively, when the channel information is insufficient, providing feedback actively from the receiver is necessary. As the agent gathers more channel information, the feedback interval can be reduced. Therefore different feedback strategies can be employed and are studied as follows.

3.3.1 Fixed Feedback Strategy

A naïve strategy is to have a fixed number of transmission frames between every 2 feedback frames. The transmitter will receive the feedback frame after h frames have been transmitted.

3.3.2 Time-varying Feedback Strategy

With an increasing number of frames transmitted, the agent gathers channel information. Active feedback helps the agent learn the UAC environment quickly in the initial phase and the agent gradually reduces its dependence on the feedback to make decisions in the later phases. Therefore h is approximately given by:

$$h = \lceil \beta j \rceil, \tag{3.16}$$

where β is to determine the change rate of h versus transmitted frames.

3.3.3 Target-oriented Feedback Strategy

A change in channel conditions can render previously learned knowledge invalid, and consequently, the Time-varying feedback strategy might become a poor choice to adopt. We need a more adaptive feedback strategy to adjust h according to the varying channel conditions. If we have an estimate for an achievable data rate w_a in the channel (we call this the *target data rate*), we can calculate the ratio r_w between the immediate data rate w_c of the transmitted h frames and w_a, and use it to adapt the value of h. Although we typically do not know w_a, we can estimate it from our knowledge of the channel:

$$w_a = \max p_j^i d^i, \tag{3.17}$$

and

$$r_w = \frac{w_c}{w_a}. \tag{3.18}$$

The value of h can then be adapted using a sigmoidal transformation:

$$h = \left\lceil \frac{h_m}{1 + e^{-f(r_w)}} \right\rceil, \tag{3.19}$$

in which $f(\cdot)$ is chosen to ensure h stays bounded in the range $[1, h_m]$. The value of h_m is updated according to h' (the value of h from the previous state), and $\triangle r_w$ (the difference between r_w calculated at the previous state $S_{j-h'}$ and the current state $\boldsymbol{s_j}$):

$$h_m = h'(1 + \alpha(\triangle r_w)), \tag{3.20}$$

where $\alpha(\triangle r_w)$ is:

$$\alpha(\triangle r_w) = \begin{cases} (\lg \frac{N}{n})^{\triangle r_w} & \triangle r_w > 0 \\ \frac{\triangle r_w}{\lg \frac{N}{n}} & \triangle r_w \leq 0. \end{cases} \tag{3.21}$$

Equations (3.19)-(3.21) ensure that h follows the change of r_w closely: h will be larger (or close to the maximal value h_m) when r_w increases slightly (or dramatically). If r_w becomes smaller, indicating that our selected scheme is not the most appropriate and we need to reconsider our policy, then a smaller h (or even $h = 1$) is selected to help track the channel rapidly.

3.4 Simulation Results

3.4.1 Discussion on Various Look-ahead Levels without Propagation Delays

In order to select an appropriate look-ahead level K in K-MCTS, we try $K = 0, 1, 2, 3$. The simulation is set with the following parameters:

1. The transmitter and receiver are placed very close, i.e., $l = 0$.

2. Feedback delay duration $\tau_{\text{fd}} = 0$ and the propagation delay $\tau_{\text{pd}} = 0$.

3. The fixed feedback strategy is employed in this section with $h = 1$.

4. When n is set to $2, 5, 10$, the examples of simulation parameters are generated and are tabulated in Table 3.1. The data rate $d^i \in [300, 1500]$ bps is randomly generated and the probability of frame success is generated by a Beta distribution $\gamma^i \sim Be(2, 4)$ but is unknown to the agent. $\hat{w_u}$ is the maximal effective data rate, given by $\hat{w_u} = \max \gamma^i d^i$ and $\hat{w_l}$ is the minimal

effective data rate, given by $\hat{w}_l = \min \gamma^i d^i$.

5. We run 1000 simulations for every n and each is associated with different d^i and γ^i. Similarly, $n = 100$ has also been simulated to test 70^{th} percentile method.

6. Although the transmission duration τ in practical modems might vary, we assume $\tau = 1s$ and hence the length of frames $d^i\tau$ are possible to be different depending on d^i with selected a^i.

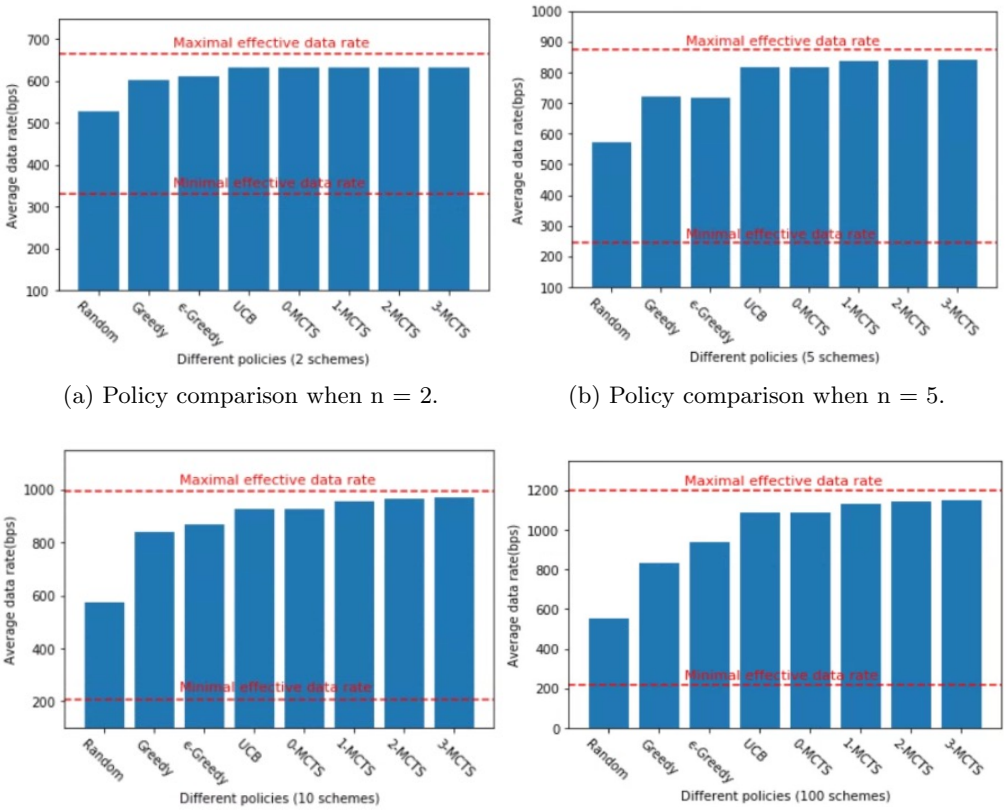
7. The stopping criteria for all policies is when $N = 50000$ bits are successfully transmitted.

8. ϵ is set to be 10% in the ϵ-Greedy policy.

TABLE 3.1: Simulation Parameters

Simulations	**Scheme** a^i	γ^i	d^i/(bps)	$\hat{w}_u$/(bps)	$\hat{w}_l$/(bps)
Simulation 1	x^1	0.12	1204	526.08	144.48
	x^2	0.59	896		
Simulation 2	x^1	0.59	826	768.3	167.2
	x^2	0.28	861		
	x^3	0.61	1090		
	x^4	0.13	1270		
	x^5	0.53	1452		
Simulation 3	x^1	0.87	1371	1189	149
	x^2	0.53	697		
	x^3	0.30	491		
	x^4	0.35	1020		
	x^5	0.30	1391		
	x^6	0.11	1340		
	x^7	0.49	664		
	x^8	0.32	1461		
	x^9	0.46	1141		
	x^{10}	0.41	586		

Simulation results with $n = 2, 5, 10, 100$ are shown in Fig. 3.3. For $n =$



(a) Policy comparison when n = 2.

(b) Policy comparison when n = 5.

(c) Policy comparison when n = 10.

(d) Policy comparison when n = 100.

Figure 3.3: Policy comparison with different numbers of modulation schemes.

$2, 5, 10$, the result of *Random* policy is about 50% between the maximal and minimal effective data rate as expected. UCB and K-MCTS wisely exploit by taking advantage of prior knowledge and explore to try new schemes and hence their advantages are obvious. With the increase of look-ahead levels K, K-MCTS is more prominent. Especially, when look-ahead level $K = 0$ which means no exploration, the action is selected by UCB policy. A similar advantage of K-MCTS is also observed when n is set to 100.

3.4.2 Feedback Strategies Comparison with Propagation Delays

K-MCTS policy outperforms the other strategies and therefore we select this policy for studying the different feedback strategies. For this simulation, the distance between the transmitter and receiver is $l = 1$ km. As the sound speed in underwater environment is around 1500 m/s, the propagation delay $\tau_{\text{pd}} = 0.67$ s. The duration of one feedback frame $\tau_j = 1$ s and the feedback delay $\tau_{\text{fd}} = \tau_{\text{pd}} + \tau_j$. For the Time-varying feedback strategy, β is set to 0.1. For target-oriented feedback strategy, $f(r_w) = 12(r_w - 0.5)$ in (3.19) is helpful to realize $h \in [0, h_m]$ when $r_w \in [0, 1]$. Schemes are generated in Table 3.2 and results are shown in Fig. 3.4. In terms of the Fixed feedback strategies, as the fixed FRI value increases, the average data rate initially rises, reaching its peak at FRI $h = 10$, and then gradually declines. This trend can be attributed to the fact that a smaller fixed FRI results in extended waiting time for feedback. Conversely, when the FRI value is significantly large, such as $h = 40$, the transmission does not receive timely feedback updates, causing wasted time on suboptimal or even inferior MCSs and resulting in unsatisfactory data rates. Similarly, the packet-varying strategy exhibits suboptimal data rate performance initially due to time wasted on waiting for feedback. Notably, the target-oriented feedback strategy outperforms other strategies when the feedback delay duration $\tau_{\text{pd}} > 0$ because of its adaptive FRI adjustment capability based on the transmission progress.

TABLE 3.2: Feedback Strategy Simulation Parameters

Scheme a^i	γ^i	d^i/(bps)	$\hat{w}_u$/(bps)	$\hat{w}_l$/(bps)
x^1	0.26	1158	629.2	32.9
x^2	0.38	984		
x^3	0.06	483		
x^4	0.09	995		
x^5	0.12	441		
x^6	0.44	1253		
x^7	0.28	602		
x^8	0.54	1149		
x^9	0.65	351		
x^{10}	0.68	348		

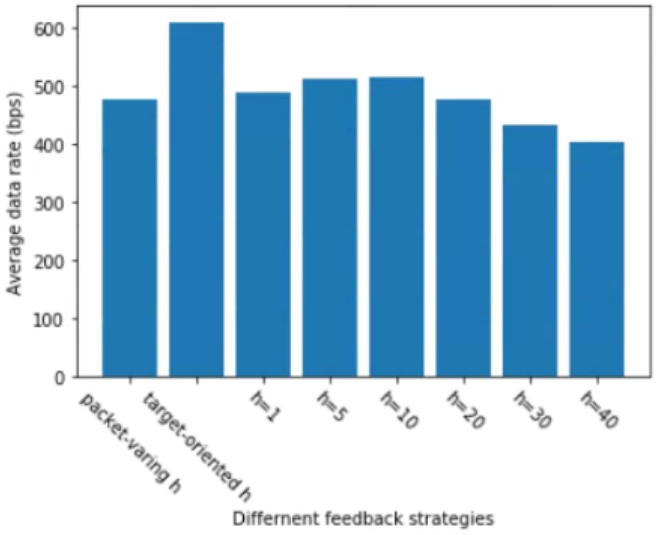


Figure 3.4: Comparison of different feedback strategies.

3.5 Summary

We have evaluated several popular data-driven methods for AMC, including the Random policy, Greedy policy, ϵ-greedy policy, and UCB policy. Additionally, we propose a novel algorithm called K-MCTS, which leverages MCTS to construct a look-ahead tree with K levels. Simulation results demonstrate that the K-MCTS algorithm outperforms the aforementioned

data-driven algorithms in various scenarios while significant training is required for this data-driven method. Furthermore, we examine the impact of propagation delays in UAC on channel throughput by comparing heuristic feedback scheduling strategies. Simulation results for various feedback strategies highlight the dynamic regulation of feedback timing, influencing the AMC convergence rate and reducing transmission time. The implementation of these heuristic feedback scheduling strategies proves indispensable for effective AMC in UAC.

Chapter 4

Physics-informed AMC with Feedback Scheduling by Neural Network

In Chapter 3, we presented a data-driven approach for AMC based on MCTS, i.e., the K-MCTS. Our proposed K-MCTS algorithm achieves a favorable trade-off in AMC, maximizing the average data rate without prior knowledge of the UAC channel. Meanwhile, the data requirement of K-MCTS increases significantly as the size of the action or state space grows due to the expanding search in the tree structure. As shown in Section 2.1.2, the physics-informed methods offer an alternative to data-driven models, reducing the need for extensive training data and computational complexity.

In this chapter, we address the challenge of dealing with an extensive set of modulation schemes in a practical communication system, rendering traditional data-driven methods impractical for real-time AMC. Insufficient data availability and the significant computational complexity associated with training data-driven models motivate the incorporation of channel physics knowledge in the design of the AMC algorithm. Specifically, we focus on OFDM, considering its dominance in contemporary underwater modems, as our study model for implementing a channel physics-informed AMC approach. Within the OFDM framework, we adopt channel throughput as the primary

performance metric for AMC. Notably, in UAC channels, an increased coded data rate typically signifies superior channel throughput. The understanding of BER helps in selecting code rates. Leveraging channel physics information, we construct a heuristic BER estimation model to guide AMC strategy selection. In order to account for the impact of feedback overhead on channel throughput, our UAC system incorporates a feedback scheduling algorithm. Due to difficulties in finding relevant physics information to facilitate the training of the feedback scheduling algorithm, we introduce a DNN-based dynamic feedback mechanism, calibrated to balance the convergence speed of our AMC model and the implications of UAC channel propagation delays, ensuring optimal communication performance.

4.1 Problem Formulation

Information frames are transmitted from a TX node to a RX node deployed at a distance l from the TX. This point-to-point channel is modeled as a Binary Symmetric Channel (BSC) [119]. Prior to each frame transmission, we select a communication scheme $\boldsymbol{a} \in \mathcal{A}$ to suit the current channel conditions, based on the best estimate of the channel that we have. We also decide on a FRI, i.e., the number of transmission frames $h \in \mathcal{H}$ between two consecutive feedback frames. The feedback is sent over a robust link for providing CSI. Therefore the decision space (also known as the *action space*) has a cardinality $|\mathcal{A} \times \mathcal{H}|$. An agent learns a policy Π via continuous interaction with the environment to make sequential decisions on $\boldsymbol{a}$ and h to transmit totally N bits. The policy Π is a function that maps from the state space to the action space, i.e., $\Pi : \mathcal{S} \rightarrow [\mathcal{A} \times \mathcal{H}]$. After

transmitting the j^{th} frame, the agent is in state $\boldsymbol{s_j} \in \mathcal{S}$. State transitions occur following a transition function Γ, based on information received via feedback. Finally, agent transits to the terminal state $\boldsymbol{s_J}$ (and thus $j = 0, \cdots, J$) to have all N bits transferred with total K feedback frames sent out from RX. The round-trip frame exchange duration includes the frame transmission duration τ_j, a two-way propagation delay $2\tau_{\text{pd}}$ and a feedback duration τ_{fd}, also as illustrated in Fig. 3.1.

In state $\boldsymbol{s_j}$, given a modulation scheme $\boldsymbol{a}$ with uncoded data rate $d(\boldsymbol{a})$, BER estimation model $\zeta(\cdot)$ predicts the uncoded BER $\hat{\epsilon}(\boldsymbol{a})$:

$$\hat{\epsilon}(\boldsymbol{a}) = \zeta(\boldsymbol{a}; \boldsymbol{\theta_j}), \tag{4.1}$$

where $\boldsymbol{\theta_j}$ denotes the parameters of $\zeta(\cdot)$, which are adapted as the CSI is updated. Shannon's channel capacity [120] determines the code rate limit $\hat{\rho}(\boldsymbol{a})$ for error-free communication. With the use of a good FEC technique, one can achieve robust communication at rates close to (but strictly less than):

$$\hat{\rho}(\boldsymbol{a}) = 1 - f(\hat{\epsilon}(\boldsymbol{a})), \tag{4.2}$$

where the entropy is computed as $f(x) = -x \log_2 x - (1-x) \log_2 (1-x)$. The transmitter adds redundant bits to the information, forming codewords, resulting in an effective data rate:

$$\hat{D}(\boldsymbol{a}) \approx d(\boldsymbol{a}) \hat{\rho}(\boldsymbol{a}). \tag{4.3}$$

The agent then evaluates a scheme $\boldsymbol{a}$ on the basis of $\hat{D}(\boldsymbol{a})$ and selects the optimal

scheme $\boldsymbol{a}^*$ using:

$$\boldsymbol{a}^* = \underset{\boldsymbol{a}\in\mathcal{A}}{\operatorname{argmax}}\, \hat{D}(\boldsymbol{a}). \tag{4.4}$$

We define a model $\mathcal{M}(\cdot)$ that helps us decide FRI h:

$$h = \mathcal{M}\left(\{N'_j, r_j\}; \boldsymbol{\omega_j}\right), \tag{4.5}$$

where $\boldsymbol{\omega_j}$ represents the trainable parameters of model $\mathcal{M}(\cdot)$. The model operates on two parameters: the immediate throughput r_j, computed using previous h frames, and the percentage of transmitted bits N'_j. The parameters N'_j, and r_j are good indicators of the quality of $\boldsymbol{a}^*$.

After gathering the feedback information, the state $\boldsymbol{s_j} \equiv \{\boldsymbol{\theta_j}, N'_j, r_j, \boldsymbol{\omega_j}\}$ transits to the state $\boldsymbol{s_{j+h}}$ using the state transition function $\Gamma(\cdot)$, i.e.,

$$\begin{aligned} \boldsymbol{s_{j+h}} &= \Gamma(\boldsymbol{s_j}, \boldsymbol{a}^*, h) \\ &= \{\boldsymbol{\theta_{j+h}}, N'_{j+h}, r_{j+h}, \boldsymbol{\omega_{j+h}}\}. \end{aligned} \tag{4.6}$$

In transmitting N bits, it takes J data frames and K feedback frames (both of which are unknown). We wish to minimize the total time to transmit all N bits:

$$\text{minimize} \sum_{i=0}^{J} \tau_i + K(\tau_{\text{fd}} + 2\tau_{\text{pd}}). \tag{4.7}$$

4.2 AMC Strategy

In this section, we delve into the adaptation strategy of MCSs, focusing on OFDM given its prevalence in modern underwater modems. Leveraging

channel physics information, we construct a heuristic BER estimation model to guide AMC strategy selection. We validate this model using a data set collected from Singapore waters. Given the BER estimation for every possible modulation scheme, we suggest using a dynamic ϵ-greedy policy to address the exploration-exploitation dilemma in modulation scheme selection given the high-dimensional MCSs space.

4.2.1 BER Estimation Model

We consider a modem that uses OFDM for communication but allows several parameters to be tuned. There are two key parameters in OFDM: the cyclic prefix length n_{p} and the number of subcarriers n_{c}. Another important parameter is the bandwidth B occupied by the OFDM signal. Before a frame is transmitted, $n_{\mathrm{c}}, n_{\mathrm{p}}$ and B are selected to optimize for the performance. An AMC scheme $\boldsymbol{a}$ is therefore defined as a point in $(n_{\mathrm{c}}, n_{\mathrm{p}}, B)$ space. The uncoded data rate $d(\boldsymbol{a})$ is then:

$$d(\boldsymbol{a}) = \frac{mBn_{\mathrm{c}}}{n_{\mathrm{c}} + n_{\mathrm{p}}}, \tag{4.8}$$

where m is the number of bits per PSK symbol on each subcarrier used for the underlying OFDM carrier modulation (e.g. $m = 1$ for BPSK and $m = 2$ for QPSK).

In [16], the channel delay spread τ_{ds}, channel coherence time τ_{c} and, bandwidth B, were utilized to define boundaries c_1, c_2 and c_3

$$n_{\mathrm{c}} > 2\pi B\tau_{\mathrm{ds}} = Bc_1, \tag{4.9}$$

$$n_{\mathrm{p}} > B\tau_{\mathrm{ds}} = Bc_2, \tag{4.10}$$

$$n_{\mathrm{c}} + n_{\mathrm{p}} < B\tau_{\mathrm{c}} = Bc_3, \tag{4.11}$$

in the $(n_{\mathrm{c}}, n_{\mathrm{p}})$ plane. These boundaries divided the region into a relatively good region represented by blue shaded color and a bad region represented by red shaded color (see Fig. 4.1 which is reproduced from [27]). Schemes inside the good region are more likely to achieve a higher frame success rate which is usually associated with a lower uncoded BER [121]. In line with [16], a sigmoid function

$$s(d) = \frac{1}{1 + e^{-b_i d}}, \quad i = 1, 2, 3, \tag{4.12}$$

is utilized to characterize the BER estimation model based on the relative position of the point $(n_{\mathrm{c}}, n_{\mathrm{p}})$ with respect to the three boundaries c_i, $i = 1, 2, 3$ (see Fig. 4.1). The slope of the three sigmoid functions is controlled by b_i, $i = 1, 2, 3$. Additionally, there is a relationship between the bandwidth B and the BER as a broader bandwidth is likely to contain more noise and thus might result in a higher BER [122]. Based on this, a simple parametric BER estimation model $\zeta(\boldsymbol{a}; \boldsymbol{\theta})$ to estimate uncoded BER $\hat{\epsilon}(\boldsymbol{a})$ is proposed as:

$$\zeta(\boldsymbol{a}; \boldsymbol{\theta}) = (b_4 B + c_4) s(-d_1) s(-d_2) s(d_3), \tag{4.13}$$

$$d_1 = n_{\mathrm{c}} - Bc_1, \tag{4.14}$$

$$d_2 = n_{\mathrm{p}} - Bc_2, \tag{4.15}$$

$$d_3 = \frac{n_c + n_p - Bc_3}{\sqrt{2}}, \tag{4.16}$$

where d_1, d_2, d_3 are distances as shown in Fig. 4.1 and $\boldsymbol{\theta} \equiv (c_1, c_2, c_3, c_4, b_1, b_2, b_3, b_4)$. To enhance the model's accuracy, it is vital to measure how closely the model's estimates align with actual BER values. Hence, we introduce a loss function $L(\boldsymbol{\theta_j})$ for training $\zeta(\cdot)$. The loss function evaluates the Mean Absolute Error (MAE) between the output of $\zeta(\boldsymbol{a}; \boldsymbol{\theta_j})$ and the measured BER $\boldsymbol{\epsilon_j}(\boldsymbol{a})$, i.e.,

$$L(\boldsymbol{\theta_j}) = \frac{1}{|\mathcal{A}|} \sum_{\boldsymbol{a} \in \mathcal{A}} (|\zeta(\boldsymbol{a}; \boldsymbol{\theta_j}) - \boldsymbol{\epsilon_j}(\boldsymbol{a})|). \tag{4.17}$$

Through the minimization of $L(\boldsymbol{\theta_j})$ at state $\boldsymbol{s_j}$ using techniques like gradient descent, the model's weight parameters, $\boldsymbol{\theta_j}$, of our model, are refined, enhancing its BER estimation during transmission. This iterative refinement utilizes measured BER data, ensuring the model's predictions remain closely aligned with empirical observations.

4.2.2 Validation of BER Estimation Model

The BER estimation model presented in Section 4.2.1 is validated using experimental data that was collected in Singapore waters. Subnero M25M modems operating in the 18 to 32 kHz band were used for this data collection. The transmission range between the TX and RX was about 600 m, and the water depth was between 10 and 20 m. The BER for 1979 schemes $\boldsymbol{a} = (n_c, n_p, B)$ were measured, where n_c was set to different values from the set $\{64, 128, 256, 512, 1024, 2048\}$ and n_p ranged from 0 to 2046.

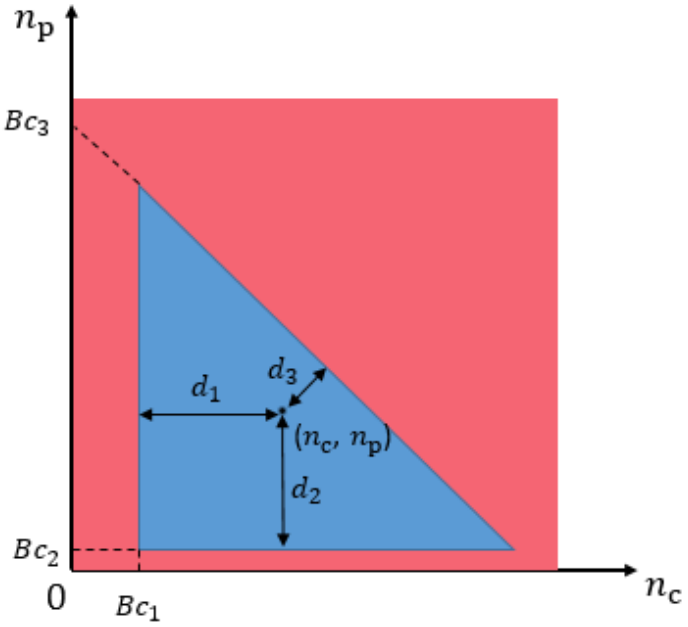


Figure 4.1: Visualization of the boundaries c_1, c_2 and c_3 in the (n_c, n_p) plane.

Notes: This figure is reproduced from [27]. Three boundaries c_1, c_2 and c_3 are with bandwidth B fixed in the (n_c, n_p, B) space.

An ADAM optimizer [123] together with a maximal absolute error loss was utilized to train $\boldsymbol{\theta}$ on 70% of the data set. In Fig. 6.5, we compare the BER $\hat{\epsilon}$ estimated using the parametric model $\zeta(\cdot)$ with the actual BER ϵ measured for each (n_c, n_p, B) in the remaining 30% of the data set. Since the measured BER was observed to be similar between adjacent values of n_p, the result is only plotted for every n_c with n_p grouped with a bin size of 128. Furthermore, since the effect of B was observed to be small, all values of B for which the measurements were performed are clubbed together in the plot. We observe that the model $\zeta(\cdot)$ is able to approximate the median BER from sea measurements well.

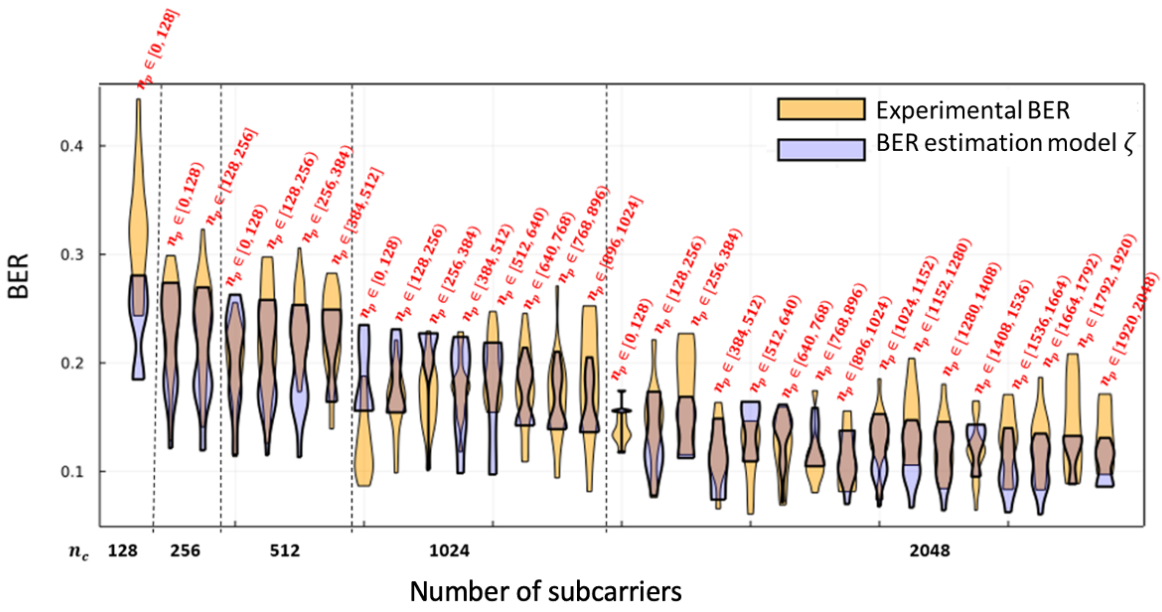


Figure 4.2: Comparison of the measured BER from a field experiment and the BER estimated by model $\zeta(\cdot)$.

4.2.3 Scheme Selection Policy

Since we have no knowledge of the quality of the communication link beforehand, the agent needs to make decisions on $\boldsymbol{a}$ while learning about the channel information via feedback. A well-known problem that occurs in scenarios like this is the trade-off between exploration and exploitation of different schemes. We need to select $\boldsymbol{a}$ given $\hat{\epsilon}(\boldsymbol{a})$ estimated using $\zeta(\boldsymbol{a}; \boldsymbol{\theta_j})$. Should we repeat decisions with lower $\hat{\epsilon}(\boldsymbol{a})$ (exploit) or select schemes that are never tried before hoping to gain greater rewards and expand the channel knowledge (explore)? The adaptive ε-greedy policy is a simple but efficient strategy to solve the

explore-exploit dilemma and determines scheme $\boldsymbol{a}^*$ as follows:

$$\boldsymbol{a}^* = \begin{cases} \underset{a \in \mathcal{A}}{\operatorname{argmax}} \Big(d(\boldsymbol{a}) \hat{\rho}(\boldsymbol{a}) \Big), & \text{with probability } 1 - \varepsilon \\ \text{Random}, & \text{with probability } \varepsilon \end{cases}, \tag{4.18}$$

where ε gradually decreases from 1 according to the common ratio ε_{d} along with the transmission of frames.

4.3 Neural Network for Feedback Scheduling

Gathering CSI via feedback from the receiver is essential for any AMC technique. However, due to the long propagation delays in underwater acoustic communications, waiting for feedback on every individual frame is expensive, and hence a policy to determine the FRI h that adapts with channel is necessary. A fixed FRI is commonly used in the literature while being able to adapt h to optimize for achieving higher throughput is an interesting problem. Conventional regression algorithms may help determine FRI h, but they require a large number of samples to train on. Such data with many different values of h is usually hard to obtain. A feedback strategy based on a heuristic sigmoidal function in [124] utilizes the immediate data rate of the previous FRI h to deduce the next h and was demonstrated to achieve a significant reduction in the feedback overhead. We aim to improve on this heuristic and design a more generic adaptive feedback strategy next.

While in state $\boldsymbol{s_j}$, we receive the feedback and therefore update N'_j. The number of bits in each frame n_j and the frame duration τ_j remain unchanged in a particular transaction once an action $\boldsymbol{a}$ is selected. Therefore, the immediate

throughput is computed using:

$$r_j = \frac{hn_j}{h\tau_j + 2\tau_{\text{pd}} + \tau_{\text{fd}}}. \tag{4.19}$$

Now that we know, r_j, N'_j and previous FRI, we are interested in determining h as a function of these. However such a function is analytically unknown, and so we turn to ML to learn such a function. We build the model $\mathcal{M}(\cdot)$ using a simple 3-layer Neural Network (NN) with input being $\{r_j, N'_j, h\}$ and output as the predicted throughput $\tilde{r}_{j+h}$ and therefore,

$$h^* = \underset{h\in\mathcal{H}}{\text{argmax}}\, \mathcal{M}\left(\left\{N'_j, r_j, h\right\}; \boldsymbol{\omega_j}\right), \tag{4.20}$$

where $\boldsymbol{\omega_j}$ contains the parameters of $\mathcal{M}(\cdot)$. The ADAM optimizer was used for training of $\boldsymbol{\omega_j}$ to minimize the MSE loss between the actual r_{j+h} and predicted $\tilde{r}_{j+h}$.

Since the action space $[\mathcal{A}\times\mathcal{H}]$ is large, collecting samples, i.e., $\{r_j, N'_j, h\} \rightarrow r_{j+h}$, for training $\mathcal{M}(\cdot)$ in real time is infeasible. We, therefore, propose a method to generate training samples through simulation, and to pre-train an initial estimate of parameter vector $\boldsymbol{\omega_0}$ (see Algorithm 2). We assume $\bar{\boldsymbol{\theta}}$ and thus $\zeta(\boldsymbol{a};\bar{\boldsymbol{\theta}})$ represent specific ocean environments. To simulate the uncertainty in the environment, we generate random errors in a frame containing n bits following a Poisson distribution P, and thus the BER of scheme $\boldsymbol{a}$ is $\epsilon(\boldsymbol{a}) = \frac{P(\zeta(\boldsymbol{a};\bar{\boldsymbol{\theta}})n)}{n}$. With different values of $\bar{\boldsymbol{\theta}}$, we collect training samples by simulating transmission with scheme $\boldsymbol{a}$ and h. To generate different values of $\bar{\boldsymbol{\theta}}$, we take inspiration

from measured values of τ_{ds} and τ_{c} in different ocean [125]–[128], and randomly generate τ_{ds} less than 10 ms and τ_{c} between 0.01s and 2s and compute c_1, c_2, c_3. Without an obvious quantitative relationship between the remaining parameters of $\bar{\boldsymbol{\theta}}$ and BER, b_4 and c_4 are uniformly generated from 0 to 1. Slopes b_1, b_2, b_3 are also uniformly generated between 0 and 1.

Algorithm 2 Algorithm to obtain an initial parameter estimates $\boldsymbol{\omega_0}$

Initialize state $\boldsymbol{s_0} = \{\boldsymbol{\theta_0}, N_0' = 0, r_0 = 0, \boldsymbol{\omega_0}\}$ where $\boldsymbol{\theta_0}$ and $\boldsymbol{\omega_0}$ are randomized.
Generate $\bar{\boldsymbol{\theta}}$.
while $N_j' < 1$ **do**
  Select $\boldsymbol{a}$ using (4.18) and randomly select $h \in \mathcal{H}$.
  Transmit frames and receive feedback.
  Perform state transition $\boldsymbol{s_j} \rightarrow \boldsymbol{s_{j+h}}$:
  Update $\boldsymbol{\theta_j}$ based on $\boldsymbol{a}$ and ϵ from $\zeta(\boldsymbol{a}; \bar{\boldsymbol{\theta}})$.
  Update $\boldsymbol{\omega_j}$ based on $\{r_j, N_j', h\} \rightarrow r_{j+h}$.
  Update N_{j+h}' and r_{j+h}.
end while
Reset initial state $\boldsymbol{s_0} = \{\boldsymbol{\theta_0}, N_0' = 0, r_0 = 0, \boldsymbol{\omega_0}\}$ where $\boldsymbol{\theta_0}$ is randomized while the final $\boldsymbol{\omega_J}$ during the previous simulation is assigned to $\boldsymbol{\omega_0}$.
Back to line 2 until training ends.
Output $\boldsymbol{\omega_0}$.

4.4 Simulation Results

We compare the proposed feedback strategy of adaptive model $\mathcal{M}(\cdot)$ with the pre-trained $\boldsymbol{\omega_0}$ with a *Random* strategy, where h is randomly picked between 1 and 50, and the Fixed strategy where $h = 5, 10, 15, \cdots, 50$ is fixed a priori. Transmission distance l is set to 3 km and $\tau_{\text{fd}} = 100$ ms. The number of bits per PSK symbol m is set to 1 in (4.8) and $\varepsilon_{\text{d}} = 0.9$ in the adaptive ε-greedy policy. The simulation stops when $N = 100000$ bits are transferred. The number of sub-carriers are set to $n_{\text{c}} = 64, 128, 256, 512, 1024, 2048$, and n_{p} ranges from 0 to 2048. 100 different $\bar{\boldsymbol{\theta}}$ values are generated to obtain a pre-trained $\boldsymbol{\omega_0}$. For different feedback strategies, policy Π is executed 100 times, and the average

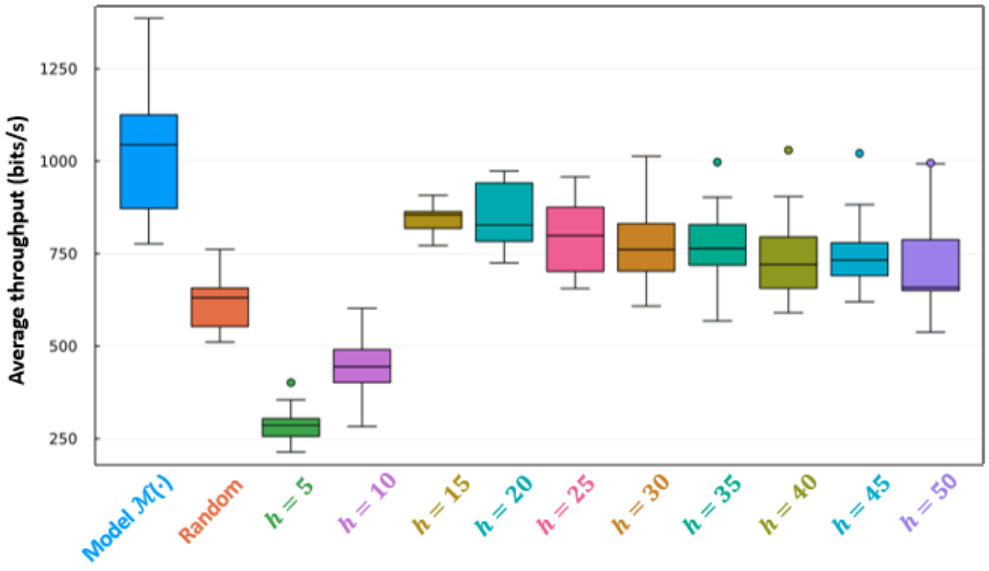


Figure 4.3: Average throughput on a point-to-point link when using different feedback strategies.

throughput computed is shown in Fig. 4.3. We observe that the policy Π when using *Random* strategy provides a throughput that is significantly poorer than what can be achieved via the proposed adaptive strategy. When using the *Regular* strategy, the median throughput initially increases with h, but gradually reduces when the value of h increases beyond a point. While outliers at $h = 35, 40, 45$ demonstrate higher throughput results compared to $h = 15$, which achieves the highest median among all Regular FRI strategies, our emphasis lies on optimizing medians to ensure relatively robust transmission performance. Selecting the optimal h a priori is generally difficult, but our proposed adaptive policy is able to select it dynamically.

4.5 Summary

In this chapter, We evaluated the joint effect of AMC and the feedback strategy on the average throughput of a communication link. With the

assistance of channel physics, our proposed BER estimation model for AMC achieves satisfactory channel performance with low computational complexity. An adaptive feedback strategy proved to be essential in deciding the appropriate times at which the feedback needs to be sent out.

Chapter 5

Adaptive Feedback Scheduling Strategy

In Chapters 3 and 4, we undertook an exploration of two distinct approaches, data-driven and physics-informed methods, to enhance the efficiency of AMC in UAC. In addition, we elaborate on the need to consider feedback scheduling for optimizing the performance of AMC. The feedback scheduling involves deciding the appropriate time to obtain CSI through feedback from the receiver and tuning the modulation and coding schemes. However, in UAC systems characterized by long propagation delays, the process of tuning modulation schemes $\boldsymbol{a}$ and waiting for feedback on each frame introduces delays. Additionally, early-stage communication with incomplete channel knowledge or channel variability during transmission can lead to sub-optimal AMC strategies. Insufficient feedback collection under such sub-optimal strategies can further diminish data throughput.

In Chapter 3, we presented a heuristic strategy addressing feedback scheduling. This strategy focuses on the real-time performance of the current AMC policy and exhibits sensitivity to dynamic channel conditions. However, its applicability is limited by a lack of generality. In Chapter 4, generality was improved through the use of a NN algorithm. Nevertheless, this approach might yield sub-optimal results due to the inclination toward optimizing short-term

throughput, and its efficacy relies on a substantial training dataset that hinders practical experimentation.

To address these challenges, we propose an algorithmic that combines the advantages of tree search due to its planning ability, and DQN due to its capacity to learn from interactions with the environment and generalize to unseen states in complex MDPs. Illustrated through the example of feedback scheduling in AMC, our goal with this framework is to showcase its proficiency in enhancing channel throughput by adaptively deciding modulation scheme modifications and feedback timings based on channel conditions.

5.1 TS-DQN for Feedback Scheduling

The sequential decision process involving the modulation scheme $\boldsymbol{a}$ and FRI h_j for transmitting N bits is formulated as an MDP. Tree search algorithms are suitable for solving MDPs as they optimize long-term rewards and balance exploration-exploitation trade-offs. However, traditional tree search algorithms face computational challenges when building search trees in high-dimensional action or state spaces. Uncertainty in untried state-action pairs is initially unknown. In large action and state spaces, handling uncertainty in tree search also introduces significant computational complexity, potentially resulting in suboptimal decisions.

RL is another popular method that enables the agent to learn an optimal policy in MDPs through interactions with the environment, without explicit knowledge of the environment's dynamics [104]. DQN, a powerful RL algorithm, aims to learn a mapping function that predicts the expected reward for

state-action pairs, known as the Q-value function [79], [109]. The use of DNN as function approximators in DQN enhances its ability to generalize to unseen states. However, in DQN, quick but possibly biased action selections without planning the potential consequences and future states may result in short-sighted decisions and suboptimal long-term outcomes [112].

Therefore, We propose Tree Search with DQN (TS-DQN) to benefit from the planning capabilities of tree search and the generalization capabilities of DQN. DQN focuses on learning the optimal Q-value function for state-action pairs using observed experiences, even leveraging its generalization abilities to approximate the value function in a continuous state space. Tree search utilizes the updated Q-values to guide the exploration process, prioritize promising state-action pairs, and provide a way to estimate long-term rewards. Fig. 5.1 illustrates the fundamental framework of the proposed TS-DQN algorithm and Fig. 5.2 further depicts the details of the tree search procedure. We will explain them in the rest of this section.

5.1.1 Q-value Function

agent aims to select $\boldsymbol{a}$ and h_{j+1} at state $\boldsymbol{s_j}$ to maximize the expected throughput until reaching the terminal state $\boldsymbol{s_J}$. In Fig.5.1, the output of DQN provides a prediction of the Q-value function, i.e., $\hat{Q}(\boldsymbol{s_j}, \boldsymbol{a}, h_{j+1})$. $\hat{Q}(\boldsymbol{s_j}, \boldsymbol{a}, h_{j+1})$ approximates the throughput till terminal state $\boldsymbol{s_J}$ given any state-action pair $\{\boldsymbol{a}, h_{j+1}\}$. Agent prioritizes actions that are more likely to result in favorable

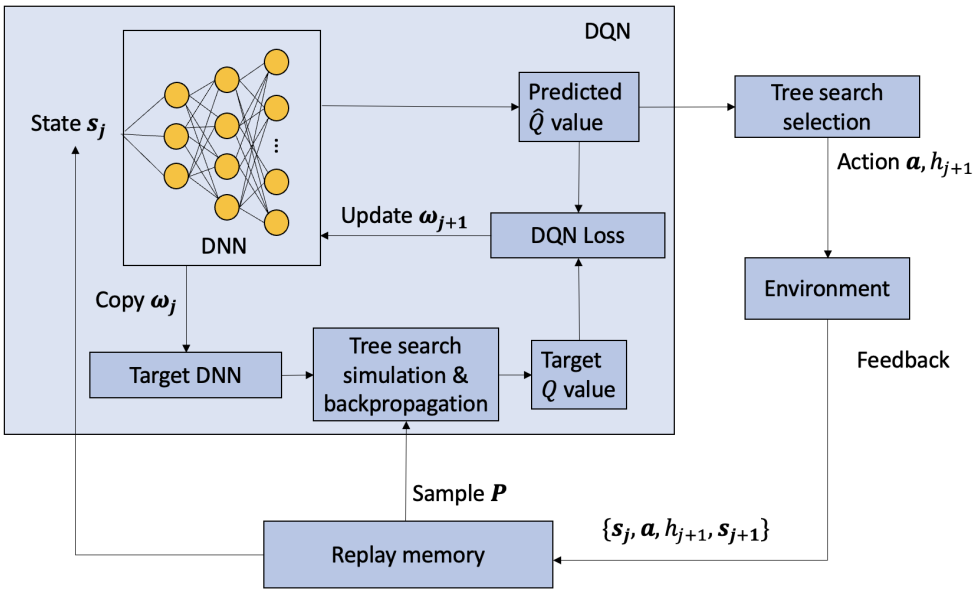


Figure 5.1: The framework of the TS-DQN algorithm.

long-term rewards. At state $\boldsymbol{s_j}$, the throughput R_j should be calculated by

$$R_j = \frac{(1-n'_j)N}{\sum_{i=j+1}^{J} h_i\tau_i + (J-j)(\tau_{\mathrm{fd}} + 2\tau_{\mathrm{pd}} + \tau_{\mathrm{m}})}. \tag{5.1}$$

The feedback of FRI h_j from the receiver node updates $(n'_{j+1} - n'_j)N$ bits transmitted during FRI h_j within the time period encompassing h_{j+1} frames with a transmission duration of τ_{j+1} each, the duration of frames containing modulation information τ_{m} and feedback τ_{fd}, and a two-way propagation delay of $2\tau_{\mathrm{pd}}$. Then in DQN, the target Q-value is updated by

$$Q(\boldsymbol{s_j}, \boldsymbol{a}, h_{j+1}) = \frac{(n'_{j+1} - n'_j)N + (1-n'_{j+1})N}{h_{j+1}\tau_{j+1} + \sum_{i=j+2}^{J} h_i\tau_i + (J-j)(\tau_{\mathrm{fd}} + 2\tau_{\mathrm{pd}} + \tau_{\mathrm{m}})}. \tag{5.2}$$

However, from state $\boldsymbol{s_{j+1}}$ to the terminal state $\boldsymbol{s_J}$, the number of transmitted bits and their corresponding duration are unknown as they have not been attempted. In the upcoming subsection 5.1.4, we will present the tree search approach for approximating the target Q-value $Q(\boldsymbol{s_j}, \boldsymbol{a}, h_{j+1})$.

5.1.2 State-value Approximation

Traditional tree search methods lack explicit policies and require repeated tree building at each state. This procedure can be time-consuming and memory-intensive. Our proposed TS-DQN algorithm addresses these limitations by leveraging the approximated Q-value function to learn the rewards associated with potential state-action pairs during repeated look-ahead tree construction.

In Fig. 5.1, the structure of DNN is one input layer with a size of all possible states, three hidden layers, and one output layer with the size of the $\mathcal{H}$. This DNN is utilized to model the analytical function between the state $\boldsymbol{s_j}$, and the estimated reward given the selected $\boldsymbol{a}$ and any possible value of h_{j+1} which represents the predicted Q-value $\hat{Q}(\boldsymbol{s_j}, \boldsymbol{a}, h_{j+1})$, i.e.,

$$\hat{Q}(\boldsymbol{s_j}, \boldsymbol{a}, h_{j+1}) = \mathcal{M}\left(\boldsymbol{s_j}; \boldsymbol{\omega_j} | \boldsymbol{a}, h_{j+1}\right), \tag{5.3}$$

where $\boldsymbol{\omega_j}$ is the weights of model $\mathcal{M}(\cdot)$ and updated once CSI received.

5.1.3 Replay Memory

As shown in Fig. 5.1, agent utilizes a Replay Memory buffer to train the TS-DQN where the experiences of the agent, $p_i = \{\boldsymbol{s_i}, \boldsymbol{a}, h_{i+1}, \boldsymbol{s_{i+1}}\}$, $i \in [0, j]$, are stored up to state $\boldsymbol{s_j}$. When updating the TS-DQN within the state

transition, the target DNN is a copy of the DNN model with the weight parameters $\boldsymbol{\omega_j}$ updated in state $\boldsymbol{s_j}$. A batch, denoted by $\mathbf{P}$, including 32 samples from the Replay Memory buffer. For each $p_i \in \mathbf{P}$, the target throughput from state $\boldsymbol{s_i}$ until reaching the terminal state $\boldsymbol{s_J}$, i.e., $Q(\boldsymbol{s_i}, \boldsymbol{a}, h_{i+1})$, is approximated by the tree search shown in Fig. 5.2. During each memory replay for training TS-DQN, the parameters of the target DNN remain fixed across the training batch.

An ADAM optimizer is employed to minimize the DQN Loss based on the mean squared loss between the predicted $\hat{Q}$-value and the target Q-value to update the weight parameters $\boldsymbol{\omega_j}$ to $\boldsymbol{\omega_{j+1}}$ of our state-value approximator $\mathcal{M}(\cdot)$, i.e.,

$$\boldsymbol{\omega_{j+1}} = \underset{\boldsymbol{\omega_{j+1}}}{\operatorname{argmin}} \left(\sum_{p_i \in \mathbf{P}} (Q(\boldsymbol{s_i}, \boldsymbol{a}, h_{i+1}) - \hat{Q}(\boldsymbol{s_i}, \boldsymbol{a}, h_{i+1}))^2 \right). \tag{5.4}$$

5.1.4 Tree Search

Fig. 5.2 presents the structure of our lookahead tree search structure with a three-section procedure: Selection, Simulation, and Backpropagation. When reaching the state $\boldsymbol{s_j}$, the three sections are depicted as follows.

- Selection: Agent selects modulation scheme $\boldsymbol{a}$ first and determines FRI h_{j+1} following a ϵ-greedy strategy, where ϵ is for exploring FRI randomly to avoid being trapped in a local optimum and with probability $1-\epsilon$, agent tends to choose FRI as

$$h_{j+1} = \underset{h_{j+1} \in \mathcal{H}}{\operatorname{argmax}} \hat{Q}(\boldsymbol{s_j}, \boldsymbol{a}, h_{j+1}). \tag{5.5}$$

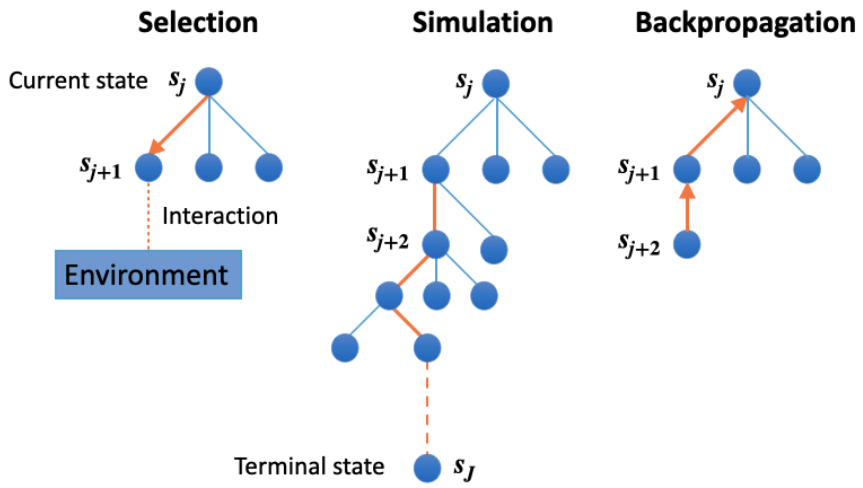


Figure 5.2: Tree search structure.

After determining $\boldsymbol{a}$ and FRI h_{j+1}, frames are encoded for transmission, and feedback is obtained to update the AMC strategy and the next state $\boldsymbol{s_{j+1}}$. The weight parameters of model $\mathcal{M}$ are maintained as $\boldsymbol{\omega_j}$ in $\boldsymbol{s_{j+1}}$ and they will be updated later during the Replay Memory. The tuple $p_j = \{\boldsymbol{s_j}, \boldsymbol{a}, h_{j+1}, \boldsymbol{s_{j+1}}\}$ is then stored in the Replay Memory.

- Simulation: For each $p_i = \{\boldsymbol{s_i}, \boldsymbol{a}, h_{i+1}, \boldsymbol{s_{i+1}}\}$, $i \in [0, j]$ of the sample $\mathbf{P}$ from Replay Memory, the agent conducts simulations from the newly added node $\boldsymbol{s_{i+1}}$ to the end of the transmission. Throughout the simulation, the target DNN provides the estimated value associated with any potential state-action pairs and makes greedy move selections until a terminal state is reached.

- Backpropagation: The outcomes of the simulation, including FRI values, throughput, and timestamps of each FRI between visited states, are backpropagated up the tree to update the target Q-value in (5.2).

The target Q-value for each $p_i \in \mathbf{P}$ is collected to train $\boldsymbol{\omega_{j+1}}$ using the loss function defined in (5.4). Then the tuple $\{\boldsymbol{s_j}, \boldsymbol{a}, h_{j+1}, \boldsymbol{s_{j+1}}\}$ collected from the state $\boldsymbol{s_{j+1}}$ is updated with the new value of $\boldsymbol{\omega_{j+1}}$ and stored in the Replay Memory.

5.2 Simulation Validation

We verify our TS-DQN algorithm with a toy problem. This problem is formulated similarly to Section. 4.1. The problem scenario encompasses a high-dimensional action and state space, providing a sufficiently complicated context for robust evaluation of our feedback scheduling approach. Frames are transmitted between a TX node and a RX node deployed at a distance $l = 3$ km from the TX node. A large file with $N = 100,000$ bits is requested to be transmitted in the communication system within the shortest possible time. Given the varying nature of the UAC environment, AMC is employed to select MCS for every frame based on the current channel conditions.

We build a surrogate model to represent an UAC channel based on [129] for generating the BER given any MCS selected during simulation. In the surrogate model, the Pekeris ray model with red Gaussian noise is employed. The Pekeris ray model is a very fast fully differentiable 2D/3D ray model for isovelocity range-independent environments. In the Pekeris ray model, the isovelocity sound speed profile is with sound speed $c = 1500$ m/s, and we assume a flat sea surface and a sandy clay seabed. We choose the variance in the red Gaussian noise to be $1e^6$ which is aligned with Singapore waters. The depth of the TX node and RX node $d_1 = 25$ and $d_2 = 25$. The number of bits per PSK symbol on each

subcarrier m is set to 2 in (4.8). The timing diagram is aligned with Fig. 3.1 where the propagation delay $\tau_{pd} = 2$ s and the feedback duration τ_{fd} is fixed as 100 ms.

We perform AMC in OFDM system. The possible number of subcarrier values are $n_{\mathrm{c}} = 64, 128, 256, 512, 1024, 2048$, the value n_{p} can be selected from 0 to 2048, the possible occupy ratio of the bandwidth 24 KHz is $0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9$. For any MCS $\boldsymbol{a} = \{n_{\mathrm{c}}, n_{\mathrm{p}}, B\}$, BER prediction will be given by Section. 4.2.1. We aim to compare our TS-DQN for feedback scheduling with the NN-based model we have proposed in Section. 4.3. Therefore, in our TS-DQN, the state $\boldsymbol{s_j}$ to estimate the $\hat{Q}$ value to select the next FRI value in (5.3) is the same as that we have in (4.20), i.e., $\{N'_j, r_j\}$. The parameter N'_j calculated the transmitted percentage of N bits until state $\boldsymbol{s_j}$ and r_j calculates the throughput of the frames during the previous FRI. Simulation is executed 20 times for both models and their comparison results are shown in Fig. 5.3 Compared with the throughput under NN guidance to determine the FRI value, the NN's throughput result in Fig. 5.3 is lower. The reason is that in this simulation, BER is provided by a surrogate model which considers the BER uncertainties in the real sea. BER prediction given by (4.13) tends to underestimate the BER and takes time to converge around the median BER value. However, our TS-DQN gains better throughput results than the NN-based feedback scheduling strategy. Therefore, we will apply TS-DQN for our adaptive feedback scheduling part in the subsequent chapters.

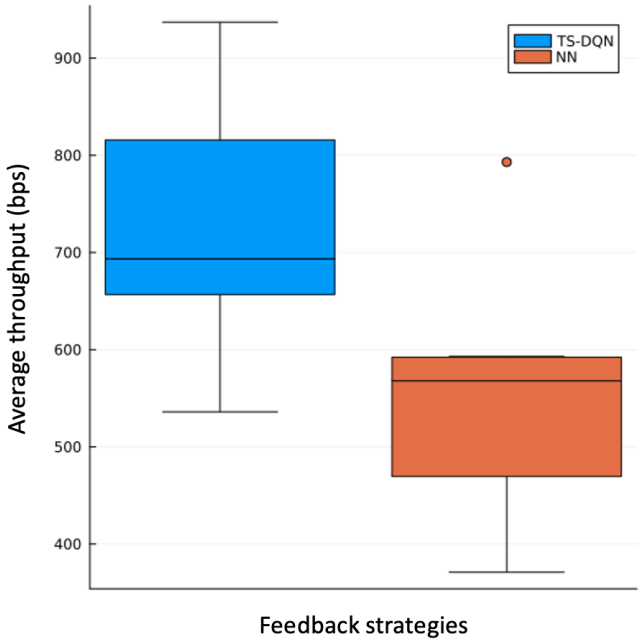


Figure 5.3: Throughput comparison given feedback scheduling under TS-DQN and NN.

5.3 Summary

We explain the details of how to employ our proposed TS-DQN in sequential decision problems, highlighting its aptitude for long-term rewards in scenarios with vast action and state spaces. We take the feedback scheduling to automatically find the right balance and optimize communication performance for AMC for example. A toy problem of transmitting frames in a one-to-one communication system is solved via simulation and the results show TS-DQN achieves better throughput than our previous NN-based model.

Chapter 6

From Theory to Practice

In the preceding chapters, we conducted extensive simulations to evaluate the performance of our algorithms and demonstrate their advantages. However, in order to effectively apply the physics-aided AMC strategy on underwater acoustic modems, there are practical challenges that need to be addressed. We present the design, implementation, and testing of a comprehensive communication system that incorporates the BER upperbound estimation model along with a consolidated feedback scheduling strategy. Specifically, we report the following contributions in this chapter:

- To ensure the selection of robust modulation schemes in adverse channel conditions, we extend our BER prediction model in Chapter 4 from point prediction to the interval predictor by integrating GPR. This integration takes into account the inherent uncertainty of the BER and enables reliable AMC in real-sea experiments with a higher frame success reception rate. The predictions obtained from the GPR model can balance the trade-off in AMC by providing a probabilistic estimate of the BERs of different MCSs. To demonstrate the advantages of our AMC algorithm, including reduced dependence on data, and improved robustness and accuracy, we conduct a comparative analysis against a purely data-driven BER estimation method

on several datasets collected from a test tank and Singapore waters.

- We introduce a more comprehensive and realistic approach for determining the timing of obtaining feedback. Real-world experiments are conducted using actual underwater acoustic communication modems, where a "test" mode is enabled on the modems. By enabling the "test" mode, known bits are transmitted to facilitate the accurate acquisition of CSI. We consider the timing of enabling or disabling the "test" mode as one of the factors related to feedback scheduling.

- Simulations may not fully capture the complexity and variability of the challenging and dynamic underwater acoustic environment. We conducted real-world experiments in a sea environment to bridge the gap between theoretical and practical applications. These experiments involved the use of actual underwater acoustic communication modems, taking into account the limitations and constraints that are not captured in simulations. While simulations often simplify the modeling of interference and noise sources, our real experiments encountered unpredictable ambient noise and potential sources of interference. In real-world experiments, we demonstrate the efficacy and reliability of our approaches and showcase their applicability and effectiveness to industry stakeholders. This can lead to potential technology transfer and adoption in practical underwater acoustic communication systems.

6.1 Problem Formulation

6.1.1 Problem Overview

We aim to transmit N bits from the TX node to the remote RX node located at a distance l within the shortest possible time. Modulation and Coding techniques are used to encode these bits onto frames for reliable communication. After modulation, the coding techniques, like the FEC, add redundant bits to the modulated frames, allowing the RX node to detect and correct errors. Due to the variability in UAC channels, it is hard to design a single modulation scheme that works well in all situations. Therefore, AMC techniques are employed, enabling tuning the modulation scheme $\boldsymbol{a}$ and coding strategy based on current channel conditions. In the experiment, we establish a DATA link between the TX node and RX node to transmit the modulated and coded frames, with the intent of providing as high a data rate as feasible. However, optimal communication performance in various environmental conditions necessitates fine-tuning this DATA link.

Successful transmission is achieved only when the TX and RX nodes employ identical modulation schemes. When the modulation schemes are determined at the TX node, a crucial task is to inform the remote RX node about the modulation information reliably before the DATA link frames are exchanged. A separate communication link, referred to as the CONTROL link, is first established. The CONTROL link exhibits robust communication albeit at a lower data rate than the DATA link, and the modulation and error correction parameters of the CONTROL link are pre-determined. The modulation

information for the DATA link is then encoded onto frames and transmitted over this CONTROL link to the remote RX node.

Performing AMC heavily relies on obtaining accurate CSI. The CSI, such as measured BER based on the number of bits corrected during FEC decoding, is acquired through feedback from the RX node. However, employing modulation schemes and coding rates blindly may lead to failed frame receptions at the RX node. Although such failures indicate that the BER exceeds a certain threshold, they hinder acquiring accurate BER for reliable AMC. To address this challenge, a "test" mode is introduced, where frames carrying known bits are transmitted over the DATA link. In this mode, the BER can be accurately computed as the transmitted frames are known, and the CSI is updated. When the "test" mode is disabled, the N unknown bits are encoded and transmitted to the RX node over the DATA link. In this case, the BER is approximated after demodulation and decoding of the frames at the RX node. All CSI, including BER measurements, are then encoded onto frames and sent back to the TX node via the CONTROL link, thereby improving the performance of AMC.

Given the possibly long propagation delay between frames exchanged over DATA and CONTROL links, tuning modulation schemes and awaiting feedback for each frame consumes time. Conversely, employing a modulation scheme across multiple frames without timely feedback can lead to suboptimal performance, resulting in a loss of received frames at the RX node and reduced throughput. This motivates the consideration of the optimal timings for tuning modulation schemes and waiting for feedback to optimize the channel throughput. Therefore, a Feedback Report Interval (FRI), which decides the

number of transmission frames over the DATA link between two consecutive feedbacks, is proposed. During the j^{th} FRI, a specific number of frames (h_j), are transmitted using the same modulation scheme $\boldsymbol{a}$ and its corresponding coding rate.

6.1.2 Mathematical Formulation

We formulate the sequential decision-making of modulation scheme $\boldsymbol{a}$ and FRI h_j as well as the subsequent interaction with the environment to receive feedback as a MDP. In this MDP, $\mathcal{A}$ is a set containing all possible modulation schemes, i.e., $\boldsymbol{a} \in \mathcal{A}$ and $\mathcal{H}$ is another set including all possible values of h_j, i.e., $h_j \in \mathcal{H}$. The action space now has a cardinality of $|\mathcal{A} \times \mathcal{H}|$. An intelligent decision-making entity, known as the agent, engages in iterative interactions with the environment to learn and optimize a policy denoted as Π. This policy guides the agent for selecting actions from action space $|\mathcal{A} \times \mathcal{H}|$ to transmit N bits within the possible shortest time. In the state space $\mathcal{S}$, a state $\boldsymbol{s_j}$ is reached upon receiving the feedback of the j^{th} FRI as shown in Fig. 6.1. It encompasses the completion ratio of N bits, knowledge related to decision-making of actions, and communication performance metrics. The state transitions from state $\boldsymbol{s_{j-1}}$ to state $\boldsymbol{s_j}$ follows a transition function Γ. Therefore, Π is a function that maps the state space to the action space, i.e., $\Pi : \mathcal{S} \rightarrow |\mathcal{A} \times \mathcal{H}|$. After successfully transmitting all N bits, the agent reaches the terminal state $\boldsymbol{s_J}$ and hence $j = 0, \cdots, J$. The round-trip frame exchange duration comprises the transmission duration of h_j frames, i.e., $h_j\tau_j$, the duration of frames containing modulation information τ_{m}, a two-way propagation delay $2\tau_{\text{pd}}$, and the duration of frames

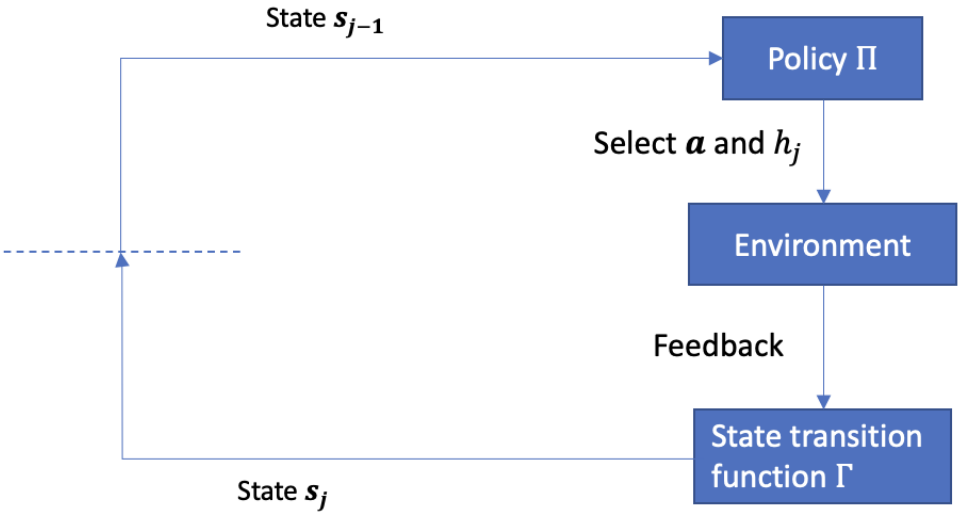


Figure 6.1: The framework of the state transition.

containing feedback τ_{fd}, as shown in Fig. 6.2.

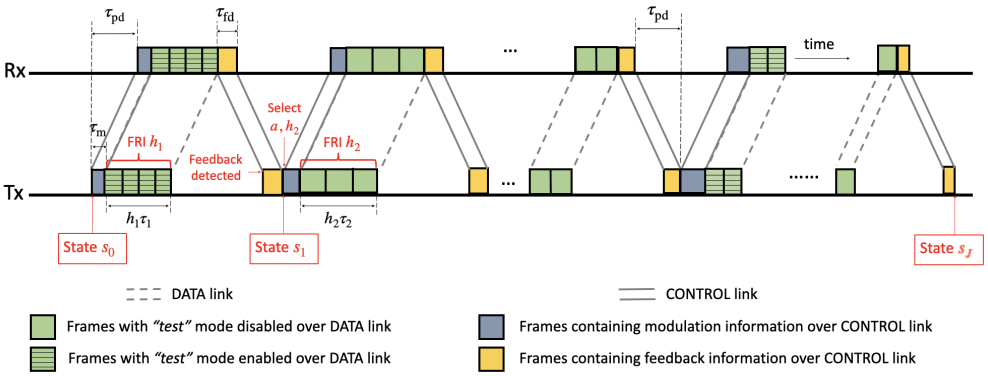


Figure 6.2: An illustration of the delays in frame exchange between the transmitter and receiver nodes.

The throughput over the transmission is the performance metric for selecting the actions in our MDP. When modulation scheme $\boldsymbol{a}$ is selected, the coding technique, such as FEC, is then applied. The FEC adds redundant bits to the transmitted data frames, facilitating error detection and correction during transmission. A set $\boldsymbol{\varrho}$ contains available FEC rates associated with their

respective affordable BER levels. With knowledge of the BER statistics $\epsilon(\boldsymbol{a})$, the agent can easily determine the optimal FEC rate $\rho(\epsilon(\boldsymbol{a}))$. Specifically, if no FEC rate in $\boldsymbol{\varrho}$ is available for correcting the BER $\epsilon(\boldsymbol{a})$, the "test" mode is enabled, in which the known bits are transmitted. Given the uncoded data rate $d(\boldsymbol{a})$ of the modulation scheme $\boldsymbol{a}$, the coded data rate is calculated as $d(\boldsymbol{a})\rho(\epsilon(\boldsymbol{a}))$. The coded data rate in a communication system is closely correlated to the throughput and thus serves as a valuable metric for enhancing channel throughput.

Knowledge of BER is crucial for calculating the coded data rate in communication systems. However, in time-varying UAC channels, measuring accurate BER can be challenging, particularly given a possibly large size of $\mathcal{A}$. Obtaining an estimation of BER $\epsilon(\boldsymbol{a})$ over the action space $\mathcal{A}$ is hence required. A heuristic BER model based on channel physics knowledge from [79] estimates the median of the BER statistics $\zeta(\boldsymbol{a};\boldsymbol{\theta_j})$. $\boldsymbol{\theta_j}$ represents the model weight parameters and are updated given the feedback of the latest CSI. However, it may not capture the worst-case performance due to the inherent uncertainty of BER measurement. To ensure robust modulation selection in adverse channel conditions, η_j is proposed to help estimate the upperbound of the unknown BER distribution. The BER upperbound $\hat{\epsilon}_j(\boldsymbol{a})$ is given by

$$\hat{\epsilon}_j(\boldsymbol{a}) = \zeta(\boldsymbol{a};\boldsymbol{\theta_j}) + \eta_j(\boldsymbol{a}). \tag{6.1}$$

Obtaining CSI through feedback from the RX node plays a vital role in facilitating accurate BER estimation and subsequent AMC. However, the

feedback process consumes time due to significant propagation delays between frames exchange. Therefore, the selection of FRI involves a tradeoff between the optimization speed of the AMC strategy and maximizing the throughput of transmitting N bits. When the transmission result of the last FRI is poor, choosing a smaller value for the next FRI enables faster convergence of the BER estimation model by updating the CSI more frequently. However, this may lead to increased latency due to propagation delays. On the other hand, selecting a larger FRI does not always guarantee improved channel throughput. In a varying UAC channel, selecting a larger FRI would mean that the model would operate far from the optimum and hence result in poorer performance. Thus, a dynamic approach is required to determine the optimal sequence of FRI that balances these two objectives.

As depicted in Fig. 6.2, the system transitions from $\boldsymbol{s_{j-1}}$ to $\boldsymbol{s_j}$ upon the completion of the j^{th} FRI. Within the state transition, the measured BER $\boldsymbol{\epsilon_j}(\boldsymbol{a})$ from the updated CSI is utilized to train (6.8). The ratio of N bits that have been transmitted, denoted by a percentage value n'_j, along with the timestamps of frames exchange is recorded. The number of frames with and without the "test" mode enabled up to state $\boldsymbol{s_j}$ are respectively tracked by k_1 and k_2. The throughput r_j of FRI h_j is calculated as the measure of $(n'_j - n'_{j-1})N$ bits transmitted over the DATA link within a time period encompassing h_j frames with a transmission duration of τ_j each, along with the duration of frames containing modulation information τ_{m}, the feedback frame duration τ_{fd}, and

a two-way propagation delay of $2\tau_{\mathrm{pd}}$, i.e.,

$$r_j = \frac{(n'_j - n'_{j-1})N}{h_j\tau_j + 2\tau_{\mathrm{pd}} + \tau_{\mathrm{fd}} + \tau_{\mathrm{m}}}. \tag{6.2}$$

The parameters n'_j, r_j, and the ratio $\bar{k} = \frac{k_1}{k_1+k_2}$ provide valuable insights into the communication performance under current policy Π. Therefore, the feedback scheduling model $\mathcal{M}(\cdot)$ takes inputs n'_j, $\bar{k}$, and r_j to predict the values of all possible $h_{j+1} \in \mathcal{H}$. The optimal h_{j+1} is determined by

$$h_{j+1} = \underset{h_{j+1}\in\mathcal{H}}{\operatorname{argmax}} \mathcal{M}\left(\left\{n'_j, \bar{k}, r_j\right\}; \boldsymbol{\omega_j} | \boldsymbol{a}, h_{j+1}\right), \tag{6.3}$$

where $\boldsymbol{\omega_j}$ denotes the parameters of $\mathcal{M}(\cdot)$, which are updated once CSI is received. The detail of state transition between $\boldsymbol{s_{j-1}}$ and $\boldsymbol{s_j}$ is

$$\begin{aligned} \boldsymbol{s_j} &= \Gamma(\boldsymbol{s_{j-1}}, \boldsymbol{a}, h_{j-1}) \\ &= \{\boldsymbol{\theta_j}, n'_j, \boldsymbol{\omega_j}, \eta_j, \bar{k}, r_j\}. \end{aligned} \tag{6.4}$$

To summarize, our objective is to choose the sequence of modulation schemes $\boldsymbol{a}$ and FRI h_j, $j = 1, \cdots, J$, where J is unknown, to transmit N bits within the shortest time and thereby maximize the throughput:

$$\begin{aligned} &\min \left(\sum_{i=1}^{J} h_i\tau_i + J(\tau_{\mathrm{fd}} + 2\tau_{\mathrm{pd}} + \tau_{\mathrm{m}}) \right) \\ &s.t. \quad n'_J \geq 1. \end{aligned} \tag{6.5}$$

6.2 BER Upperbound Predictor

In this section, we delve into the adaptation strategy of MCSs, focusing on OFDM given its prevalence in modern underwater modems. Leveraging channel physics information, we construct a heuristic BER estimation model to guide AMC strategy selection. We validate this model using various datasets and offer a reference table for BER-based FEC rate selection. Once the coded data rate for each modulation scheme is ascertained, we suggest a dynamic ϵ-greedy policy to address the exploration-exploitation dilemma in modulation scheme selection given the high-dimensional MCSs space.

6.2.1 Estimation of BER Uncertainty

The work presented in Section 4.2.1 demonstrates the capability of $\zeta(\cdot)$ for estimating the median from the time-varying BER distribution. It also illustrates the BER tends to cluster around this median value although variability is observed within the actual BER distribution. To enhance the reliability of the modulation scheme selection in view of the BER uncertainty, evaluating the upperbound in the BER distribution becomes necessary. Referring to the maximal BER for selecting the FEC rate ensures a stringent level of error correction performance and maximizes reliability. However, this approach may be overly conservative, potentially compromising data throughput. We employ the Quantile Absolute Deviation (QAD) method [130] to help estimate the BER upperbound. It entails computing the q^{th} quantile of the absolute difference between the predicted median BER given by $\zeta(\boldsymbol{a}; \boldsymbol{\theta_j})$ and the measured BER

$\boldsymbol{\epsilon_j}(\boldsymbol{a})$ from feedback. The training set $\mathcal{D}_j$ to train $\eta_j(\cdot)$ is composed of

$$\{\boldsymbol{a} \rightarrow QAD(|\boldsymbol{\epsilon_i}(\boldsymbol{a}) - \zeta(\boldsymbol{a}; \boldsymbol{\theta_i})|; q)\}, \quad i = 1, \cdots, j, \tag{6.6}$$

which is composed of attempted modulation schemes $\boldsymbol{a}$ and their corresponding QAD values during previous transmissions. $\eta_j(\cdot)$ is trained through regression analysis on $\mathcal{D}_j$, enabling it to estimate QAD for any potential input $\boldsymbol{a} \in \mathcal{A}$. The choice of q in the QAD calculation (6.6) is set to 75 which guarantees that at least 75% of transmitted frames are successfully delivered, as it encompasses the range within which 75% of the actual BER values reside. It allows for a degree of error tolerance and also considers the trade-off between error rate and data throughput.

However, performing an exhaustive search over all modulation schemes $\boldsymbol{a} \in \mathcal{A}$ and storing their corresponding QAD values is computationally impractical. However, BER uncertainty tends to be highly correlated across modulation schemes that share similar characteristics in n_{c}, n_{p}, or B. Modulation schemes with the same n_{c} value tend to exhibit comparable levels of frequency diversity, and modulation schemes with similar n_{p} values would experience similar levels of protection against intersymbol interference. Similar bandwidth B values often encounter comparable channel conditions and noise levels. The QAD estimator η_j utilizes GPR [131] to learn these correlations within $\mathcal{A}$ based on the observed QAD values up to state $\boldsymbol{s_j}$. This correlation enables QAD predictions for all potential modulation schemes $\boldsymbol{a}$, eliminating the need for exhaustive evaluation.

A GPR model includes a crucial component known as the mean function,

which establishes a prior expectation of the general trend in the predicted QAD. The mean of all the previously observed QAD values is used as the mean function, denoted as μ_{QAD}. Another essential component is the kernel function, which determines the similarity between data points and governs the smoothness and behavior of the GPR model. In our approach, we employ the Matérn kernel [132], known for its flexibility in modeling different levels of smoothness, to govern the correlation and smoothness of the estimated QAD for any two modulation schemes $\boldsymbol{a_1}$ and $\boldsymbol{a_2}$ from the training set $\mathcal{D}_j$, denoted as $K(\boldsymbol{a_1}, \boldsymbol{a_2})$. The Matérn kernel has two key parameters: the length scale ℓ and the smoothness parameter ν. The length scale parameter ℓ determines the range over which data points influence each other. A small ℓ confines the influence of a modulation scheme $\boldsymbol{a}$ to a narrow $(n_{\text{c}}, n_{\text{p}}, B)$ space, resulting in rapid changes in the GPR function over short distances. Conversely, a large length scale allows a modulation scheme $\boldsymbol{a}$ to have a significant influence on other schemes that are farther apart, even over long distances. The smoothness parameter ν controls the flexibility of the Matérn kernel in capturing complex patterns. Higher values of ν lead to smoother functions, while lower values introduce more roughness and allow for intricate variations in the modeled functions. In our model, we choose $\ell = 0.3$, approximated using the distance between the nearest neighbor modulation schemes in $\mathcal{A}$. To determine an appropriate value of ν, a range of commonly used values, namely $\{0.5, 1.5, 2.5\}$, is visualized on datasets collected from a tank and Singapore water which we list in subsection 6.2.2. $\nu = 1.5$ is finally selected as it achieves superior performance by striking a suitable balance between demonstrating regularity in BER and allowing for fluctuations between

neighboring modulation schemes.

The QAD estimator $\eta_j(\cdot)$ predicts QAD for $\boldsymbol{a} \in \mathcal{A}$ at state $\boldsymbol{s_j}$ follows

$$\eta_j(\boldsymbol{a}) \sim \mathcal{GP}(\mu_{\mathrm{QAD}}, K(\boldsymbol{a_1}, \boldsymbol{a_2})), \tag{6.7}$$

where $\boldsymbol{a_1}, \boldsymbol{a_2} \in \mathcal{D}_j$. The BER upperbound $\hat{\epsilon}_j(\boldsymbol{a})$ is given by

$$\hat{\epsilon}_j(\boldsymbol{a}) = \zeta(\boldsymbol{a}; \boldsymbol{\theta_j}) + \eta_j(\boldsymbol{a}). \tag{6.8}$$

6.2.2 Data Sets

Two datasets containing the measured BER statistics for tuning n_{c}, n_{p}, and B using Subnero [133] M25M modems. The Subnero M25M modem operating bandwidth is up to 12 kHz, i.e., from 20 to 32 kHz. The details of these datasets are provided below.

- A data set collected from an easily accessible tank of the Acoustic Research Laboratory shown in Fig. 6.3. The transmission distance l between the TX and RX nodes was about 1.5 m, and the depth of TX and RX modems was 1.5 m. The BER was measured for $(n_{\mathrm{c}}, n_{\mathrm{p}}, B)$, where n_{c} was set to different values from the set $\{128, 256, 512, 1024, 2048, 4096, 8192\}$ and n_{p} ranged from 0 to 8192, B was set to different values from $\{2.4, 4.8, 7.2, 9.6, 12, 14.4, 16.8\}$ kHz.

- A data set, denoted by SEADATA1, was collected by colleagues from the Acoustic Research Laboratory. The experiment to collect SEADATA1 was in Singapore waters. Fig. 6.4 is adapted from [134] to show the

experiment setup to collect SEADATA1. The transmission range between the TX and RX nodes, i.e., the node B and node C in Fig. 6.4 was about 600 m, and the water depth was between 10 and 20 m. The BER was measured for $(n_{\mathrm{c}}, n_{\mathrm{p}}, B)$, where n_{c} was set to different values from the set $\{64, 128, 256, 512, 1024, 2048\}$, n_{p} ranged from 0 to 2046, B was set to different values from $\{4.8, 7.2, 9.6, 10.8\}$ kHz.

These two datasets are separated into a train set and a test set by a ratio $7:3$ for each iteration of training BER upperbound estimation model in (6.8).

6.2.3 BER Upperbound Estimation Model Validation

6.2.3.1 Validation on a Large Data Set

The improved BER estimation model is validated on the data set SEADATA1. To assess the model, SEADATA1 was split into a training set (SEATRAIN1) and a testing set (SEATEST1) using a $7:3$ ratio.

Fig. 6.5 compares the measured BER with the estimated BER obtained from the proposed model (6.8) trained on SEATRAIN1. To reduce redundancy, the results are presented for each n_{c}, with n_{p} binned into groups of size 128 due to similar BER values between adjacent n_{p} values. The proposed model demonstrates better responsiveness to changes in n_{p} and greater accuracy in capturing BER uncertainties compared to the findings reported in [79].

6.2.3.2 Validation on a Small Data Set

Determining the optimal modulation scheme $\boldsymbol{a}$ with minimal transmission is a critical performance indicator in real-world experiments for improving transmission time efficiency. The ability of a BER model to accurately estimate

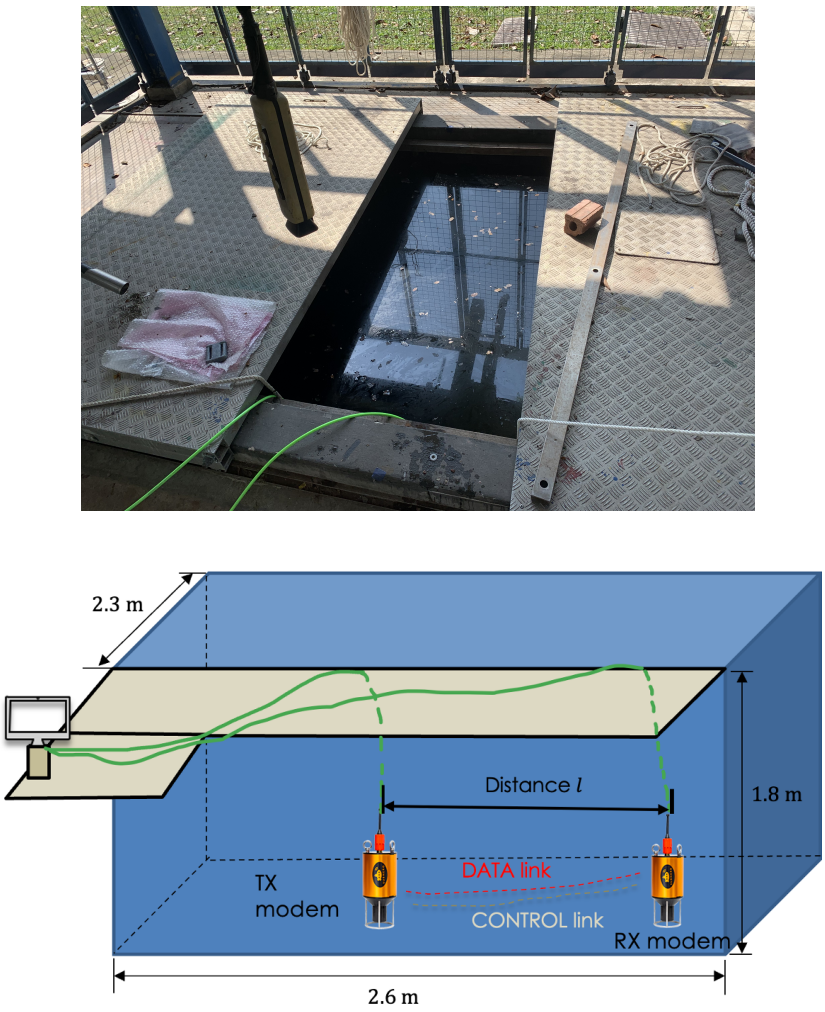


Figure 6.3: Test tank and deployment of modems in the tank.

the BER with limited transmission and CSI feedback is a significant advantage. Our proposed BER estimation model incorporates knowledge of channel physics, which is supposed to enhance its ability to make reliable and accurate predictions with fewer training data or transmissions than pure data-driven methods. To demonstrate its superiority, five pairs of $(n_{\mathrm{c}}, n_{\mathrm{p}}, B)$ and its corresponding measured BER sampled from the SEATRAIN1 data set randomly are utilized as the training set.

The GPR is employed as a purely data-driven comparison method, denoted

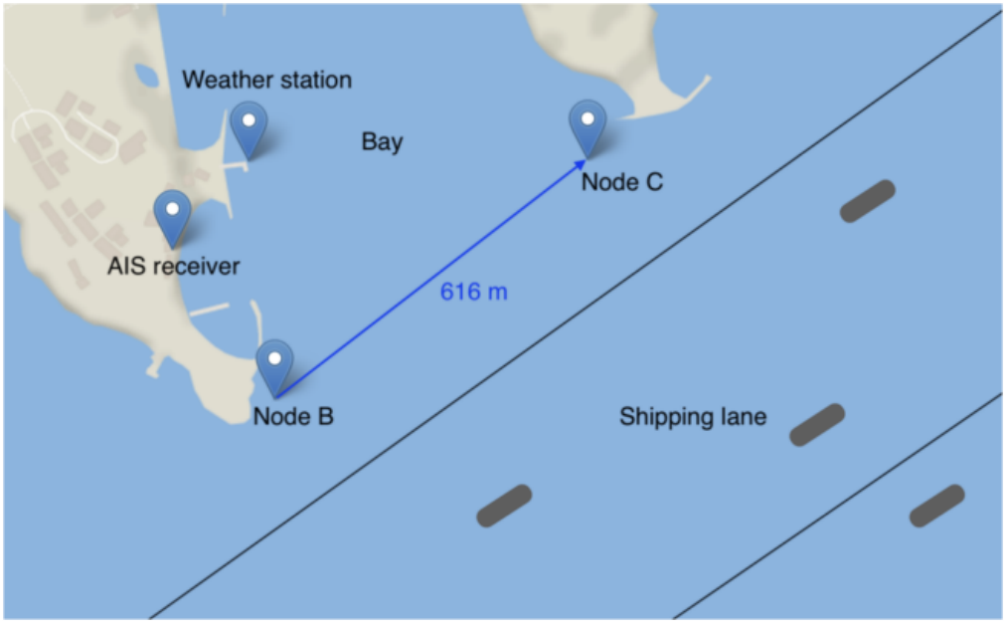


Figure 6.4: Experiment setup for collecting SEADATA1.

Notes: This figure is adapted from [134] to show the experiment setup to collect SEADATA1.

as $\mathcal{GP}'$ to easily distinguish it from the previous QAD estimator $\mathcal{GP}$. The prior mean function for $\mathcal{GP}'$, μ_ϵ, is the mean of the five samples. Assuming QAD and BER estimation share identical correlation functions over $\mathcal{A}$, we employ the same Matérn kernel in the QAD estimator $\mathcal{GP}$ for $\mathcal{GP}'$. This purely data-driven GPR method assumes the estimated BER $\epsilon_{\text{gp}}(\boldsymbol{a})$ for $\boldsymbol{a} \in \mathcal{A}$ follows the $\mathcal{GP}'$ after trained on the five samples, i.e.,

$$\epsilon_{\text{gp}}(\boldsymbol{a}) \sim \mathcal{GP}'(\mu_\epsilon, K(\boldsymbol{a_1}, \boldsymbol{a_2})), \tag{6.9}$$

where $\boldsymbol{a_1}, \boldsymbol{a_2}$ are any two modulation schemes from the five samples. The comparison between our physics-informed model $\zeta(\cdot) + \eta_j(\cdot)$ and this purely data-driven GPR model $\mathcal{GP}'$ is performed on the SEATEST1 set. Table 6.1 and

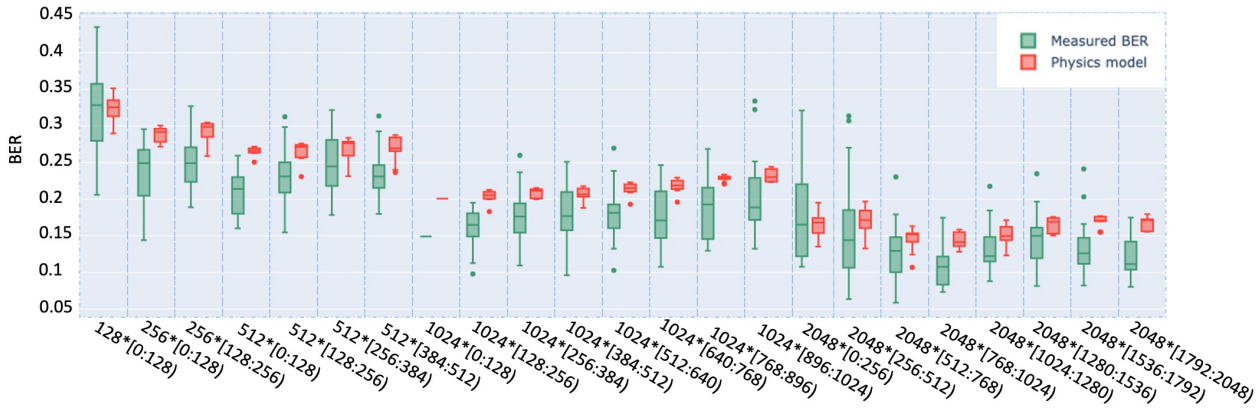


Figure 6.5: Comparison of the measured BER in SEATEST1 and the BER upperbound estimation.

Notes: The x-axis labels correspond to the values of n_{c} and a range of n_{p}, for example, $128 * [0 : 128) \rightarrow \{n_{\mathrm{c}} = 128, n_{\mathrm{p}} \in [0 : 128)\}$. For each $n_{\mathrm{c}} * n_{\mathrm{p}}$ pairs, there are two boxplots, the left one is the measured BER, and the right one represents the estimated BER upperbound by our physics-informed model given SEATEST1.

Fig. 6.6 illustrates an example of the training set and the comparison results for one five-sample set.

TABLE 6.1: An Example of a 5 Data Points Training Set.

Row	Number of Subcarriers n_{c}	Cyclic Prefix Length n_{p}	Bandwidth B (kHz)	BER
1	2048	47	10.8	0.177
2	64	59	10.8	0.334
3	1024	211	10.8	0.173
4	2048	168	7.2	0.150
5	1024	28	7.2	0.094

The results presented in Fig. 6.6 demonstrate the superior performance of our proposed physics-informed model when channel knowledge is limited, as indicated by the red boxplots. Compared to the yellow boxplots generated by $\mathcal{GP}'$, our model exhibits a better ability to identify the potential upperbound of the time-varying BER distribution. In contrast, the yellow boxplots generated by the purely data-driven GPR model $\mathcal{GP}'$ tend to underestimate the BER and are clustered around the lowerbound. This tendency leads to an overconfident

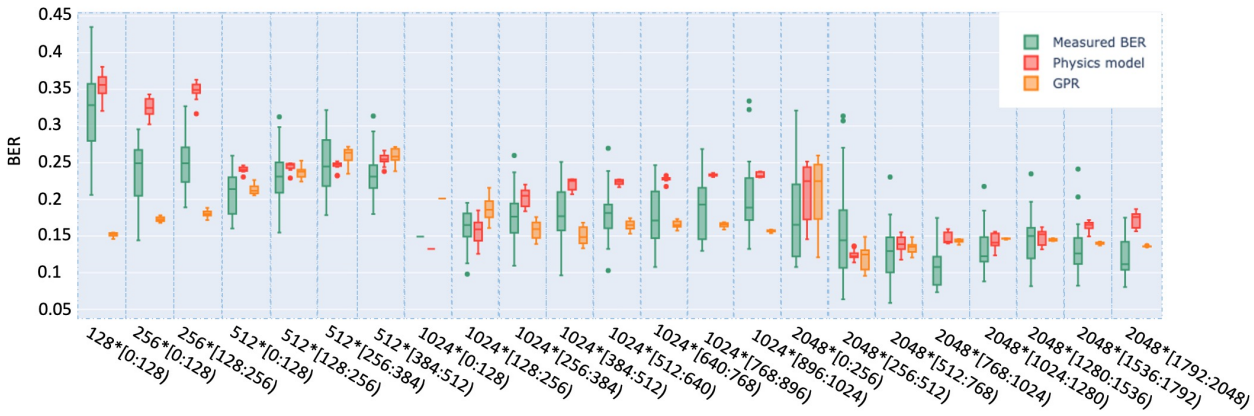


Figure 6.6: Comparison of the measured BER, the estimated BER upperbound by our physics-informed model, and estimated BER via a pure data-driven GPR model for 5 samples from SEATEST1.

Notes: The x-axis labels correspond to the values of n_c and a range of n_p, for example, $128 * [0 : 128) \rightarrow \{n_c = 128, n_p \in [0 : 128)\}$. For each $n_c * n_p$ pairs, there are three boxplots, the left one is the measured BER, the middle one represents the estimated BER upperbound by our physics-informed model, and the right one is estimated via a pure data-driven GPR model of 5 samples from SEATEST1.

selection of the FEC rate, resulting in a higher probability of frame failures. Although the BER estimates provided by our model may be higher than the measured upperbound for modulation schemes $\boldsymbol{a}$, this conservative approach allows for the selection of a more robust FEC rate. Consequently, a better frame success rate can be achieved while still maintaining an acceptable compromise in terms of data rate.

6.2.4 Forward Error Correction

Given the estimated BER upperbound, we apply the low-density parity-check (LDPC) code [135] as the FEC technique. To choose an appropriate LDPC rate, Table 6.2 provided by Subnero is consulted. This table is obtained via simulations. In the simulation, three different block size frames with $18, 432, 1450$ bytes are embedded errors manually. They then employ 6 different

LDPC rates from $\{\frac{2}{3}, \frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \frac{1}{5}, \frac{1}{6}\}$ to decode that three different block size frames with error included and tested the maximal LDPC rate for a certain BER level with 90% frame success rate. For BER values less than 0.18, they showed the 6 LDPC rates mentioned previously were capable of correcting the errors. When none of the BER ranges specified in Table 6.2 are met, we enable the *test* mode, and known bits are sent from the TX node to the RX node.

TABLE 6.2: LDPC Rate Selection Criterion

BER Estimation	**LDPC Rate**
$\hat{\epsilon}(\boldsymbol{a}) = 0$	1
$\hat{\epsilon}(\boldsymbol{a}) < 0.03$	$\frac{2}{3}$
$\hat{\epsilon}(\boldsymbol{a}) < 0.07$	$\frac{1}{2}$
$\hat{\epsilon}(\boldsymbol{a}) < 0.12$	$\frac{1}{3}$
$\hat{\epsilon}(\boldsymbol{a}) < 0.15$	$\frac{1}{4}$
$\hat{\epsilon}(\boldsymbol{a}) < 0.18$	$\frac{1}{6}$
Otherwise	Enable "test" mode

Notes: This table is provided by Subnero. Colleagues in Subnero operated simulations where three different block size frames with $18, 432, 1450$ bytes are embedded errors manually. 6 different LDPC rates from $\{\frac{2}{3}, \frac{1}{2}, \frac{1}{3}, \frac{1}{4}, \frac{1}{5}, \frac{1}{6}\}$ to decode that three different block size frames with error included and tested the maximal LDPC rate for a certain BER level with 90% frame success rate. When none of the BER ranges specified in Table 6.2 are met, the *test* mode is enabled, and known bits are sent from the TX node to the RX node.

6.2.5 Exploration & Exploitation in AMC

During the process of sequential decision-making on modulation schemes $\boldsymbol{a}$ while simultaneously collecting feedback to gain insights into the channel behavior, the agent faces the dilemma of whether to exploit the existing knowledge by choosing $\boldsymbol{a}$ or to explore new, untested schemes to enhance its

understanding of the channel dynamics and possibly achieve higher rewards. To balance this trade-off, we employ a dynamic ε-greedy algorithm to select the modulation scheme $\boldsymbol{a}$. The classic ε-greedy policy is expressed as

$$\boldsymbol{a} = \begin{cases} \underset{a\in\mathcal{A},\rho(\epsilon(\boldsymbol{a}))\in\boldsymbol{\varrho}}{\operatorname{argmax}} \; d(\boldsymbol{a})\rho(\hat{\epsilon}_j(\boldsymbol{a})), & \text{with probability } 1-\varepsilon \\ \text{Random}, & \text{with probability } \varepsilon \end{cases}, \tag{6.10}$$

where ε is for exploring $\boldsymbol{a}$ randomly to avoid being trapped in a local optimum. In our dynamic ε-greedy algorithm strategy, ε gradually decayed by

$$\varepsilon = \begin{cases} \varepsilon_{\text{decay}} \times \varepsilon, & \text{if } \varepsilon > \varepsilon_{\min} \\ \varepsilon_{\min}, & \text{if } \varepsilon \leq \varepsilon_{\min} \end{cases}, \tag{6.11}$$

where $\varepsilon_{\text{decay}}$ is the decay coefficient to control the degree of randomness and $\varepsilon_{\min}$ is the minimum value of the random factor. During the initial phase of learning, a larger ε value is applied to encourage the agent to explore untried modulation schemes with a higher probability. As the number of transmission frames increases, the agent tends to rely and exploit more on the accumulated knowledge base, which improves the learning efficiency.

6.3 AMC with TS-DQN-based Feedback Scheduling

When the MCS $\boldsymbol{a}$ and its corresponding LDPC rate, the agent starts to select the FRI h_{j+1}. TS-DQN introduced in Chapter 5 is employed. As we demonstrated in Section 6.1, the parameters n'_j, r_j, and the ratio $\bar{k} = \frac{k_1}{k_1+k_2}$ provide valuable insights into the communication performance under current

policy Π. The parameter n'_j represents the ratio of completed bits at state $\boldsymbol{s_j}$, r_j is the throughput of the last FRI h_j and computed by (7.1), and $\bar{k} = \frac{k_1}{k_1+k_2}$ calculate the ratio between the number of frames with "test" mode enabled or disabled till state $\boldsymbol{s_j}$. Therefore, when determining the next FRI h_{j+1}, the estimated $\hat{Q}$ in (7.8) is updated to

$$\hat{Q}(\boldsymbol{s_j}, \boldsymbol{a}, h_{j+1}) = \mathcal{M}\left(\left\{n'_j, \bar{k}, r_j\right\}; \boldsymbol{\omega_j} | \boldsymbol{a}, h_{j+1}\right). \tag{6.12}$$

As shown in Fig. 5.1, in the tree search selection step, h_{j+1} is selected based on (7.8), and the memory replay and TS-DQN training steps are aligned with the elaboration in Section 5.1.

6.4 Experiments and Results

6.4.1 Experimental Setup

In our experimental setup, the TX and RX nodes are WNC-M25MSS3 modems [136] from Subnero. These modems support two types of links: CONTROL and DATA. A "type" parameter distinguishes frames transmitted over these links. A value of 1 indicates transmission over the CONTROL link, while a value of 2 represents transmission over the DATA link. The use of different "type" parameter values enables the RX node to make appropriate decoding decisions.

The CONTROL link employs Frequency-Hopping Binary Frequency Shift Keying (FHBFSK) as the modulation technique and LDPC code as the FEC method. In different environments, the LDPC code rate and power level of

the CONTROL link are pre-tuned and remain static. On the DATA link, the LDPC code is used as FEC, and OFDM serves as the modulation technique. For each FRI, the number of subcarriers, cyclic prefix length, and bandwidth in OFDM are tuned adaptively. The "test" mode of the DATA link in the WNC-M25MSS3 modems can be enabled or disabled using the *true* or *false* command, respectively.

Frames transmitted over the CONTROL link serve two purposes. The first purpose involves sending frames containing modems' modulation and setup information at the beginning of each FRI. Specifically, this information encompasses the number of subcarriers, cyclic prefix length, and bandwidth in the OFDM system, the LDPC code rate, "test" command, and FRI value. Secondly, the CSI feedback from the RX modem to the TX modem is sent over the CONTROL link. The feedback frame contains measured BER statistics, namely the median and 75^{th} QAD of the measured BER values, along with the number of bits transmitted during one FRI.

To ensure sufficient reception of frames over the DATA link without compromising throughput unnecessarily, we should consider an appropriate wait duration on the RX modem after the modulation setup. This wait duration, denoted as τ_{wait}, is calculated as $h_j\tau_j$, where τ_j represents the transmission duration of each frame in the j^{th} FRI. Upon receiving the frame with modems' modulation and setup information over the CONTROL link, the RX modem triggers the waiting phase for the upcoming h_j frames over the DATA link, lasting for $\tau_{\text{wait}} = h_j\tau_j$.

The operational procedures on the TX modem and RX modem are detailed

in Algorithm 3 and Algorithm 4, respectively.

Algorithm 3 Operations on TX Modem over Transmission

INITIALIZATION: state $\boldsymbol{s_0} = \{\boldsymbol{\theta_0}, \boldsymbol{\eta_0}, \boldsymbol{\omega_0}, n'_0 = 0, k = 0.5\}$ where $\boldsymbol{\theta_0}$ is randomized, $\boldsymbol{\omega_0}$ is pre-trained in Section 6.4.2 and $\eta_0(\boldsymbol{a}) = 0.18$ for $\boldsymbol{a} \in \mathcal{A}$.
for $i = 0, 1, \cdots, J$ **do**
  Estimate BER $\hat{\epsilon}_i(\boldsymbol{a})$ for $\boldsymbol{a} \in \mathcal{A}$ using (6.8).
  Determine FEC rate $\rho(\hat{\epsilon}_i(\boldsymbol{a}))$ for $\boldsymbol{a} \in \mathcal{A}$ using Table 6.2.
  Select $\boldsymbol{a}$ using (6.10) and h_{i+1} using (7.8).
  Determine whether to enable or disable "test" mode.
  Transmit frame carrying modulation and "test" information over CONTROL link.
  Transmit h_{i+1} frames over DATA link.
  Detect and decode feedback frames from the RX modem over the CONTROL link.
  perform state transition $\boldsymbol{s_i} \rightarrow \boldsymbol{s_{i+1}}$.
  if $n'_{i+1} \geq 1$ **then**
    Stop transmission.
  end if
end for

Algorithm 4 Operations on RX Modem over Transmission

while receive frame with "type" $= 1$ **do**
  Decode and Modulate.
  if "test" command is *true* **then**
    Enable "test" mode.
  else
    Disable "test" mode.
  end if
  Calculate τ_{wait}.
  while in τ_{wait} **do**
    if receive frame with "type" $= 2$ **then**
      Decode and store CSI.
    end if
  end while
  Send feedback frame to TX modem.
end while

6.4.2 Pre-training of Feedback Model

In the action space $[\mathcal{A} \times \mathcal{H}]$ of MDP, the tunable OFDM parameters offer an extensive range of possible values, which results in a very large action space.

Real-time training of the model $\mathcal{M}(\cdot)$ in the sea trial setting becomes challenging and time-consuming. To address this, we opt to pre-train $\mathcal{M}(\cdot)$ in the Acoustic Research Laboratory's easily accessible tank and deploy as depicted in Fig. 6.3. This controlled and safe environment allows us to learn the initial features and patterns of model $\mathcal{M}(\cdot)$, denoted by $\hat{\boldsymbol{\omega}}$. For the CONTROL link, LDPC with a $\frac{1}{3}$ rate FEC is employed, and both DATA and CONTROL links have a power level set to 155 dB re 1 μPa on TX and RX modems. Each pre-training iteration follows operational procedures aligned with Algorithm 3 and Algorithm 4 and terminates until N bits have been transmitted. The learned parameters $\boldsymbol{\omega_J}$ at the terminal state $\boldsymbol{s_J}$ of each iteration are stored and utilized as the initial $\boldsymbol{\omega_0}$ for the subsequent pre-training iteration. Given the propagation delay is negligible in the tank, we assume the distances l between the TX modem and RX modem as 1, 2, or 3 km, and accordingly, we include $\tau_{\text{pd}} = \frac{2}{3}, \frac{4}{3}, 2$ s into each FRI during the pre-training of $\hat{\boldsymbol{\omega}}$. After conducting 100 pre-training iterations, the trained $\hat{\boldsymbol{\omega}}$ is utilized as the initial value of $\boldsymbol{\omega_0}$ in subsequent experiments.

6.4.3 Tank Experiment

Before the sea trial, we first test our algorithm in the same tank shown in Fig. 6.3. The same two WNC-M25MSS3 modems from Subnero are deployed in four different positions as shown in Fig. 6.7. For the CONTROL link, the FEC employed LDPC with $\frac{1}{3}$ rate. The power level was set to 155 dB re 1 μPa on TX and RX modems on both CONTROL and DATA links. We maintain the consistent use of the AMC strategy in policy Π within the MDP. Meanwhile, we compare our feedback scheduling algorithm TS-DQN with other strategies:

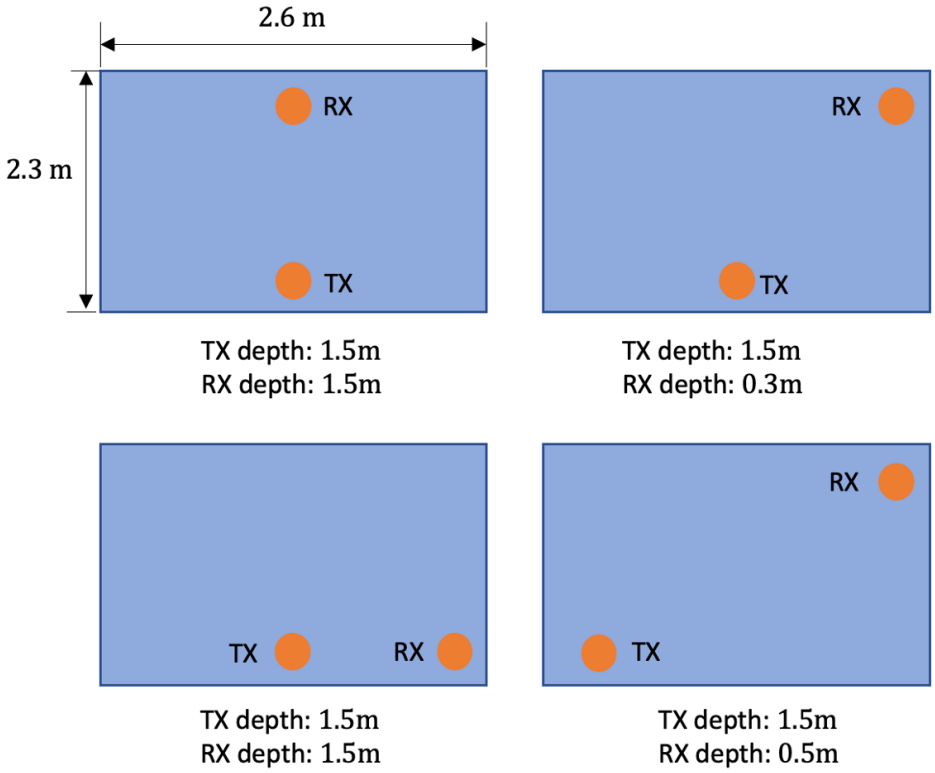


Figure 6.7: 4 different deployments in the test tank.

Random strategy, where h is randomly picked between 1 and 20. The Fixed strategy where $h = 5, 10, 15, 20$ is fixed a prior, and a progressive increase policy, named Time-varying, that FRI increases with n'_j, the ratio of bits transmitted. Each transmission run employing different feedback strategies is terminated until $N = 100000$ bits have been transmitted. As the size of the tank, the propagation delays τ_{pd} are negligible and the average results are demonstrated in Fig. 6.8. To observe the impact of propagation delays, we assume a distance l of 3 km, incorporating a $\tau_{\text{pd}} = 2$ s, and present the throughput results in Fig.6.9. Notably, when propagation delays are negligible, a smaller FRI value allows the policy Π to converge faster by acquiring CSI feedback more

frequently. Conversely, under the Fixed FRI policies, increasing the fixed FRI value gradually decreases the median throughput. Our proposed TS-DQN algorithm adaptively selects a small FRI value for quick feedback updates during AMC strategy optimization. As the confidence in the AMC strategy increases along with the transmission, both TS-DQN and Time-varying strategies raise the FRI value to save time on propagation while still ensuring close-to-optimal AMC performance. However, the Time-varying strategy's slower FRI adjustment speed results in lower throughput. Hence, when propagation delays exist, the advantages of TS-DQN become prominent, saving time on propagation and feedback delays when the AMC strategy performs well. Fig.6.9 also shows the Fixed policy with a small value like 5 and the Time-varying policy have lower throughput results, as they waste too much time on obtaining feedback when the AMC strategy is already performing well. Conversely, a large fixed FRI value, like 20, leads to slow convergence when the agent has limited channel knowledge, consequently hindering throughput.

6.4.4 Sea Trial

Sea trial is conducted at the marina as shown in Fig. 6.10. The setup of our one-to-one communication system is illustrated in Fig. 6.11. Two WNC-M25MSS3 modems act as the TX node and the RX node. The TX modem is controlled from a nearby ground station via a laptop. The RX modem is operated using a laptop 100 m away. The RX laptop was only responsible for receiving modulation setup information, guiding the RX modem's modulating and receiving frames, and sending the feedback frames. The depths of TX and

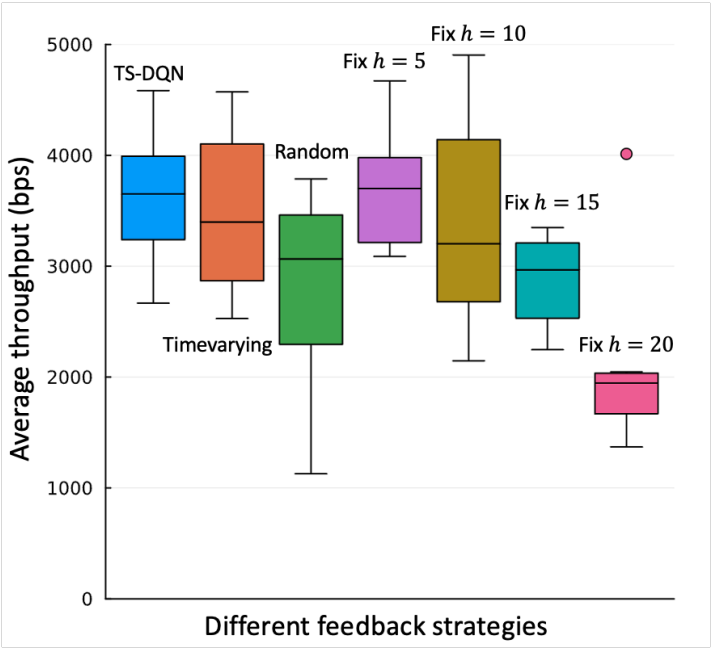


Figure 6.8: Tank throughput comparison before propagation delays added.

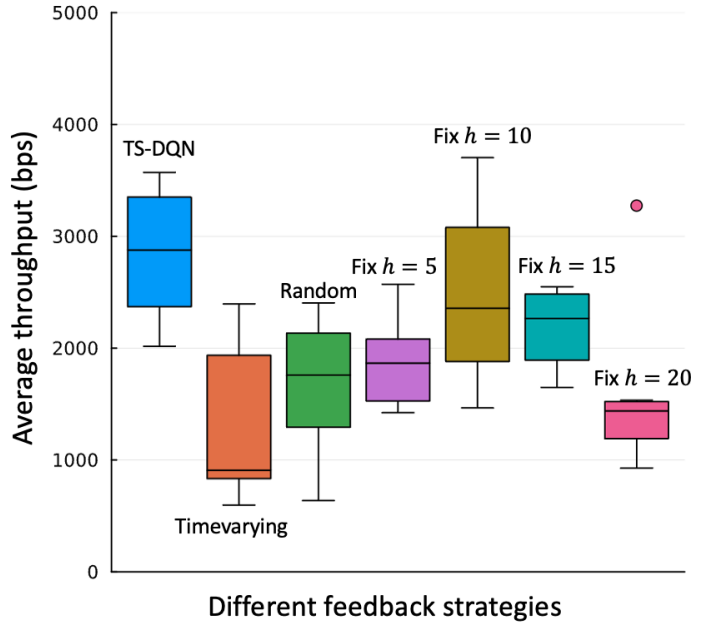


Figure 6.9: Tank throughput comparison after propagation delays added.

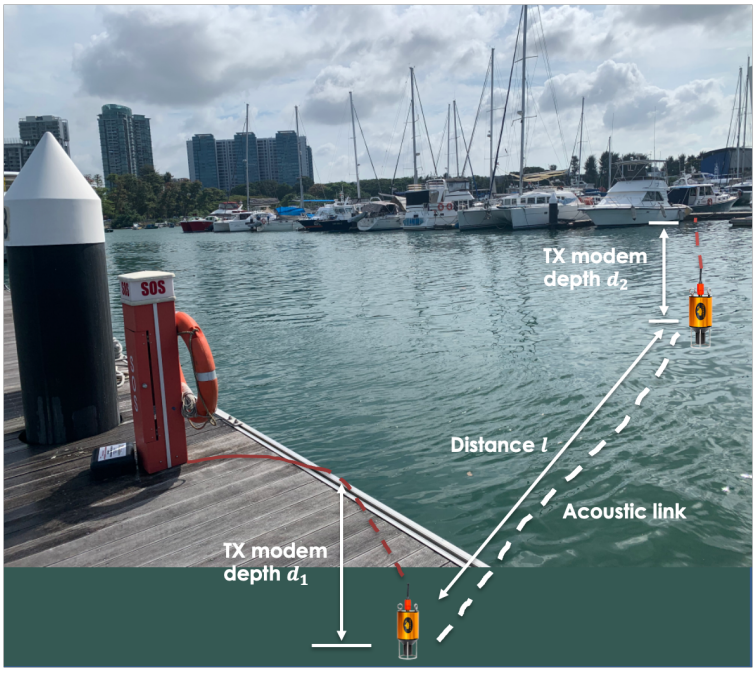


Figure 6.10: Test environment of sea trial.

RX are set as $d_1 = 5$ m and $d_2 = 3$ m, respectively. The transmission distance $l = 100$ m. In our experiment, both the TX and RX modems utilize LDPC with a $\frac{1}{6}$ rate for FEC on the CONTROL link. The TX modem's power level is set to 175 dB re 1 μPa, and the RX modem's power level is set to 185 dB re 1 μPa. Each run is terminated after transmitting $N = 100,000$ bits. The algorithms on both the TX and RX modems are aligned with Algorithm 3 and 4, respectively. The throughput results for each feedback strategy, i.e., TS-DQN, Random, Fixed, and Time-varying, is the average of 20 transmissions. Fig.6.12 presents the results with the actual delay propagation ($\tau_{\text{pd}} \approx 0.067$ s) at the marina, and Fig.6.13 shows the throughput when we assume the distance $l = 3$ km.

The timing diagram of a series of transmissions from the sea trial is depicted in Fig. 6.14 where frames labeled "test frames" indicate the "test" mode is

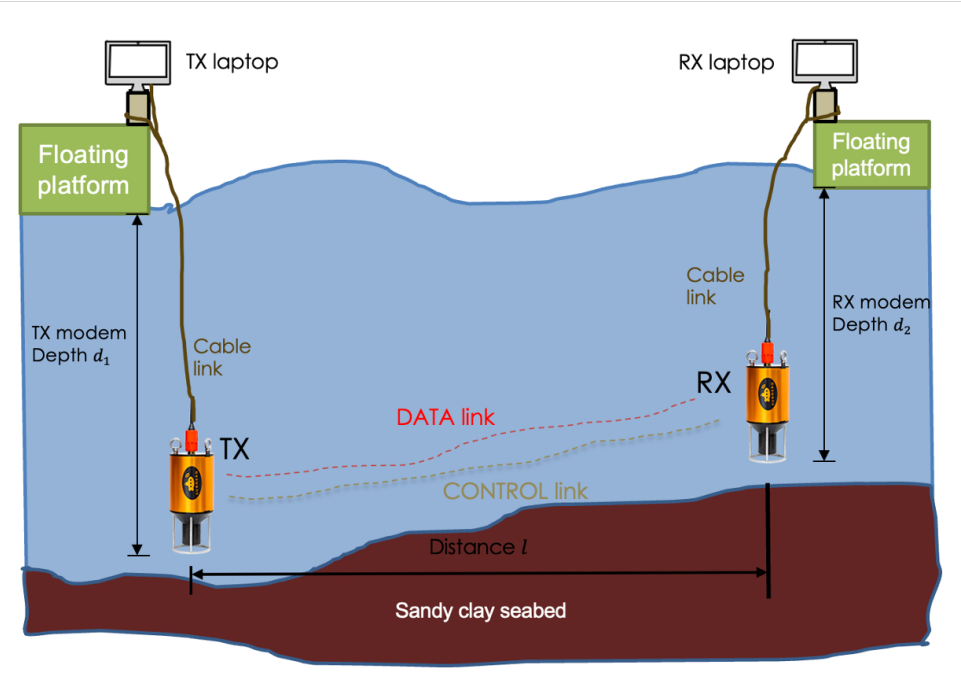


Figure 6.11: Sea trial deployment.

on and frames labeled "data frames" indicate the "test" mode is switched off. The sea trial throughput results demonstrate a decrease when compared with the tank throughput results presented in Fig.6.8 and Fig.6.9. This decrease is attributed to the increased noise from nearby shipping and construction in the marina environment. Under the Fixed policy, the throughput shows a decreasing trend as the FRI value increases, and the advantage of our proposed TS-DQN is not notably evident. This is due to the satisfactory convergence speed of the AMC strategy under the Time-varying and Fixed policies. When considering propagation delays for a distance of 3 km, the impact of the propagation delay increased. Our TS-DQN algorithm outperforms, as it can dynamically determine the FRI value based on the channel conditions. Specifically, TS-DQN tends to select a smaller FRI when there is limited channel knowledge or when channel

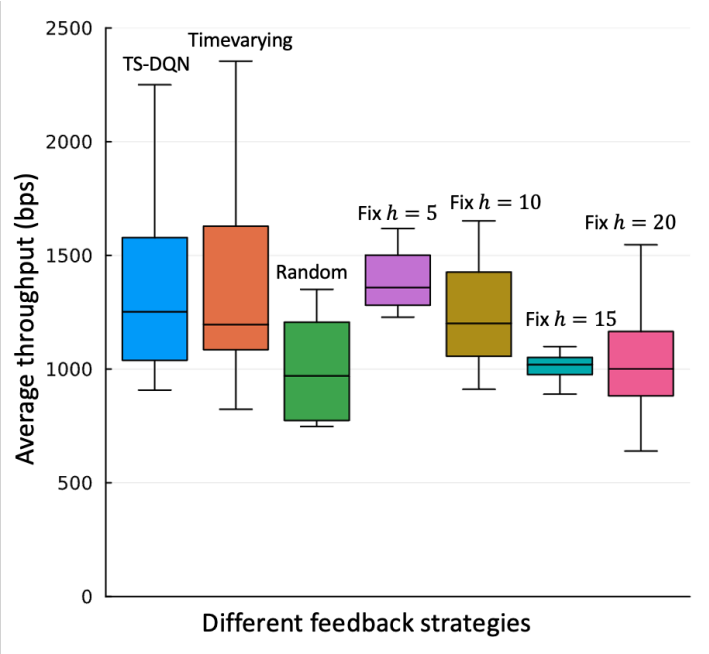


Figure 6.12: Sea trial throughput comparison before propagation delays added.

conditions change, while it chooses a larger FRI when the AMC strategy is optimized or operates in a stable channel condition.

6.5 Simulation and Results

We have verified our algorithm in a controlled tank and open sea environments. We aim to verify our algorithm in more different underwater environments via simulations. The surrogate model to represent the DATA link of different UAC environments is built based on [129], where the Pekeris ray model with red Gaussian noise is employed. There are some parameters in the Pekeris ray model we modify, such as the bathymetry with a constant depth, and the ambient noise model with variance σ^2. We fix the isovelocity sound speed profile with sound speed $c = 1540$ m/s, a flat sea surface, and a sandy clay seabed. We choose the variance in the red Gaussian noise to be $1e^6$ which

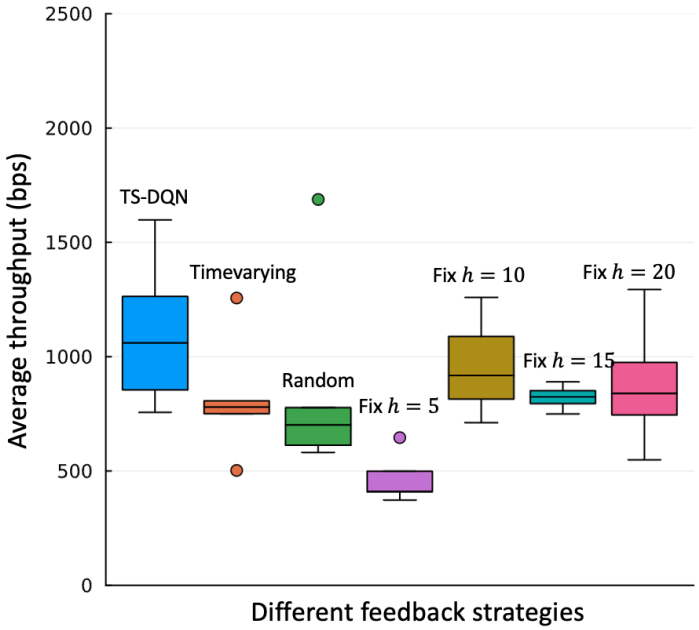


Figure 6.13: Sea trial throughput comparison after propagation delays added.

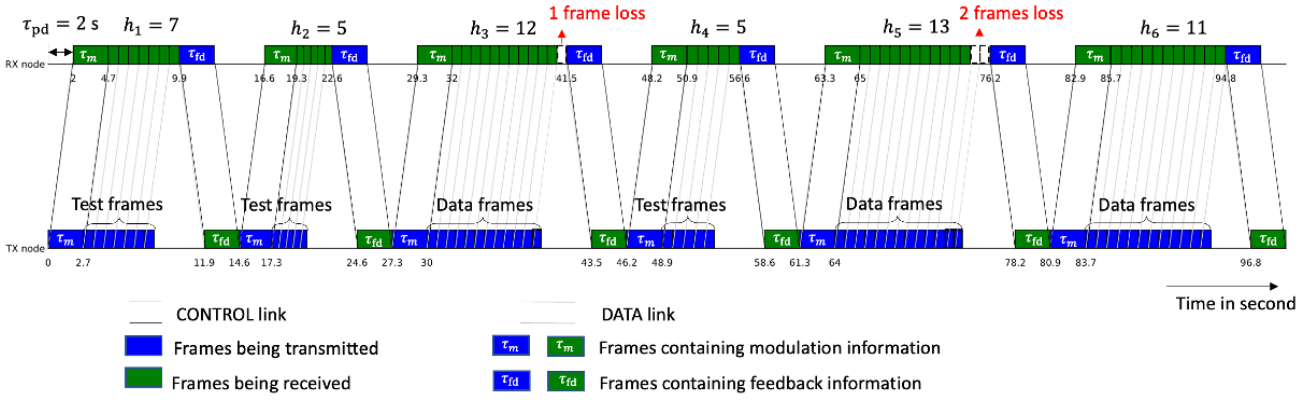


Figure 6.14: An illustration of the timestamps in frame exchange between the TX and RX modems in the sea trial.

is aligned with Singapore waters. The transmission distance l and the depth of the TX node and RX node d_1 and d_2 for different simulation surrogate models are listed in Table. 6.3. The number of bits per PSK symbol on each subcarrier m is set to 2 in (4.8).

In the simulation, the TX and RX nodes are running on the same machine

and the measured BER is provided directly by the surrogate model. Therefore, the duration of frames containing modulation information and feedback no longer existed. To be consistent with the practical setup in experiments, we assume the $\tau_{\mathrm{m}} = \tau_{\mathrm{fd}} = 2.7$ s. The propagation delay τ_{pd} is determined by the distance l, i.e., $\tau_{\mathrm{pd}} = \frac{l}{1540}$ s. The FEC selection rule is identical to Table 6.2. To better simulate the real environment, if the measured BER given by the environment surrogate model is less than the BER limit of each LDPC rate, the frame has a very high probability, 90%, to be successfully received. Meanwhile, a frame has a probability of 10% to be received successfully even when its measured BER is larger than the given BER limit.

We compare our feedback strategy TS-DQN with Random, Fixed, Time-varying, and DNN (our previous work presented in [79]) strategies. The simulation stops when $N = 100,000$ bits have been transmitted. The possible number of sub-carrier values are $n_{\mathrm{c}} = 64, 128, 256, 512, 1024, 2048, 4096, 8192$, the value n_{p} can be selected from 0 to 8192, the possible occupy ratio of the bandwidth 24 KHz is $0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9$. For different feedback strategies, transmission is executed 20 times given a different UAC surrogate model.

TABLE 6.3: UAC Surrogate Model Parameters

Surrogate Model	Constant Depth of Bathymetry	TX node depth	Rx node depth	Distance
1	50 m	25 m	25 m	3000 m
2	50 m	25 m	25 m	2000 m
3	10 m	5 m	5 m	100 m

For different surrogate models listed in Table 6.3, the throughput results of different feedback strategies are presented in Fig. 6.15, Fig. 6.16, and Fig. 6.17.

The Random strategy yields significantly lower throughput compared to the proposed TS-DQN strategy. For the Fixed strategy, the median throughput initially increases as the FRI increases but gradually decreases when FRI exceeds a certain point for $l = 2000$ m or 3000 m. However, for $l = 100$ m, the throughput continues to increase as FRI increases. This is due to the surrogate model being sensitive to the distance l between the TX node and RX node. The channel becomes more challenging, i.e., higher probability to include errors in the transmitted frames as l increases. Increasing the value of FRI does not directly improve the throughput since it slows down the convergence speed of AMC when UAC channels are complex. That explains why a fixed value of FRI determined beforehand is not conducive to optimizing throughput and varies with different UAC channels. The difficulty of selecting an optimal FRI in advance emphasizes the significance of studying dynamic feedback scheduling strategies like TS-DQN to enhance AMC and optimize channel throughput. Additionally, our comparison of TS-DQN with the DNN proposed in [79] shows that TS-DQN exhibits improved robustness in complex channel conditions.

6.6 Summary

We transitioned from simulations to real experiments and tested our communication system. In our AMC policy, the upperbound BER estimation provides a safer selection of modulation and coding schemes compared with the point estimator of BER in Chapter 4 given the UAC channels are usually associated with high variability. The incorporation of TS-DQN for feedback scheduling combines the advantages of DQN and tree search that offer low

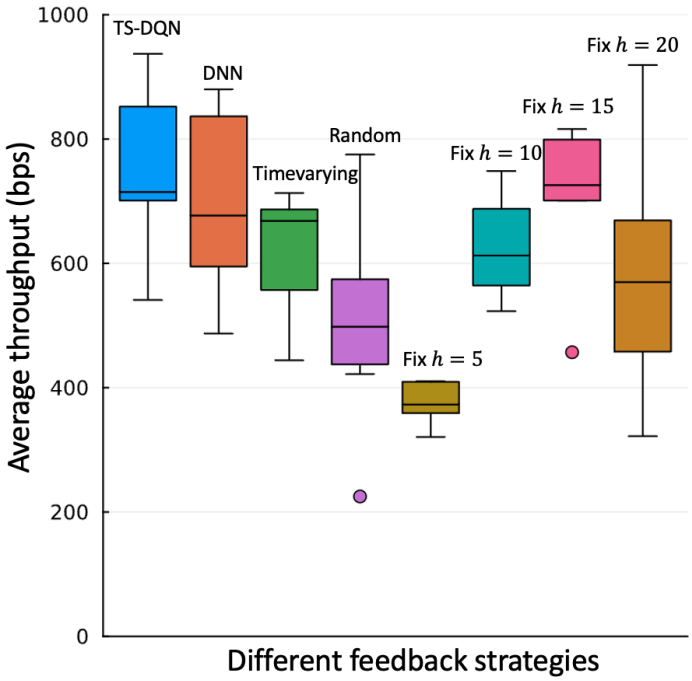


Figure 6.15: Results of average throughput with different feedback strategies under surrogate model 1.

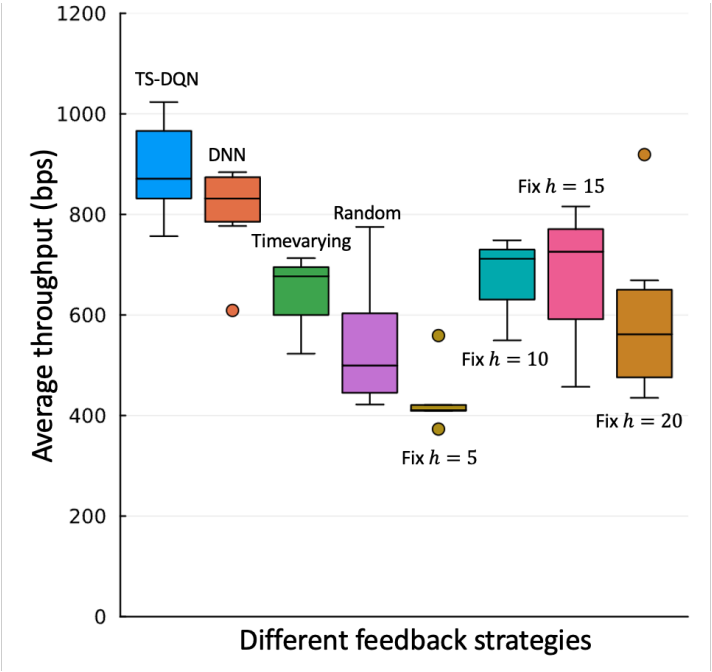


Figure 6.16: Results of average throughput with different feedback strategies under surrogate model 2.

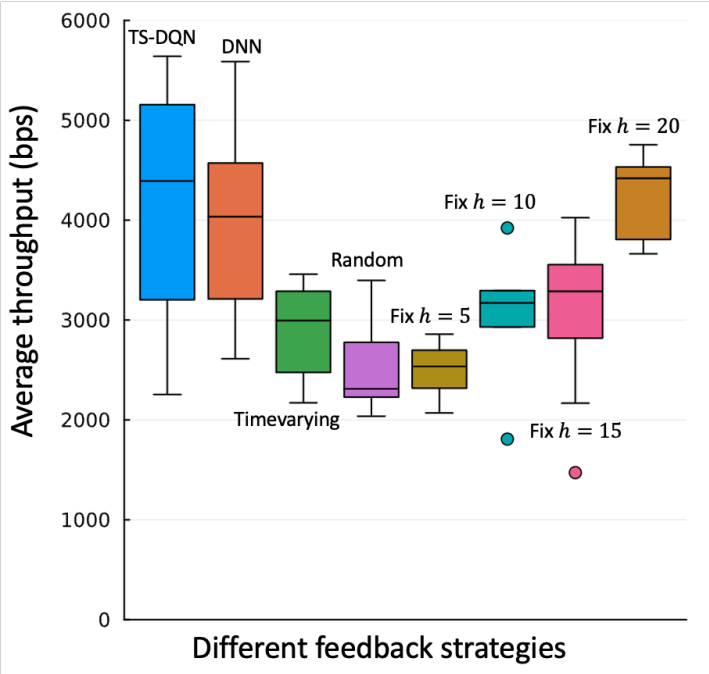


Figure 6.17: Results of average throughput with different feedback strategies under surrogate model 3.

computational complexity and demonstrate universality across different channel conditions. Our algorithm is modified and embedded on modems due to the additional information and unique setup on modems. These experiments complement simulations and provide a comprehensive understanding of system performance and capabilities.

Chapter 7

Joint Exploration & Exploitation in AMC and Feedback Scheduling

In the preceding chapters, we have formalized the AMC problem along with feedback scheduling as a MDP. Within this MDP framework, the agent balances exploration and exploitation while independently selecting MCS and FRI. This sequential approach entails consistently opting for the MCS first and subsequently determining the FRI based on the chosen MCS at each state. However, this manner of decision-making holds the potential to yield sub-optimal outcomes in the MDP due to ignoring the joint impacts of MCS and FRI. By treating these decisions as separate steps, the agent might miss out on opportunities to make choices that lead to higher overall rewards or better performance. In other words, a choice that seems optimal for the MCS might not be as effective when considering the FRI, and vice versa.To address this inherent concern and attain a more optimal decision-making approach, this chapter adopts an alternative strategy. Rather than proceeding sequentially, we intend to concurrently consider the interplay between MCS and FRI. This approach is grounded in the TS-DQN algorithm introduced in Chapter 5.

7.1 Problem Formulation

We focus on a problem where a TX node and a remote RX node are placed at a distance l in a varying underwater environment. The objective is to transmit N bits from the TX node to the remote RX node located at a distance l within the shortest possible time, where N can be any value. A total of $|\mathcal{A}|$ MCSs in action space $\mathcal{A}$ are available to transmit these N bits of information between the TX node and RX node. We aim to tune the MCS based on the varying channel conditions.

Performing AMC heavily relies on obtaining accurate CSI. The CSI, such as measured BER based on the number of bits corrected during FEC decoding, is acquired through feedback from the RX node. However, employing modulation schemes and coding rates blindly may lead to failed frame receptions at the RX node. Although such failures indicate that the BER exceeds a certain threshold, they hinder acquiring accurate BER for reliable AMC. To address this challenge, a "test" mode is introduced, where frames carrying known bits are transmitted over the DATA link. In this mode, the BER can be accurately computed as the transmitted frames are known, and the CSI is updated. When the "test" mode is disabled, the N unknown bits are encoded and transmitted to the RX node over the DATA link. In this case, the BER is approximated after demodulation and decoding of the frames at the RX node. All CSI, including BER measurements, are then encoded onto frames and sent back to the TX node via the CONTROL link, thereby improving the performance of AMC.

Obtaining CSI can introduce time costs in UAC due to the huge propagation

delays. Conversely, employing a modulation scheme across multiple frames without timely feedback can lead to suboptimal performance, resulting in a loss of received frames at the RX node and reduced throughput. This motivated an adaptive feedback strategy to decide the time to tune MCS and request feedback during transmission to optimize the channel throughput. In the following content, we use Feedback Report Interval (FRI) to represent the waiting time between every two feedback frames.

We formulate the sequential decision on MCS and FRI as well as the subsequent interaction with the environment to receive feedback as a MDP. An agent in the MDP works as an intelligent decision-making entity that engages in iterative interactions with the environment to learn a policy Π. Action space is $\mathcal{A} \times \mathcal{H}$ where $\mathcal{A}$ contains all possible modulation schemes, i.e., $\boldsymbol{a} \in \mathcal{A}$ and $\mathcal{H}$ includes all possible values of h_j, i.e., $h_j \in \mathcal{H}$. In the state space $\mathcal{S}$, a state $\boldsymbol{s_j}$ which encompasses knowledge related to decision-making of actions. The policy Π guides the agent for selecting actions from action space $\mathcal{A} \times \mathcal{H}$ to transmit N bits within the possible shortest time. $\Pi : \mathcal{S} \rightarrow |\mathcal{A} \times \mathcal{H}|$.

There are two main exploration and exploitation dilemmas. The first one appears when selecting the MCS $\boldsymbol{a}$. In Chapter 4, we adopted a dynamic ε-Greedy policy in (6.10) to help balance whether to explore new MCS to achieve potentially better reward or exploit the current knowledge of the BER upperbound estimation model $\zeta(\cdot; \boldsymbol{\theta_j}) + \eta_j$ in Section 6.2. The incorporation of GRP for the BER upperbound prediction provides estimates of uncertainty guides the agent's exploration in risk-sensitive decision-making, and generalizes well to unseen states which are particularly valuable in MDPs with large

state spaces. However, due to the absence of a holistic consideration of the intertwined impacts of MCS and FRI, the agent may overlook opportunities to arrive at decisions that yield elevated cumulative rewards or enhanced performance. For smaller bit-size file transmissions, excessive exploration can detrimentally affect channel throughput. However, a sufficiently large N can facilitate optimal average channel throughput. The second dilemma concerns determining the appropriate FRI value, balancing AMC strategy optimization speed, and maximizing N bit throughput. The round-trip frame exchange duration comprises the transmission duration of h_j frames, i.e., $h_j\tau_j$, the duration of frames containing modulation information τ_{m}, a two-way propagation delay $2\tau_{\mathrm{pd}}$, and the duration of frames containing feedback τ_{fd}, as shown in Fig. 3.1.

As depicted in Fig. 3.1, the system transitions from $\boldsymbol{s_{j-1}}$ to $\boldsymbol{s_j}$ upon the completion of the j^{th} FRI. Within the state transition, the measured BER $\boldsymbol{\epsilon_j}(\boldsymbol{a})$ from the updated CSI is utilized to train (6.8). The ratio of N bits that have been transmitted, denoted by a percentage value n'_j, along with the timestamps of frames exchange is recorded. The number of frames with and without the "test" mode enabled up to state $\boldsymbol{s_j}$ are respectively tracked by k_1 and k_2. The throughput r_j of FRI h_j is calculated as the measure of $(n'_j - n'_{j-1})N$ bits transmitted over the DATA link within a time period encompassing h_j frames with a transmission duration of τ_j each, along with the duration of frames containing modulation information τ_{m}, the feedback frame duration τ_{fd}, and a two-way propagation delay of $2\tau_{\mathrm{pd}}$, i.e.,

$$r_j = \frac{(n'_j - n'_{j-1})N}{h_j\tau_j + 2\tau_{\mathrm{pd}} + \tau_{\mathrm{fd}} + \tau_{\mathrm{m}}}. \tag{7.1}$$

The parameters n'_j, r_j, and the ratio $\bar{k} = \frac{k_1}{k_1+k_2}$ provide valuable insights into the communication performance under current policy Π. Intuitively, to determine whether the next modulation scheme $\boldsymbol{a}$ and FRI h_{j+1} as a function of n'_j, r_j, and $\bar{k}$, we utilize ML techniques since such a function is analytically unknown. We construct a model $\mathcal{F}(\cdot)$ with inputs n'_j, $\bar{k}$, and r_j to predict the values of all possible action pairs. The optimal $\boldsymbol{a}$ and h_{j+1} are determined by

$$\boldsymbol{a}, h_{j+1} = \underset{\boldsymbol{a}\in\mathcal{A}, h_{j+1}\in\mathcal{H}}{\operatorname{argmax}} \mathcal{F}\left(\left\{n'_j, \bar{k}, r_j\right\}; \boldsymbol{\vartheta_j} | \boldsymbol{a}, h_{j+1}\right), \tag{7.2}$$

where $\boldsymbol{\vartheta_j}$ denotes the parameters of $\mathcal{M}(\cdot)$, which are updated once CSI is received. Our model $\mathcal{F}(\cdot)$ is to choose the sequence of modulation schemes $\boldsymbol{a}$ and FRI h_j, $j = 1, \cdots, J$, where J is unknown, to transmit N bits within the shortest time and thereby maximize the throughput:

$$\begin{aligned} &\min \left(\sum_{i=1}^{J} h_i \tau_i + J(\tau_{\mathrm{fd}} + 2\tau_{\mathrm{pd}} + \tau_{\mathrm{m}}) \right) \\ &s.t. \quad n'_J \geq 1. \end{aligned} \tag{7.3}$$

7.2 TS-DQN for AMC and Feedback Scheduling

In this section, we aim to utilize our proposed TS-DQN framework to synergize the exploration and exploitation dilemma that existed in AMC and feedback scheduling. The BER upperbound estimation $\hat{\epsilon}_j(\boldsymbol{a})$ at state $\boldsymbol{s_j}$ is given in (6.8). The coding rate, i.e. LDPC rate $\rho(\epsilon(\boldsymbol{a}))$, for any $\boldsymbol{a} \in \mathcal{A}$ is determined by Table. 6.2. Given the uncoded data rate $d(\boldsymbol{a})$ of the modulation scheme $\boldsymbol{a}$, the coded data rate is calculated as $d(\boldsymbol{a})\rho(\epsilon(\boldsymbol{a}))$. If no LDPC rate is available, then

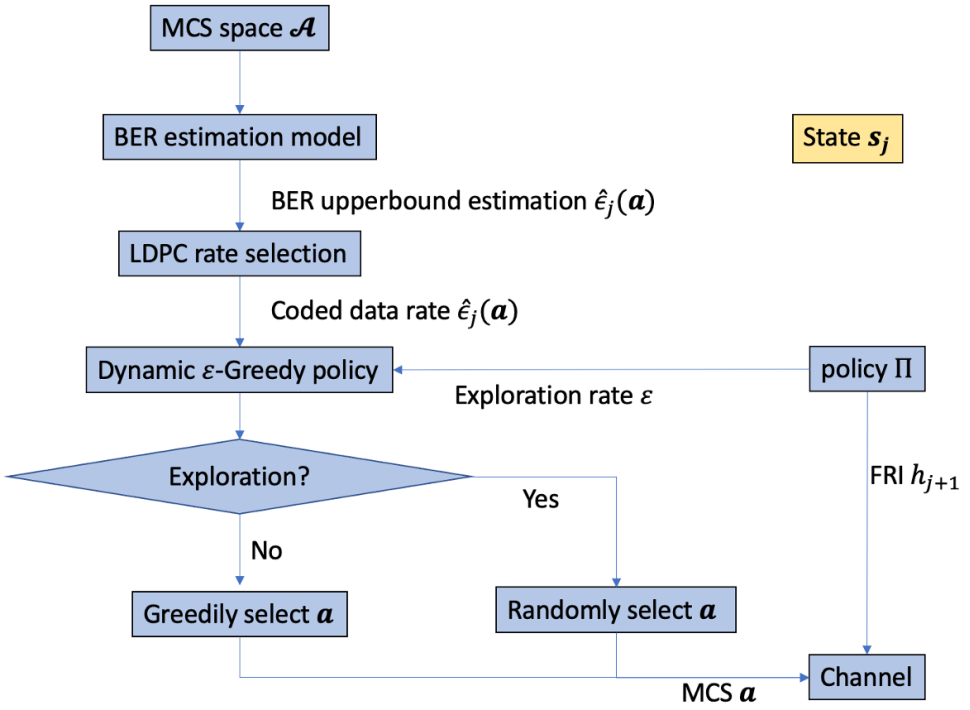


Figure 7.1: Structure of synergizing exploration & exploitation in AMC and feedback scheduling

the "test" mode is enabled. As detailed in (6.10), with our MDP framework, a ε-greedy policy is employed at state $\boldsymbol{s_j}$ to balance the trade-off between the exploration and exploitation in MCS selection given the estimated coded data rate $d(\boldsymbol{a})\rho(\hat{\epsilon}_j(\boldsymbol{a}))$. A heuristic method to gradually reduce the value of ε, the exploration rate, is well-aligned with the case channel knowledge accumulated along with transmission. However, in this Chapter, we will jointly consider the exploration and exploitation of MCS selection and feedback scheduling to achieve optimal throughput given different values of N. The structure of our strategy is shown in Fig. 7.1. The policy Π is a function $\mathcal{F}$ that takes n'_j, r_j, and $\bar{k}$ as inputs and outputs the exploration rate ε in MCS space $\mathcal{A}$ and value of FRI h_{j+1} from $\mathcal{H}$. For the exploration rate ε, a set E contains all possible values

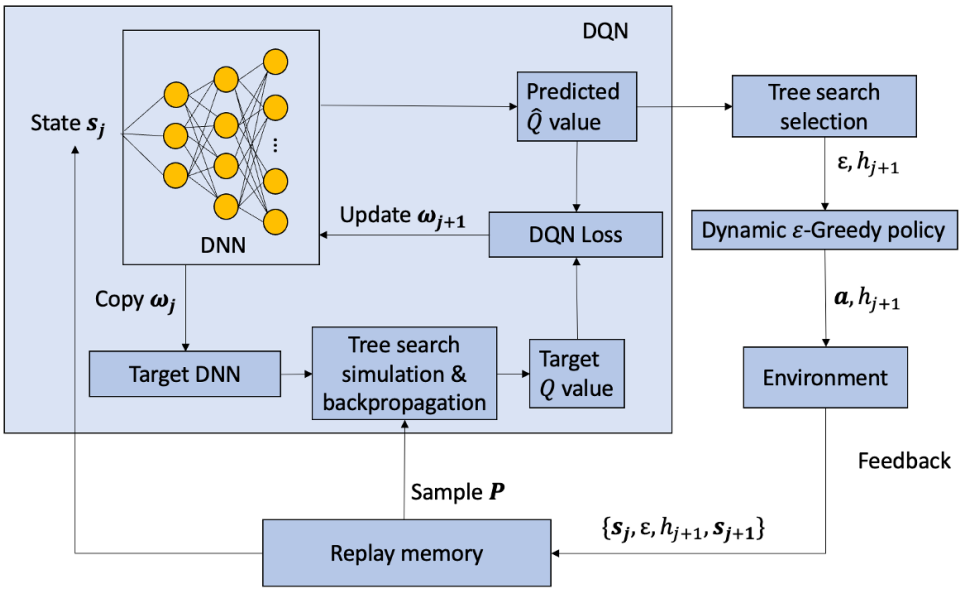


Figure 7.2: Framework of TS-DQN to determine the exploration rate and FRI value.

$\{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$ of ε. We then update (7.2) by

$$\varepsilon, h_{j+1} = \underset{\varepsilon \in E, h_{j+1} \in \mathcal{H}}{\operatorname{argmax}} \mathcal{F}\left(\left\{n'_j, \bar{k}, r_j\right\}; \boldsymbol{\vartheta_j} | \varepsilon, h_{j+1}\right), \tag{7.4}$$

TS-DQN benefits from the planning capabilities of tree search and the generalization capabilities of DQN. Based on the TS-DQN we proposed in Chapter 5, the TS-DQN to help evaluate ε, h_{j+1} is shown in Fig. 7.2.

The prediction of the Q-value function, i.e., $\hat{Q}(\boldsymbol{s_j}, \varepsilon, h_{j+1})$. $\hat{Q}(\boldsymbol{s_j}, \varepsilon, h_{j+1})$ approximates the throughput till terminal state $\boldsymbol{s_J}$ given any state-action pair $\{\varepsilon, h_{j+1}\}$. Agent prioritizes actions that are more likely to result in favorable long-term rewards. At state $\boldsymbol{s_j}$, the calculation of throughput R_j is the same

with (5.1) and the target Q-value is updated by

$$Q(\boldsymbol{s_j}, \varepsilon, h_{j+1}) = \frac{(n'_{j+1} - n'_j)N + (1 - n'_{j+1})N}{h_{j+1}\tau_{j+1} + \sum_{i=j+2}^{J} h_i\tau_i + (J-j)(\tau_{\mathrm{fd}} + 2\tau_{\mathrm{pd}} + \tau_{\mathrm{m}})}. \tag{7.5}$$

However, from state $\boldsymbol{s_{j+1}}$ to the terminal state $\boldsymbol{s_J}$, the number of transmitted bits and their corresponding duration are unknown as they have not been attempted. In the upcoming subsection 5.1.4, we will present the tree search approach for approximating the target Q-value $Q(\boldsymbol{s_j}, \varepsilon, h_{j+1})$.

In Fig. 7.1, the structure of DNN is one input layer with a size of all possible states, three hidden layers, and one output layer with the size of the $\mathcal{H}$. This DNN is utilized to model the analytical function between the state $\boldsymbol{s_j}$, and the estimated reward given the selected ε and any possible value of h_{j+1} which represents the predicted Q-value $\hat{Q}(\boldsymbol{s_j}, \varepsilon, h_{j+1})$, i.e.,

$$\hat{Q}(\boldsymbol{s_j}, \varepsilon, h_{j+1}) = \mathcal{F}\left(\boldsymbol{s_j}; \boldsymbol{\vartheta_j} | \varepsilon, h_{j+1}\right), \tag{7.6}$$

where $\boldsymbol{\vartheta_j}$ is the weights of model $\mathcal{F}(\cdot)$ and updated once CSI received. The agent employs a Replay Memory buffer to facilitate TS-DQN training, storing experiences $p_i = \{\boldsymbol{s_i}, \varepsilon, h_{i+1}, \boldsymbol{s_{i+1}}\}$, $i \in [0, j]$, up to state $\boldsymbol{s_j}$. During state transitions, the target DNN replicates the DNN model, adopting the weight parameters $\boldsymbol{\vartheta_j}$ updated in state $\boldsymbol{s_j}$. A batch, denoted by $\mathbf{P}$, comprises 32 samples from the Replay Memory. For each $p_i \in \mathbf{P}$, the target throughput from state $\boldsymbol{s_i}$ to terminal state $\boldsymbol{s_J}$, i.e., $Q(\boldsymbol{s_i}, \varepsilon, h_{i+1})$, is derived from the tree search depicted in Fig. 5.2. Throughout each memory replay for TS-DQN training, the target

DNN's parameters remain consistent across the batch. Similarly, the ADAM optimizer is employed to minimize the DQN Loss based on the mean squared loss between the predicted $\hat{Q}$-value and the target Q-value to update the weight parameters $\boldsymbol{\vartheta_j}$ to $\boldsymbol{\vartheta_{j+1}}$ of our state-value approximator $\mathcal{F}(\cdot)$, i.e.,

$$\boldsymbol{\vartheta_{j+1}} = \underset{\boldsymbol{\vartheta_{j+1}}}{\operatorname{argmin}} \left(\sum_{p_i \in \mathbf{P}} (Q(\boldsymbol{s_i}, \varepsilon, h_{i+1}) - \hat{Q}(\boldsymbol{s_i}, \varepsilon, h_{i+1}))^2 \right). \tag{7.7}$$

The tree search is the same as Fig. 5.2 with a three-section procedure: Selection, Simulation, and Backpropagation. When reaching the state $\boldsymbol{s_j}$, the three sections are depicted as follows.

- Selection: Agent selects the exploration rate ε and FRI h_{j+1} greedily:

$$\varepsilon, h_{j+1} = \underset{\varepsilon \in E, h_{j+1} \in \mathcal{H}}{\operatorname{argmax}} \hat{Q}(\boldsymbol{s_j}, \varepsilon, h_{j+1}). \tag{7.8}$$

  After determining ε and FRI h_{j+1}, modulation scheme $\boldsymbol{a}$ is determined by (6.10).

- Simulation: For each sample $p_i = \{\boldsymbol{s_i}, \varepsilon, h_{i+1}, \boldsymbol{s_{i+1}}\}$, $i \in [0, j]$ in $\mathbf{P}$, the agent simulates from $\boldsymbol{s_{i+1}}$ until transmission termination. The target DNN offers estimates for potential state-action pairs and selects the best action based on these estimates until the terminal state.

- Backpropagation: Simulation results, encompassing FRI values, throughput, and timestamps for each FRI between states, are backpropagated through the tree to refine the target Q-value in (7.5).

For each $p_i \in \mathbf{P}$, the corresponding target Q-value is utilized to train $\boldsymbol{\vartheta_{j+1}}$ based

on the loss function detailed in (7.7). Subsequently, the tuple $\{\boldsymbol{s_j}, \varepsilon, h_{j+1}, \boldsymbol{s_{j+1}}\}$ obtained from state $\boldsymbol{s_{j+1}}$ is updated using the newly computed value of $\boldsymbol{\omega_{j+1}}$ and archived in the Replay Memory.

7.3 Simulation and Results

As shown in Section 6.5, the surrogate model built based on [129] is capable of simulating actual BER variation given different transmission ranges and sea conditions. We, therefore, verify our algorithm in different underwater environments via simulations. Moreover, leveraging simulations facilitated the validation and exploration of the effectiveness of our proposed algorithms across various scenarios. The surrogate model and simulation setup are the same with Section 6.5. We compare our TS-DQN for balancing the exploration and exploitation to AMC and feedback scheduling with our previous strategy to select $\boldsymbol{a}$ by (6.10) and FRI with Random, Fixed, Time-varying, DNN, and TS-DQN (Presented in Section 6.5). The simulation stops with different numbers of bits $N \in \{10000, 100000, 300000\}$ have been transmitted. The possible number of subcarrier values are $n_{\mathrm{c}} = \{64, 128, 256, 512, 1024, 2048, 4096, 8192\}$, the value n_{p} can be selected from 0 to 8192, the possible occupy ratio of the bandwidth 24 KHz is $\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$. For different feedback strategies, transmission is executed 20 times. Meanwhile, the different UAC surrogate model is still listed as Table. 6.3.

Fig. 7.3, Fig. 7.4, and Fig. 7.5 showcase the throughput outcomes for various feedback strategies. In this chapter, we introduce the TS-DQN2 model, and the TS-DQN1 model was discussed in Chapter 6. The results from Random,

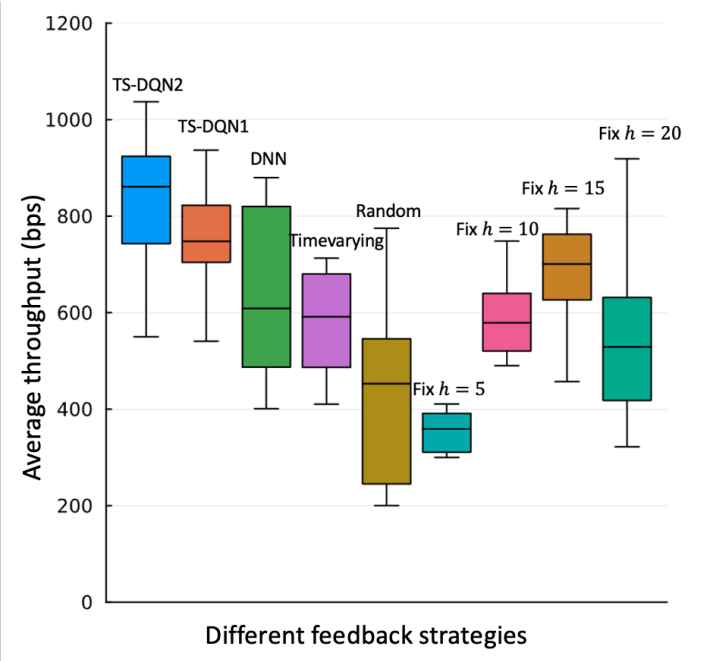


Figure 7.3: Results of average throughput with different feedback strategies given surrogate model 1.

Fixed, and Time-varying strategies align with our observations in Section 6.5. In Fig. 7.3 and Fig. 7.4, our TS-DQN2 notably outperforms TS-DQN1. Specifically, in the environment with a TX-RX distance of 100 m (as per surrogate model 3), most MCSs in the action space achieve error-free transmission. Consequently, learning optimal MCSs becomes easier, requiring less feedback for AMC model updates. Both TS-DQN2 and TS-DQN1 exhibit similar throughputs to the Fixed FRI strategy with $h = 20$. This observation aligns with the notion that in static and straightforward environments, learning becomes less challenging, and larger FRI values can save time on feedback while effectively learning the environment. These findings highlight the advantages of addressing exploration and exploitation challenges in both AMC and feedback scheduling concurrently.

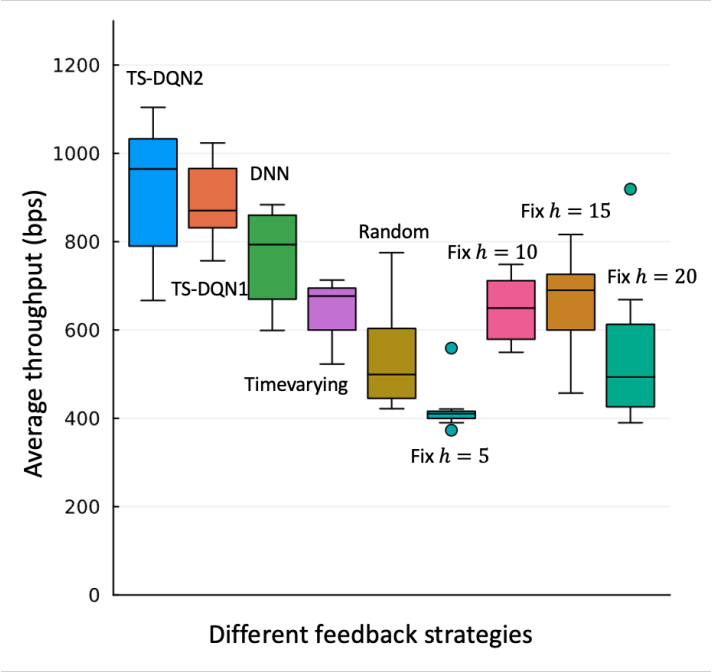


Figure 7.4: Results of average throughput with different feedback strategies given surrogate model 2.

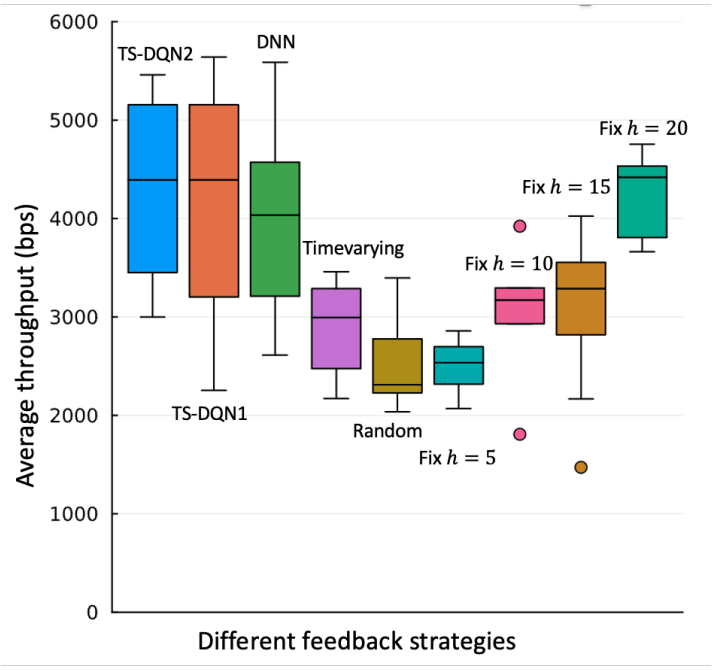


Figure 7.5: Results of average throughput with different feedback strategies given surrogate model 3.

7.4 Summary

Within the framework of MDPs, the exploration-exploitation trade-off remains crucial to obtaining an optimal policy. Addressing all actions jointly offers a comprehensive perspective on the entire action space, an approach that becomes particularly important when actions are intertwined or the outcomes of one action can modify the perceived utility of another. In this chapter, we emphasized the significance of integrating exploration and exploitation dynamics for all possible kinds of actions in MDPs, specifically concerning AMC and feedback scheduling. Instead of addressing these dynamics independently, our integrated methodology presented notable advantages in striving for a more globally optimal policy. Subsequent simulation outcomes validated these theoretical insights, highlighting the potential benefits of the integrated approach.

Chapter 8

Conclusions & Future Research

8.1 Conclusions

We set out to address aspects of AMC with feedback scheduling, primarily focusing on the optimization of channel throughput within one-to-one communication systems.

Initially, we delved into the merits and drawbacks of deploying data-driven techniques within AMC. The simulation results show our K-MCTS approach, which employs MCTS for a K-level look-ahead tree construction during the simulation step of MCTS, with superior performance across varied scenarios in optimizing the long-term throughput than some current policies, such as random, greedy, UCB. It was evident that while data-driven strategies hold promise, they are often associated with extensive computational complexity. The incorporation of channel physics knowledge within these algorithms, therefore, becomes necessary for AMC in UAC.

Next, we answered how to incorporate channel physics information into AMC strategy design and took the OFDM system as an example. A heuristic BER estimation model was proposed based on channel physics knowledge like channel coherence time, channel coherence bandwidth, and delay spread. The BER estimation from this model captures the BER median from the real-world BER

collected from Singapore waters. Furthermore, by leveraging the GPR model, our BER estimation managed to adeptly capture the fluctuations within the UAC channel, ensuring at least 75% frames are successfully transmitted during transmission.

Subsequently, we showed the channel throughput in the one-to-one system was affected not only by the performance of the AMC strategy but also by the two-way propagation delays due to the feedback acquisition in AMC. We proposed an algorithm framework, TS-DQN, to dynamically decide the time to obtain feedback and tune the modulation scheme. Experimental validations in the test tank and Singapore waters highlighted the superiority of TS-DQN over both conventional statistical paradigms and contemporary DL models for optimizing the channel throughput in the long term. Specifically, comparing our algorithm TS-DQN to schedule feedback to the best-fixed feedback policy, we reduced the time to transmit all N bits by up to 25%. Meanwhile, its robustness to UAC channel fluctuations and prowess in sequential decision-making were also evident in diverse UAC simulated channels.

In conclusion, our work furnishes an enriched understanding of the interplay between channel physics and AMC design, combined with insights into optimizing throughput in systems with significant two-way propagation delays. The TS-DQN framework proposed in this thesis can be applied in any MDP that emphasizes the long-term reward with complex exploration and exploitation dilemma and high-dimensional action or state space.

8.2 Future Work

In this thesis, the BER estimation model in AMC is built heuristically given the UAC channel physics knowledge. While the model may perform effectively under specific conditions for which it was designed, its heuristic nature can limit its ability to generalize across different channel conditions, especially those not considered during its formulation. Therefore, in the next step, we aim to design a hybrid algorithm that utilizes channel physics in data-driven techniques. In our work, we use BER prediction to aid the AMC and many studies have attempted the GRP algorithm to do the BER prediction. In developing a GPR-based BER estimation model in our future work, channel physics will be used to inform the GPR's kernel function selection. While standard kernel functions like RBF and Matérn are employed, we also contemplate composite or custom kernels incorporating UAC-specific knowledge. Furthermore, the hyperparameters of the kernel function are usually tuned using techniques like grid search, random search, or Bayesian optimization. Regularly refining these parameters can have a significant impact on the model's predictive capability. UAC channel physics can also be effective in hyperparameter optimization.

Our inquiry primarily navigates channel physics within OFDM-based AMC. However, myriad modulation techniques exist, each with unique channel physics implications. Exploring alternative techniques can reveal distinct channel physics which is crucial for AMC design. Beyond channel coherence time/bandwidth and delay spread, factors like path loss, shadowing, multipath propagation, Doppler spread, and aspects unique to UAC—like

sound speed profiles influenced by environmental variables—play crucial roles. Comprehending these facts is necessary for designing communication systems. Meanwhile, different modulation techniques are required for different communication requirements. A criterion for determining the parameters to be tuned in a communication system is also necessary. Ultimately, we will navigate the integration of varied channel physics elements into the AMC design tailored for specific UAC channels. This journey promises rich insights, poised to redefine AMC strategy paradigms.

Bibliography

[1] A. Song, M. Stojanovic, and M. Chitre, "Editorial underwater acoustic communications: Where we stand and what is next?" *IEEE Journal of Oceanic Engineering*, vol. 44, no. 1, 2019.

[2] B. Xu, X. Wang, Y. Guo, J. Zhang, and A. A. Razzaqi, "A novel adaptive filter for cooperative localization under time-varying delay and non-gaussian noise," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–15, 2021.

[3] A. Radosevic, R. Ahmed, T. M. Duman, J. G. Proakis, and M. Stojanovic, "Adaptive ofdm modulation for underwater acoustic communications: Design considerations and experimental results," *IEEE Journal of Oceanic Engineering*, vol. 39, no. 2, pp. 357–370, 2014.

[4] ——, "Adaptive ofdm modulation for underwater acoustic communications: Design considerations and experimental results," *IEEE journal of oceanic engineering*, vol. 39, no. 2, pp. 357–370, 2014.

[5] P. Xia, S. Zhou, and G. Giannakis, "Adaptive mimo-ofdm based on partial channel state information," *IEEE Transactions on Signal Processing*, vol. 52, no. 1, pp. 202–213, 2004.

[6] D. Love and R. Heath, "Limited feedback power loading for ofdm," in *IEEE MILCOM 2004. Military Communications Conference, 2004.*, vol. 1, 2004, pp. 71–77 Vol. 1.

[7] L. Huang, Q. Zhang, L. Zhang, J. Shi, and L. Zhangb, "Efficiency enhancement for underwater adaptive modulation and coding systems: via sparse principal component analysis," *IEEE Communications Letters*, pp. 1–1, 2020.

[8] M. R. Khan, B. Das, and B. B. Pati, "Channel estimation strategies for underwater acoustic (UWA) communication: An overview," *Journal of the Franklin Institute*, vol. 357, no. 11, pp. 7229–7265, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0016003220302325

[9] A. Radosevic, R. Ahmed, T. M. Duman, J. G. Proakis, and M. Stojanovic, "Adaptive ofdm modulation for underwater acoustic communications: Design considerations and experimental results," *IEEE journal of oceanic engineering*, vol. 39, no. 2, pp. 357–370, 2014.

[10] L. Liu, L. Cai, L. Ma, and G. Qiao, "Channel state information prediction for adaptive underwater acoustic downlink ofdma system: Deep neural networks based approach," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 9, pp. 9063–9076, 2021.

[11] J. P. Alan C. Farrell, "Performance of IEEE 802.11 mac in underwater wireless channels," *Procedia Computer Science*, vol. 10, no. 12, pp. 62–69, 2012. [Online]. Available: https://doi.org/10.1016/j.procs.2012.06.012

[12] A. Leon-Garcia and I. Widjaja, *Communication Networks: Fundamental Concepts and Key Architectures*, 1st ed. McGraw-Hill School Education Group, 1999.

[13] X. Guo, M. R. Frater, and M. J. Ryan, "Design of a propagation-delay-tolerant mac protocol for underwater acoustic sensor networks," *IEEE Journal of Oceanic Engineering*, vol. 34, no. 2, pp. 170–180, 2009.

[14] C.-C. Hsu, M.-S. Kuo, C.-F. Chou, and K. C.-J. Lin, "The elimination of spatial-temporal uncertainty in underwater sensor networks," *IEEE/ACM Transactions on Networking*, vol. 21, no. 4, pp. 1229–1242, 2012.

[15] J. Huang and R. Diamant, "Adaptive modulation for long-range underwater acoustic communication," *IEEE Transactions on Wireless Communications*, vol. 19, no. 10, pp. 6844–6857, 2020.

[16] P. Anjangi and M. Chitre, "Model-based data-driven learning algorithm for tuning an underwater acoustic link," in *2018 Fourth Underwater Communications and Networking Conference (UComms)*, 2018, pp. 1–5.

[17] K. Pelekanakis, L. Cazzanti, G. Zappa, and J. Alves, "Decision tree-based adaptive modulation for underwater acoustic communications," in *2016 IEEE Third Underwater Communications and Networking Conference (UComms)*, 2016, pp. 1–5.

[18] G. Qiao, Z. Babar, L. Ma, S. Liu, and J. Wu, "MIMO-OFDM underwater acoustic communication systems—A review," *Physical communication*, vol. 23, pp. 56–64, 2017.

[19] S. Zhou and Z. . e. Wang, *OFDM for underwater acoustic communications*, first;1; ed. Chichester, West Sussex, United Kingdom: Wiley, 2014, vol. 9781118458860.

[20] M. Chitre, "Underwater acoustic communications in warm shallow water channels," Ph.D. dissertation, Natinal University of Singapore, 2006.

[21] F.-L. Luo, *Machine Learning–Based Adaptive Modulation and Coding Design*, 2020, pp. 157–180.

[22] S. Kojima, K. Maruta, and C.-J. Ahn, "Adaptive modulation and coding using neural network based SNR estimation," *IEEE Access*, vol. 7, pp. 183 545–183 553, 2019.

[23] R. C. Daniels, C. M. Caramanis, and R. W. Heath, "Adaptation in convolutionally coded MIMO-OFDM wireless systems through supervised learning and SNR ordering," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 1, pp. 114–126, 2010.

[24] R. Daniels and R. W. Heath, "Online adaptive modulation and coding with support vector machines," in *2010 European Wireless Conference (EW)*, 2010, pp. 718–724.

[25] G. Xu and Y. Lu, "Channel and modulation selection based on support vector machines for cognitive radio," in *2006 International Conference on Wireless Communications, Networking and Mobile Computing*, 2006, pp. 1–4.

[26] J. Schnitzer, P. Prahladan, P. Rahimzadeh, C. Humble, J. Lee, J. Lee, K. Lee, and S. Ha, "Toward programmable docsis 4.0 networks: Adaptive modulation in ofdm channels," *IEEE Transactions on Network and Service Management*, vol. 18, no. 1, pp. 441–455, 2021.

[27] W. V. Mauricio, D. C. Araujo, F. Hugo Neto, F. Rafael Lima, and T. F. Maciel, "A low complexity solution for resource allocation and sdma grouping in massive mimo systems," in *2018 15th International Symposium on Wireless Communication Systems (ISWCS)*, 2018, pp. 1–6.

[28] S. Thrun and M. L. Littman, "Reinforcement learning: an introduction," *AI Magazine*, vol. 21, no. 1, pp. 103–103, 2000.

[29] J. P. Leite, P. H. P. de Carvalho, and R. D. Vieira, "A flexible framework based on reinforcement learning for adaptive modulation and coding in ofdm wireless systems," in *2012 IEEE Wireless Communications and Networking Conference (WCNC)*, 2012, pp. 809–814.

[30] W. Su, J. Lin, K. Chen, L. Xiao, and C. En, "Reinforcement learning-based adaptive modulation and coding for efficient underwater communications," *IEEE Access*, vol. 7, pp. 67 539–67 550, 2019.

[31] X. Ye, Y. Yu, and L. Fu, "Deep reinforcement learning based link adaptation technique for lte/nr systems," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 6, pp. 7364–7379, 2023.

[32] P. H. de Carvalho, R. Vieira, and J. Leite, "A continuous-state reinforcement learning strategy for link adaptation in ofdm wireless systems," *Journal of Communication and Information Systems*, vol. 30, no. 1, Jun. 2015. [Online]. Available: https://jcis.sbrt.org.br/jcis/article/view/16

[33] D. Lee, Y. G. Sun, S. H. Kim, I. Sim, Y. M. Hwang, Y. Shin, D. I. Kim, and J. Y. Kim, "Dqn-based adaptive modulation scheme over wireless communication channels," *IEEE Communications Letters*, pp. 1–1, 2020.

[34] L. Huang, Q. Zhang, W. Tan, Y. Wang, L. Zhang, C. He, and Z. Tian, "Adaptive modulation and coding in underwater acoustic communications: a machine learning perspective," *EURASIP Journal on Wireless Communications and Networking*, vol. 2020, pp. 1687–1499, 2020.

[35] C. Wang, Z. Wang, W. Sun, and D. R. Fuhrmann, "Reinforcement learning-based adaptive transmission in time-varying underwater acoustic channels," *IEEE Access*, vol. 6, pp. 2541–2558, 2018.

[36] Q. Fu and A. Song, "Adaptive modulation for underwater acoustic communications based on reinforcement learning," in *OCEANS 2018 MTS/IEEE Charleston*, 2018, pp. 1–8.

[37] L. Jing, C. Dong, C. He, W. Shi, and H. Yin, "Adaptive modulation and coding for underwater acoustic communications based on data-driven learning algorithm," *Remote Sensing*, vol. 14, no. 23, 2022. [Online]. Available: https://www.mdpi.com/2072-4292/14/23/5959

[38] Y. Zhang, J. Zhu, H. Wang, X. Shen, B. Wang, and Y. Dong, "Deep reinforcement learning-based adaptive modulation for underwater acoustic communication with outdated channel state information," *Remote Sensing*, vol. 14, no. 16, p. 3947, Aug 2022. [Online]. Available: http://dx.doi.org/10.3390/rs14163947

[39] Z. Tang, R. C. Cannizzaro, G. Leus, and P. Banelli, "Pilot-assisted time-varying channel estimation for ofdm systems," *IEEE Transactions on Signal Processing*, vol. 55, no. 5, pp. 2226–2238, 2007.

[40] L. Dai, Z. Wang, and Z. Yang, "Spectrally efficient time-frequency training ofdm for mobile large-scale mimo systems," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 2, pp. 251–263, 2013.

[41] M. Chitre, S. Shahabudeen, and M. Stojanovic, "Underwater acoustic communications and networking: Recent advances and future challenges," *Marine technology society journal*, vol. 42, no. 1, pp. 103–116, 2008.

[42] M. Stojanovic and J. Preisig, "Underwater acoustic communication channels: Propagation models and statistical characterization," *IEEE Communications Magazine*, vol. 47, no. 1, pp. 84–89, 2009.

[43] L. Lanbo, Z. Shengli, and C. Jun-Hong, "Prospects and problems of wireless communication for underwater sensor networks," *Wireless Communications and Mobile Computing*, vol. 8, no. 8, pp. 977–994, 2008.

[44] A. Benson, J. Proakis, and M. Stojanovic, "Towards robust adaptive acoustic communications," in *OCEANS 2000 MTS/IEEE Conference and Exhibition. Conference Proceedings (Cat. No.00CH37158)*, vol. 2, 2000, pp. 1243–1249 vol.2.

[45] S. Barua, Y. Rong, S. Nordholm, and P. Chen, "Adaptive modulation for underwater acoustic ofdm communication," in *OCEANS 2019 - Marseille*, 2019, pp. 1–5.

[46] P. Schniter, "Low-complexity equalization of OFDM in doubly selective channels," *IEEE Transactions on Signal Processing*, vol. 52, no. 4, pp. 1002–1011, 2004.

[47] S. Ohno and G. Giannakis, "Capacity maximizing MMSE-optimal pilots for wireless ofdm over frequency-selective block rayleigh-fading channels," *IEEE Transactions on Information Theory*, vol. 50, no. 9, pp. 2138–2145, 2004.

[48] Y. Yao and G. Giannakis, "Rate-maximizing power allocation in ofdm based on partial channel knowledge," *IEEE Transactions on Wireless Communications*, vol. 4, no. 3, pp. 1073–1083, 2005.

[49] S. He and M. Torkelson, "Effective SNR estimation in OFDM system simulation," in *IEEE GLOBECOM 1998 (Cat. NO. 98CH36250)*, vol. 2, 1998, pp. 945–950 vol.2.

[50] S. Gao, L. Qian, D. R. Vaman, and Q. Qu, "Energy efficient adaptive modulation in wireless cognitive radio sensor networks," in *2007 IEEE International Conference on Communications*, 2007, pp. 3980–3986.

[51] P. H. Tan, Y. Wu, and S. Sun, "Link adaptation based on adaptive modulation and coding for multiple-antenna OFDM system," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 8, pp. 1599–1606, 2008.

[52] F. Peng, J. Zhang, and W. E. Ryan, "Adaptive modulation and coding for ieee 802.11n," in *2007 IEEE Wireless Communications and Networking Conference*, 2007, pp. 656–661.

[53] L. Rugini and P. Banelli, "BER of OFDM systems impaired by carrier frequency offset in multipath fading channels," *IEEE Transactions on Wireless Communications*, vol. 4, no. 5, pp. 2279–2288, 2005.

[54] M. Luo, G. Villemaud, J.-M. Gorce, and J. Zhang, "Realistic prediction of BER for adaptive OFDM systems," in *2013 7th European Conference on Antennas and Propagation (EuCAP)*, 2013, pp. 1504–1508.

[55] M. N. Rajesh, B. K. Shrisha, N. Rao, and H. V. Kumaraswamy, "An analysis of ber comparison of various digital modulation schemes used for adaptive modulation," in *2016 IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT)*, 2016, pp. 241–245.

[56] Y. R. Zheng, J. Wu, and C. Xiao, "Turbo equalization for single-carrier underwater acoustic communications," *IEEE Communications Magazine*, vol. 53, no. 11, pp. 79–87, 2015.

[57] B. Mounika and B. K. Priya, "Analysis and comparison of different channel coding techniques for underwater channel using awgn and acoustic channel," in *2018 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT)*, 2018, pp. 1664–1669.

[58] H. Wu, Y. Li, Y. Hu, B. Tang, and Z. Bao, "On optimizing effective rate for random linear network coding over burst-erasure relay links," *IEEE Wireless Communications Letters*, vol. 8, no. 2, pp. 588–591, 2019.

[59] E. Fitzgerald, M. Pióro, and A. Tomaszewski, "Energy versus throughput optimisation for machine-to-machine communication," *Sensors*, vol. 20, no. 15, 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/15/4122

[60] P. Chaporkar and A. Proutiere, "Adaptive network coding and scheduling for maximizing throughput in wireless networks," in *Proceedings of the 13th Annual ACM International Conference on Mobile Computing and Networking*, ser. MobiCom '07. New York, NY, USA: Association for Computing Machinery, 2007, p. 135–146. [Online]. Available: https://doi.org/10.1145/1287853.1287870

[61] H. Ju and R. Zhang, "Throughput maximization in wireless powered communication networks," *IEEE Transactions on Wireless Communications*, vol. 13, no. 1, pp. 418–428, 2014.

[62] J. Xu, K. Li, and G. Min, "Reliable and energy-efficient multipath communications in underwater sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 7, pp. 1326–1335, 2012.

[63] G. Barreto, D. H. Simão, M. E. Pellenz, R. D. Souza, E. Jamhour, M. C. Penna, G. Brante, and B. S. Chang, "Energy-efficient channel coding strategy for underwater acoustic networks," *Sensors (Basel, Switzerland)*, vol. 17, no. 4, p. 728, 2017.

[64] Y. Zhang and Q. Ma, "Adaptive modulation and coding with cooperative transmission in mimo fading channels," in *Proceedings of the 2015 4th National Conference on Electrical, Electronics and Computer Engineering*. Atlantis Press, 2015/12, pp. 1363–1367. [Online]. Available: https://doi.org/10.2991/nceece-15.2016.240

[65] M.-F. Tsai, N. Chilamkurti, C.-K. Shieh, and A. Vinel, "Mac-level forward error correction mechanism for minimum error recovery overhead and retransmission," *Mathematical and Computer Modelling*, vol. 53, no. 11, pp. 2067–2077, 2011, recent Advances in Simulation and Mathematical Modeling of Wireless Networks. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S089571771000258X

[66] E. Lucas and Z. Wang, "Performance prediction of underwater acoustic communications based on channel impulse responses," *Applied Sciences*, vol. 12, no. 3, 2022. [Online]. Available: https://www.mdpi.com/2076-3417/12/3/1086

[67] G. Zhu, B. Feng, and W. Liu, "A ber model for turbo codes on awgn channel," in *Proceedings of 2005 IEEE International Workshop on VLSI Design and Video Technology, 2005.*, 2005, pp. 419–422.

[68] J. Laster, J. Reed, and W. Tranter, "Bit error rate estimation using probability density function estimators," *IEEE Transactions on Vehicular Technology*, vol. 52, no. 1, pp. 260–267, 2003.

[69] A.-A. Enescu, B.-M. Sandoi, and C.-G. Dinu, "A low-complexity bit error rate estimation algorithm for wireless digital receivers," in *2014 10th International Conference on Communications (COMM)*, 2014, pp. 1–4.

[70] S. Awino, T. J. O. Afullo, M. Mosalaosi, and P. O. Akuon, "Gmm estimation and ber of bursty impulsive noise in low-voltage plc networks," in *2019 PhotonIcs Electromagnetics Research Symposium - Spring (PIERS-Spring)*, 2019, pp. 1828–1834.

[71] R. Holzlöhner and C. R. Menyuk, "Use of multicanonical monte carlo simulations to obtain accurate bit error rates in optical communications systems," *Optics letters*, vol. 28, no. 20, pp. 1894–1896, 2003.

[72] B. Mazzeo and M. Rice, "On monte carlo simulation of the bit error rate," in *2011 IEEE International Conference on Communications (ICC)*, 2011, pp. 1–5.

[73] C. Fang, Q. Huang, F. Yang, X. Zeng, X. Li, and C. Gu, "Efficient bit error rate estimation for high-speed link by bayesian model fusion," in *Proceedings of the 2015 Design, Automation amp; Test in Europe Conference amp; Exhibition*, ser. DATE '15. San Jose, CA, USA: EDA Consortium, 2015, p. 1024–1029.

[74] C. Fang, F. Yang, X. Zeng, and X. Li, "Bmf-bd: Bayesian model fusion on bernoulli distribution for efficient yield estimation of integrated circuits," in *Proceedings of the 51st Annual Design Automation Conference*, ser. DAC '14. New York, NY, USA: Association for Computing Machinery, 2014, p. 1–6. [Online]. Available: https://doi-org.libproxy1.nus.edu.sg/10.1145/2593069.2593099

[75] S. Saoudi, T. Derham, T. Ait-Idir, and P. Coupe, "A fast soft bit error rate estimation method," *EURASIP journal on wireless communications and networking*, vol. 2010, no. 1, 2010.

[76] R. Simeon, T. Kim, and E. Perrins, "Machine learning with gaussian process regression for time-varying channel estimation," in *ICC 2022 - IEEE International Conference on Communications*, 2022, pp. 3400–3405.

[77] Gowrishankar, R. Babu H.S., and P. Satyanarayana, "Neural network based ber prediction for 802.16e channel," in *2007 15th International Conference on Software, Telecommunications and Computer Networks*, 2007, pp. 1–5.

[78] A. Charrada, "SVM based on LMMSE for high-speed coded OFDM channel with normal and extended cyclic prefix," *Physical Communication*, vol. 29, pp. 288–295, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1874490717306146

[79] W. Shuangshuang, P. Anjangi, and M. Chitre, "Adaptive modulation and feedback strategy for an underwater acoustic link," in *2022 Sixth Underwater Communications and Networking Conference (UComms)*, 2022, pp. 1–5.

[80] S. De, "Channel adaptive stop-and-wait automatic repeat request protocols for short-range wireless links," *IET Communications*, vol. 6, pp. 2128–2137(9), September 2012. [Online]. Available: https://digital-library.theiet.org/content/journals/10.1049/iet-com.2011.0795

[81] P. Mukherjee and S. De, "cdip: Channel-aware dynamic window protocol for energy-efficient iot communications," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4474–4485, 2018.

[82] ——, "Reduced-feedback scheduling policies for energy-efficient mac," in *2021 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2021, pp. 1–6.

[83] S. Hong, S. Jo, and J. So, "Machine learning-based adaptive csi feedback interval," *ICT Express*, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405959521001545

[84] P. Mukherjee and S. De, "Dynamic feedback-based adaptive modulation for energy-efficient communication," *IEEE Communications Letters*, vol. 23, no. 5, pp. 946–949, 2019.

[85] Q. Liu, S. Zhou, and G. Giannakis, "Queuing with adaptive modulation and coding over wireless links: cross-layer analysis and design," *IEEE Transactions on Wireless Communications*, vol. 4, no. 3, pp. 1142–1153, 2005.

[86] M. Chitre, M. Motani, and S. Shahabudeen, "Throughput of networks with large propagation delays," *IEEE Journal of Oceanic Engineering*, vol. 37, no. 4, pp. 645–658, 2012.

[87] R. E. Bellman and S. E. Dreyfus, *Applied dynamic programming*. Princeton university press, 2015, vol. 2050.

[88] R. A. Howard, "Dynamic programming and markov processes." 1960.

[89] A. G. Barto, S. J. Bradtke, and S. P. Singh, "Learning to act using real-time dynamic programming," *Artificial intelligence*, vol. 72, no. 1-2, pp. 81–138, 1995.

[90] J. J. Murray, C. J. Cox, G. G. Lendaris, and R. Saeks, "Adaptive dynamic programming," *IEEE transactions on systems, man, and cybernetics, Part C (Applications and Reviews)*, vol. 32, no. 2, pp. 140–153, 2002.

[91] A. V. Kalia and B. C. Fabien, "On implementing optimal energy management for erev using distance constrained adaptive real-time dynamic programming," *Electronics*, vol. 9, no. 2, p. 228, 2020.

[92] T. Keller and M. Helmert, "Trial-based heuristic tree search for finite horizon mdps," in *Proceedings of the International Conference on Automated Planning and Scheduling*, vol. 23, 2013, pp. 135–143.

[93] D. Busatto-Gaston, D. Chakraborty, and J.-F. Raskin, "Monte carlo tree search guided by symbolic advice for mdps," *arXiv preprint arXiv:2006.04712*, 2020.

[94] J. Hostetler, A. Fern, and T. Dietterich, "State aggregation in monte carlo tree search," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 28, no. 1, 2014.

[95] L. Kocsis and C. Szepesvári, "Bandit based monte-carlo planning," in *European conference on machine learning*. Springer, 2006, pp. 282–293.

[96] R. Coulom, "Efficient selectivity and backup operators in monte-carlo tree search," in *International conference on computers and games*. Springer, 2006, pp. 72–83.

[97] M. Ghallab, D. Nau, and P. Traverso, *Automated Planning: theory and practice*. Elsevier, 2004.

[98] T. Keller and P. Eyerich, "Prost: Probabilistic planning based on uct," in *Proceedings of the International Conference on Automated Planning and Scheduling*, vol. 22, 2012, pp. 119–127.

[99] M. R. Amer, S. Todorovic, A. Fern, and S.-C. Zhu, "Monte carlo tree search for scheduling activity recognition," in *Proceedings of the IEEE international conference on computer vision*, 2013, pp. 1353–1360.

[100] Z. Hu, J. Tu, and B. Li, "Spear: Optimized dependency-aware task scheduling with deep reinforcement learning," in *2019 IEEE 39th international conference on distributed computing systems (ICDCS)*. IEEE, 2019, pp. 2037–2046.

[101] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot *et al.*, "Mastering the game of go with deep neural networks and tree search," *nature*, vol. 529, no. 7587, pp. 484–489, 2016.

[102] M. Świechowski, T. Tajmajer, and A. Janusz, "Improving hearthstone ai by combining mcts and supervised learning algorithms," Aug 14 2018, - © 2018. This work is published under http://arxiv.org.libproxy1.nus.edu.sg/licenses/nonexclusive-distrib/1.0/ (the "License"). Notwithstanding the ProQuest Terms and Conditions, you may use this content in accordance with the terms of the License; - 2022-12-16. [Online]. Available: http://libproxy1.nus.edu.sg/login?url=https://www-proquest-com.libproxy1.nus.edu.sg/working-papers/improving-hearthstone-ai-combining-mcts/docview/2092774635/se-2

[103] M. Świechowski, K. Godlewski, B. Sawicki, and J. Mańdziuk, "Monte carlo tree search: a review of recent modifications and applications," vol. 56, no. 3, pp. 2497–2562. [Online]. Available: https://doi.org/10.1007/s10462-022-10228-y

[104] W. B. Powell, "From reinforcement learning to optimal control: A unified framework for sequential decisions," Dec 18 2019. [Online]. Available: http://libproxy1.nus.edu.sg/login?url=https://www.proquest.com/working-papers/reinforcement-learning-optimal-control-unified/docview/2323280441/se-2

[105] C. Chen, D. Dong, H.-X. Li, and T.-J. Tarn, "Hybrid mdp based integrated hierarchical q-learning," *Science China Information Sciences*, vol. 54, pp. 2279–2294, 2011.

[106] K. Dong, Y. Wang, X. Chen, and L. Wang, "Q-learning with ucb exploration is sample efficient for infinite-horizon mdp," *arXiv preprint arXiv:1901.09311*, 2019.

[107] Y. An, S. Ding, S. Shi, and J. Li, "Discrete space reinforcement learning algorithm based on support vector machine classification," *Pattern Recognition Letters*, vol. 111, pp. 30–35, 2018.

[108] G. Tesauro *et al.*, "Temporal difference learning and td-gammon," *Communications of the ACM*, vol. 38, no. 3, pp. 58–68, 1995.

[109] L. Jing, C. Dong, C. He, W. Shi, and H. Yin, "Adaptive modulation and coding for underwater acoustic communications based on data-driven learning algorithm," *Remote Sensing*, vol. 14, no. 23, p. 5959, 2022. [Online]. Available: http://libproxy1.nus.edu.sg/login?url=https://www.proquest.com/scholarly-journals/adaptive-modulation-coding-underwater-acoustic/docview/2748562431/se-2

[110] X. Ye and L. Fu, "Deep reinforcement learning based mac protocol for underwater acoustic networks," in *Proceedings of the 14th International Conference on Underwater Networks & Systems*, 2019, pp. 1–5.

[111] X. Geng and Y. R. Zheng, "Exploiting propagation delay in underwater acoustic communication networks via deep reinforcement learning," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–12, 2022.

[112] A. Thomas, Z. Tian, and D. Barber, "Thinking fast and slow with deep learning and tree search," Dec 03 2017. [Online]. Available: http://libproxy1.nus.edu.sg/login?url=https://www.proquest.com/working-papers/thinking-fast-slow-with-deep-learning-tree-search/docview/2076517562/se-2

[113] K. Takada, H. Iizuka, and M. Yamamoto, "Reinforcement learning to create value and policy functions using minimax tree search in hex," *IEEE Transactions on Games*, vol. 12, no. 1, pp. 63–73, 2020.

[114] A. Agarwal, K. Muelling, and K. Fragkiadaki, "Model learning for look-ahead exploration in continuous control," vol. 33, no. 1, pp. 3151–3158. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/4181

[115] X. Liu, H. Lam, and Y. Peng, "Training deep q-network via monte carlo tree search for adaptive bitrate control in video delivery," 12 2022.

[116] E. B. Wilson, "Probable inference, the law of succession, and statistical inference," *Journal of the American Statistical Association*, vol. 22, no. 158, pp. 209–212, 1927.

[117] S. Wallis, "Binomial confidence intervals and contingency tests: Mathematical fundamentals and the evaluation of alternative methods," *Journal of Quantitative Linguistics*, vol. 20, no. 3, pp. 178–208, 2013. [Online]. Available: https://doi.org/10.1080/09296174.2013.799918

[118] L. Kocsis and C. Szepesvári, "Bandit based monte-carlo planning," in *Machine Learning: ECML 2006*, J. Fürnkranz, T. Scheffer, and M. Spiliopoulou, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 282–293.

[119] M. A. Chitre, J. R. Potter, and S. H. Ong, "Viterbi decoding of convolutional codes in symmetric ¡emphasis emphasistype="italic"¿¡/emphasis¿ -stable noise," *IEEE Transactions on Communications*, vol. 55, no. 12, pp. 2230–2233, 2007.

[120] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

[121] G. Chen, E. Rodriguez-Villegas, and A. J. Casson, "Chapter 5.1 - wearable algorithms: An overview of a truly multi-disciplinary problem," in *Wearable Sensors*, E. Sazonov and M. R. Neuman, Eds. Oxford: Academic Press, 2014, pp. 353–382. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780124186620000052

[122] K. Narayanan and J. Doherty, "A convex projections method for improved narrow-band interference rejection in direct-sequence spread-spectrum systems," *Communications, IEEE Transactions on*, vol. 45, pp. 772 – 774, 08 1997.

[123] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2017.

[124] S. Wu, M. A. Chitre, and P. Anjangi, "Monte carlo tree search and delay-aware feedback adaptation for underwater acoustic link tuning," 2021.

[125] M.-S. Kim, T.-S. Lee, T.-H. Im, and H.-L. Ko, "The analysis of coherence bandwidth and coherence time for underwater channel environments using experimental data in the west sea, korea," in *OCEANS 2016 - Shanghai*, 2016, pp. 1–4.

[126] M. Chitre, J. Potter, and O. Heng, "Underwater acoustic channel characterisation for medium-range shallow water communications," in *Oceans '04 MTS/IEEE Techno-Ocean '04 (IEEE Cat. No.04CH37600)*, vol. 1, 2004, pp. 40–45 Vol.1.

[127] A. Malarkodi, G. Latha, and S. Srinivasan, "Characterization of underwater acoustic communication channel," 2020.

[128] S.-M. Kim, S.-H. Byun, S.-G. Kim, D.-J. Kim, S. Kim, and Y.-K. Lim, "Underwater acoustic channel characterization at 6khz and 12khz in a shallow water near jeju island," in *2013 OCEANS - San Diego*, 2013, pp. 1–4.

[129] M. Chitre, "Underwater acoustics in the age of differentiable and probabilistic programming," UComms 2020 webinar, Dec 2020. [Online]. Available: https://www.facebook.com/watch/live/?v=2473971036238315

[130] A. Akinshin, "Quantile absolute deviation," 2022. [Online]. Available: https://arxiv.org/abs/2208.13459

[131] C. E. Rasmussen and C. K. I. Williams, *Gaussian processes for machine learning*. Cambridge, Mass: MIT Press, 2006.

[132] M. G. Genton, "Classes of kernels for machine learning: A statistics perspective," *J. Mach. Learn. Res.*, vol. 2, p. 299–312, mar 2002.

[133] Subnero, "Home page," https://subnero.com/, 2023, accessed: September 14, 2023.

[134] M. Chitre, T.-B. Koay, G. Deane, and G. Chua, "Variability in shallow water communication performance near a busy shipping lane," in *2021 Fifth Underwater Communications and Networking Conference (UComms)*, 2021, pp. 1–5.

[135] A. Shokrollahi, "Ldpc codes: An introduction," in *Coding, cryptography and combinatorics*. Springer, 2004, pp. 85–110.

[136] "UnetStack3: the underwater networks project," https://unetstack.net/, accessed: 2022.

List of Publications

(C1) S. Wu, M. Chitre and P. Anjangi, "Monte Carlo Tree Search and Delay-Aware Feedback Adaptation for Underwater Acoustic Link Tuning," *Global OCEANS 2021 San Diego– Porto — Tethys*, September 20-23, 2021

(C2) S. Wu, P. Anjangi and M. Chitre, "Adaptive Modulation and Feedback Strategy for an Underwater Acoustic Link," in *The 6th Underwater Communications Networking (UComms 2022), (Lerici, Italy)*, 30 August – 1 September 2022.

(J1) S. Wu, M. Chitre and P. Anjangi, "Adaptive Modulation and Coding with Feedback Scheduling in Practical Underwater Acoustic Communication," in *IEEE Journal of Oceanic Engineering*, (in preparation)